Mining Large Graphs

ECML/PKDD 2007 tutorial

Jure Leskovec and Christos Faloutsos

Machine Learning Department

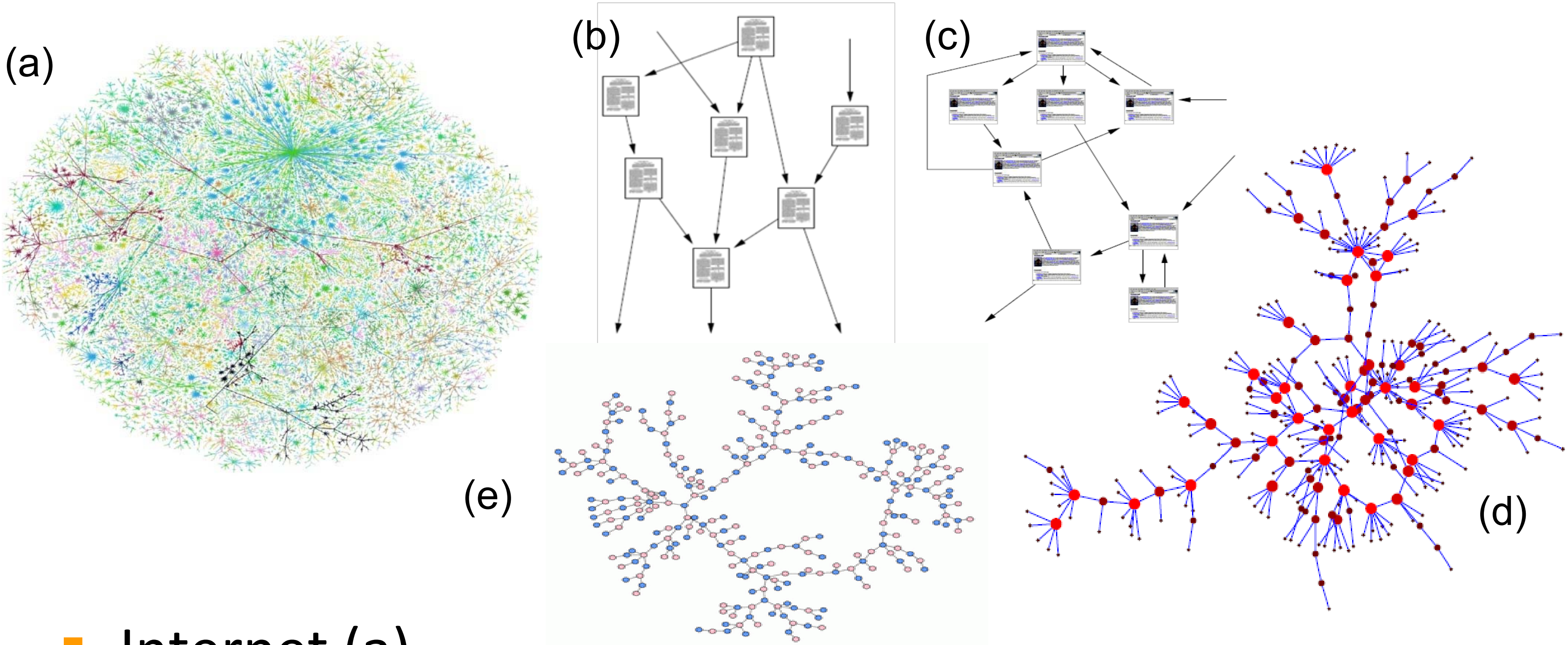
Carnegie Mellon

Joint work with: Lada Adamic, Deepay Chakrabarti, Natalie Glance, Carlos Guestrin, Bernardo Huberman, Jon Kleinberg, Andreas Krause, Mary McGlohon, Ajit Singh, and Jeanne VanBriesen.

Networks – Social and Technological

- Social network analysis: sociologists and computer scientists – influence goes both ways
  - Large-scale network data in "traditional" sociological domains
    - Friendship and informal contacts among people
    - Collaboration/influence in companies, organizations, professional communities, political movements, markets, …
  - Emerge of rich social structure in computing applications
    - Content creation, on-line communication, blogging, social networks, social media, electronic markets, …
    - People seeking information from other people vs. more formal channels: MySpace, del.icio.us, Flickr, LinkedIn, Yahoo Answers, Facebook, …

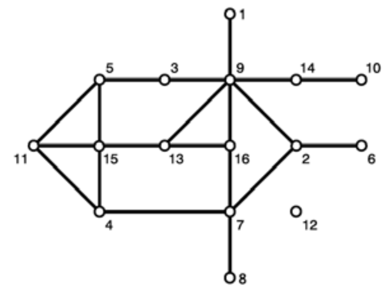
Examples of Networks



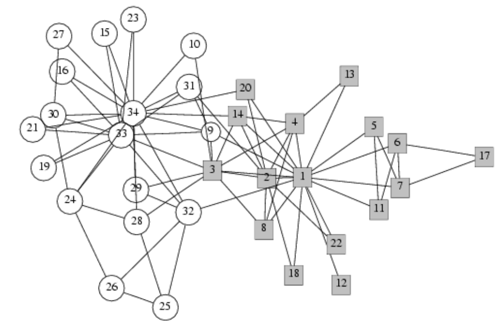
- Internet (a)
- Citation network (b)
- World Wide Web (c)
- Sexual network (d)
- Dating network(e)

Networks of the Real-world (1)

- Information networks:
  - World Wide Web: hyperlinks
  - Citation networks
  - Blog networks
- Social networks: people + interactions
  - Organizational networks
  - Communication networks
  - Collaboration networks
  - Sexual networks
  - Collaboration networks
- Technological networks:
  - Power grid
  - Airline, road, river networks
  - Telephone networks
  - Internet
  - Autonomous systems

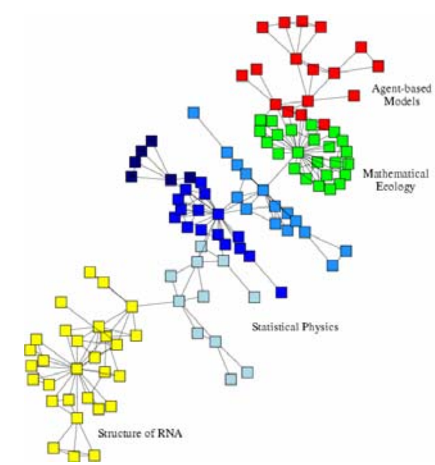


Florence families



Karate club network

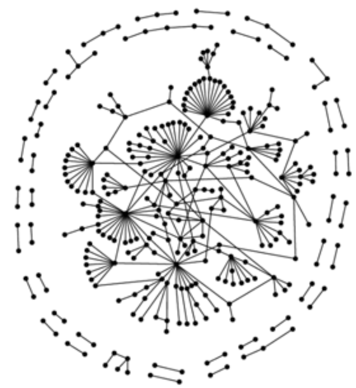
Friendship network



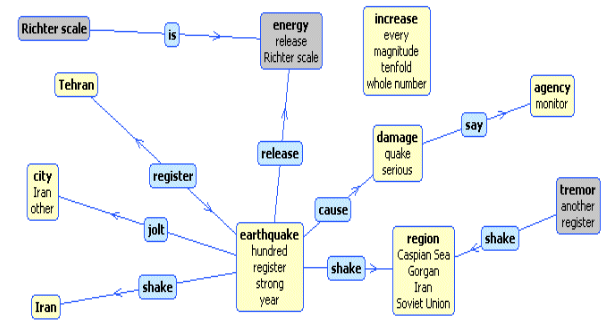
Collaboration network

Networks of the Real-world (2)

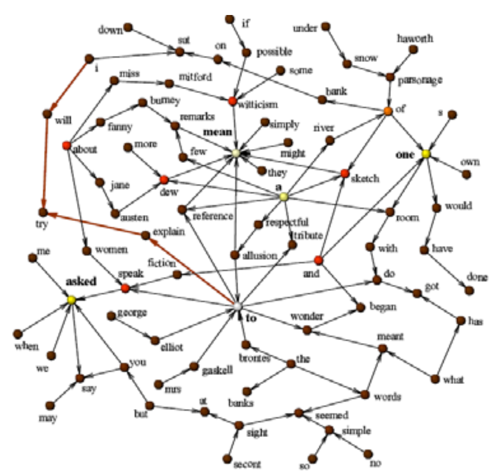
- Biological networks
  - metabolic networks
  - food web
  - neural networks
  - gene regulatory networks
- Language networks
  - Semantic networks
- Software networks
- …



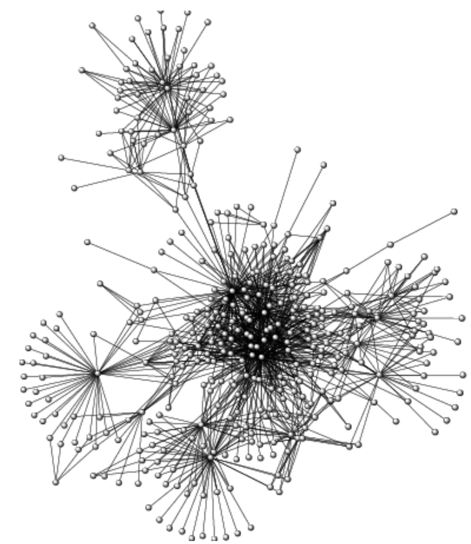
Yeast protein interactions



Semantic network



Language network



Software network

Networks as Phenomena

The emergence of 'cyberspace' and the World Wide Web is like the discovery of a new continent.

- Jim Gray, 1998 Turing Award address

- Complex networks as phenomena, not just designed artifacts
- What are the common patterns that emerge?

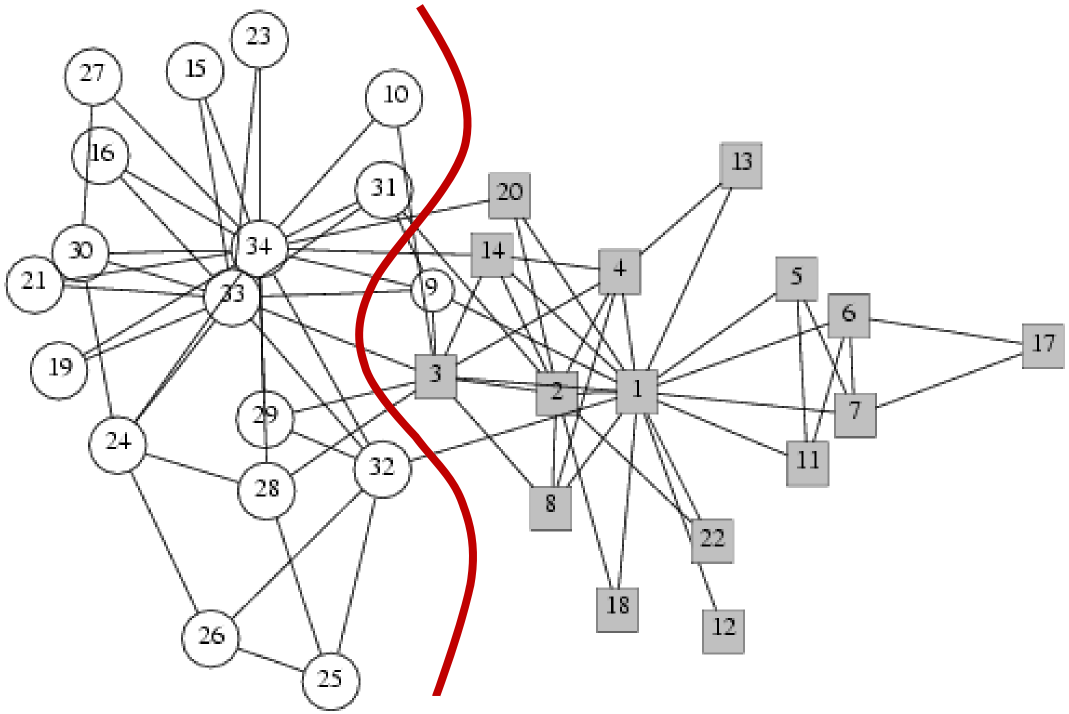
Models and Laws of Networks

We want Kepler's Laws of Motion for the Web.

- Mike Steuerwalt, NSF KDI workshop, 1998

- Need statistical methods and tools to quantify large networks
- What do we hope to achieve from models of networks?
  - Patterns and statistical properties of network data
  - Design principles and models
  - Understand why networks are organized the way they are (predict behavior of networked systems)

Mining Social Network Data

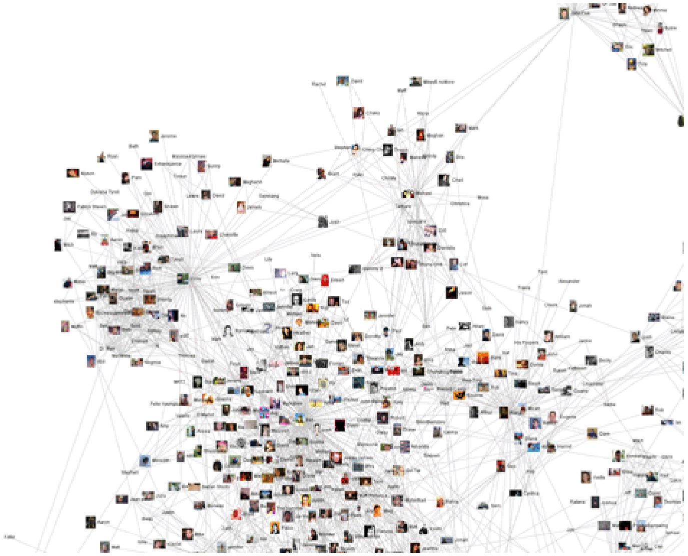


- Mining social networks has a long history in social sciences:
  - Wayne Zachary's PhD work (1970-72): observe social ties and rivalries in a university karate club
  - During his observation, conflicts led the group to split
  - Split could be explained by a minimum cut in the social network

Networks: Rich Data

- Traditional obstacle:

  Can only choose 2 of 3:
  - Large-scale
  - Realistic
  - Completely mapped



- Now: large on-line systems leave detailed records of social activity
  - On-line communities: MyScace, Facebook, LiveJournal
  - Email, blogging, electronic markets, instant messaging
  - On-line publications repositories, arXiv, MedLine

Networks: A Matter of Scale

- Network data spans many orders of magnitude:
  - 436-node network of email exchange over 3-months at corporate research lab [Adamic-Adar 2003]
  - 43,553-node network of email exchange over 2 years at a large university [Kossinets-Watts 2006]
  - 4.4-million-node network of declared friendships on a blogging community [Liben-Nowell et al. 2005, Backstrom et at. 2006]
  - 240-million-node network of all IM communication over a month on Microsoft Instant Messenger [Leskovec-Horvitz 2007]

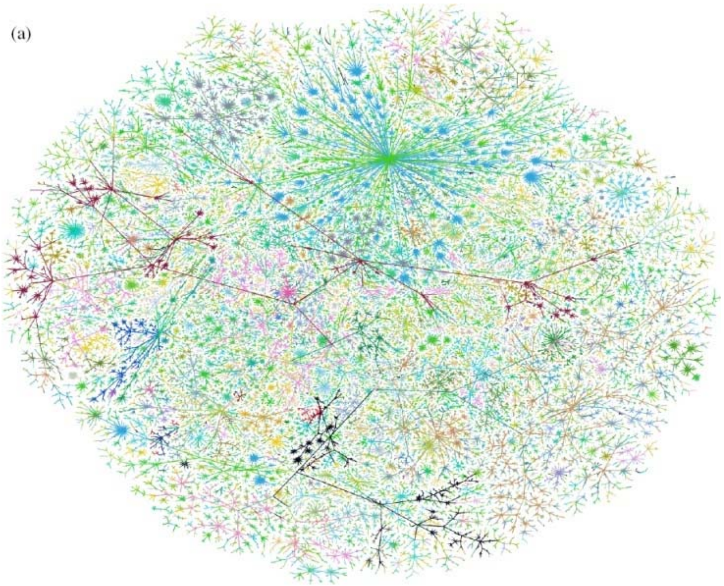
Networks: Scale Matters

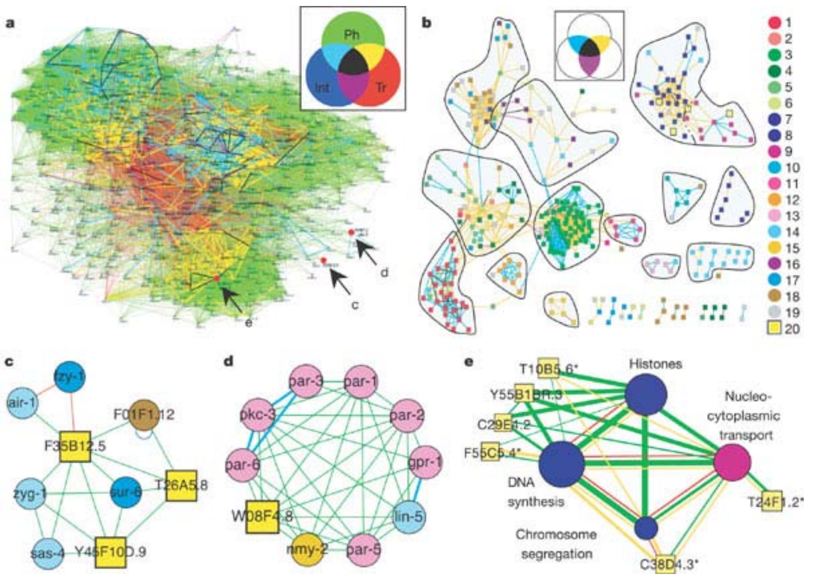
- How does massive network data compare to small-scale studies?
- Massive network datasets give you both more and less:
  - More: can observe global phenomena that are genuine, but literally invisible at smaller scales
  - Less: don't really know what any node or link means. Easy to measure things, hard to pose right questions
  - Goal: Find the point where the lines of research converge

Structure vs. Process

- What have we learned about large networks?
- We know about the structure: Many recurring patterns
  - Scale-free, small-world, locally clustered, bow-tie, hubs and authorities, communities, bipartite cores, network motifs, highly optimized tolerance
- We know about the processes and dynamics
  - Cascades, epidemic threshold, viral marketing, virus propagation, threshold model

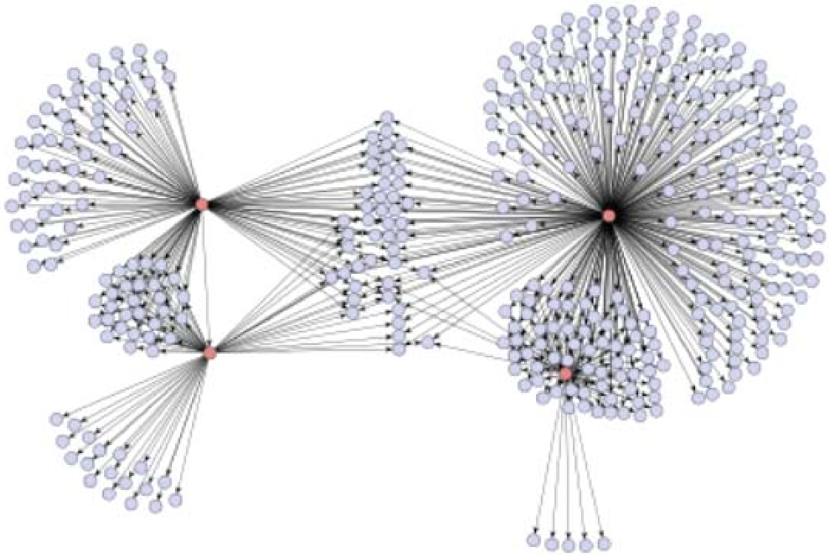
Structure of Networks

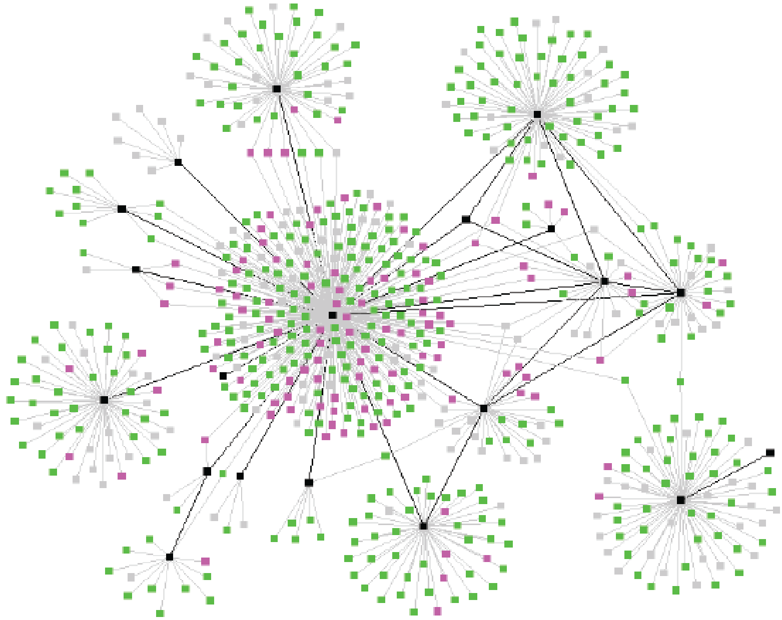




- What is the structure of a large network?
- Why and how did it became to have such structure?

Diffusion in Networks





- One of the networks is a spread of a disease, the other one is product recommendations
- Which is which? ☺

Tutorial outline

- Part 1: Structure and models for networks
  - What are properties of large graphs?
  - How do we model them?
- Part 2: Dynamics of networks
  - Diffusion and cascading behavior
  - How do viruses and information propagate?
- Part 3: Case studies
  - 240 million MSN instant messenger network
  - Graph projections: how does the web look like

Mining Large Graphs

Part 1: Structure and models of networks

Jure Leskovec and Christos Faloutsos

Machine Learning Department

Carnegie Mellon

Joint work with: Lada Adamic, Deepay Chakrabarti, Natalie Glance, Carlos Guestrin, Bernardo Huberman, Jon Kleinberg, Andreas Krause, Mary McGlohon, Ajit Singh, and Jeanne VanBriesen.

Part 1: Outline

- 1.1: Structural properties
  - What are the statistical properties of static and time evolving networks?
- 1.2: Models
  - How do we build models of network generations of evolution?
- 1.3: Fitting the models
  - How do we fit models?
  - How do we generate realistic looking graphs?

Part 1.1: Structural properties

What are statistical properties of networks across various domains?

Traditional approach

- Sociologists were first to study networks:
  - Study of patterns of connections between people to understand functioning of the society
  - People are nodes, interactions are edges
  - Questionares are used to collect link data (hard to obtain, inaccurate, subjective)
  - Typical questions: Centrality and connectivity
- Limited to small graphs (~100 nodes) and properties of individual nodes and edges

Motivation: New approach (1)

- **Large** networks (e.g., web, internet, on-line social networks) with millions of nodes
- Many traditional questions not useful anymore:
  - Traditional: What happens if a node u is removed?
  - Now: What percentage of nodes needs to be removed to affect network connectivity?
- Focus moves from a single node to study of **statistical** properties of the network as a whole

Motivation: New approach (2)

- How the network "looks like" even if I can't look at it?
- Need statistical methods and tools to quantify large networks
- 3 parts/goals:
  - Statistical properties of large networks
  - Models that help understand these properties
  - Predict behavior of networked systems based on measured structural properties and local rules governing individual nodes

Graphs and networks

- What is the simplest way to generate a graph?
- Random graph model (Erdos-Renyi model, Poisson random graph model):
  - Given n vertices connect each pair i.i.d. with probability p
- How good ("realistic") is this graph generator?

Small-world effect (1)

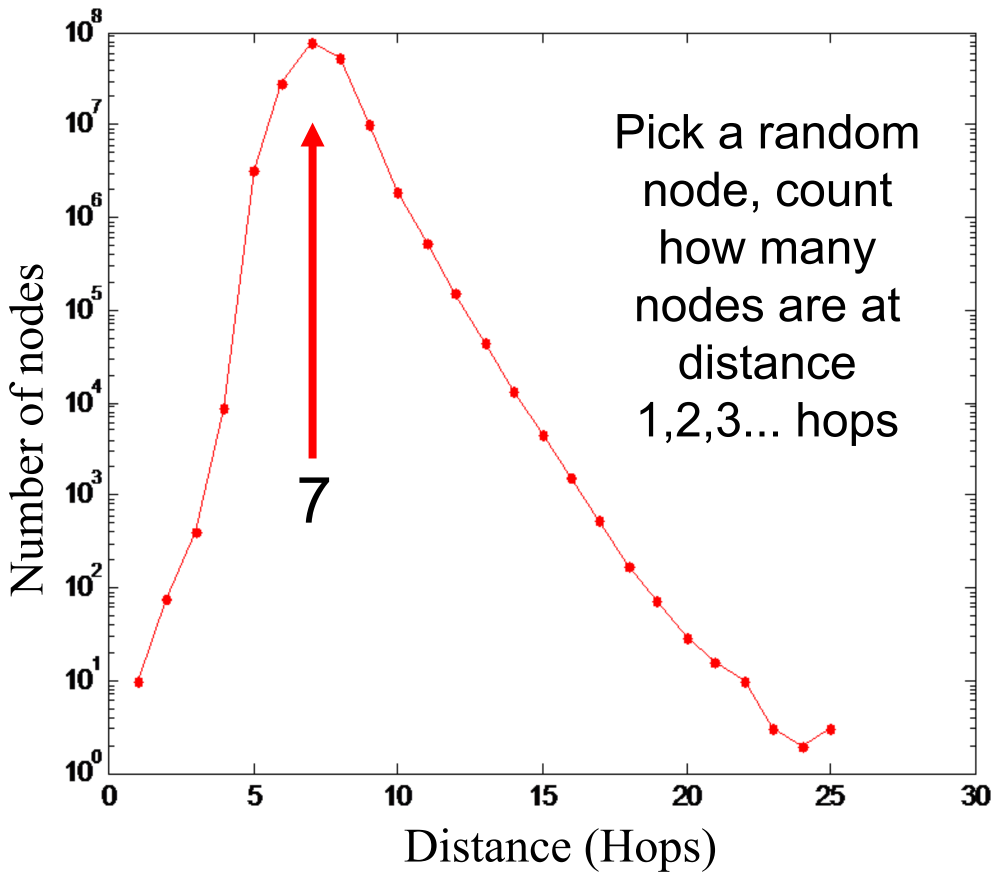
- Six degrees of separation [Milgram 60s]
  - Random people in Nebraska were asked to send letters to stockbrokes in Boston
  - Letters can only be passed to first-name acquantices
  - Only 25% letters reached the goal
  - But they reached it in about **6** steps
- Measuring path lengths:
  - Diameter (longest shortest path): $max\ d_{ij}$
  - Effective diameter: distance at which 90% of all connected pairs of nodes can be reached
  - Mean geodesic (shortest) distance l

$$\ell = \frac{1}{\frac{1}{2}n(n+1)} \sum_{i \geq j} d_{ij} \quad \text{or} \quad \ell^{-1} = \frac{1}{\frac{1}{2}n(n+1)} \sum_{i \geq j} d_{ij}^{-1}$$

Small-world effect (2)

[Leskovec&Horvitz,07]

- Distribution of shortest path lengths
- Microsoft Messenger network
  - 180 million people
  - 1.3 billion edges
  - Edge if two people exchanged at least one message in one month period

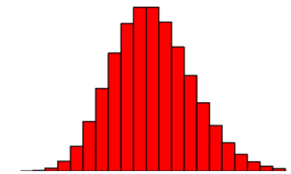


Small-world effect (3)

- If number of vertices within distance r grows exponentially with r, then mean shortest path length ℓ increases as $log\ n$
- Implications:
  - Information (viruses) spread quickly
  - Erdos numbers are small
  - Peer to peer networks (for navigation purposes)
- Shortest paths exists
- Humans are able to find the paths:
  - People only know their friends
  - People do not have the global knowledge of the network
- This suggests something special about the structure of the network
  - On a random graph short paths exists but no one would be able to find them

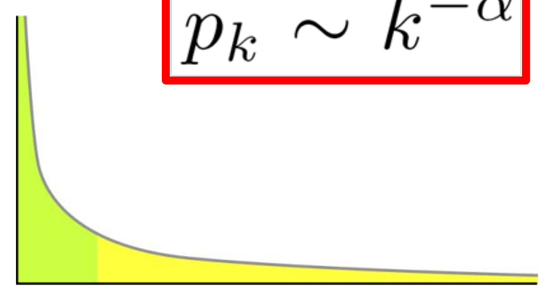
Degree distributions (1)

- Let p_k denote a fraction of nodes with degree k
- We can plot a histogram of p_k vs. k
- In a (Erdos-Renyi) random graph degree distribution follows Poisson distribution



- Degrees in real networks are heavily skewed to the right
- Distribution has a long tail of values that are far above the mean
- Power-law [Faloutsos et al], Zipf's law, Pareto's law, Long tail, Heavy-tail
- Many things follow Power-law:
  - Amazon sales,
  - word length distribution,
  - Wealth, Earthquakes, …

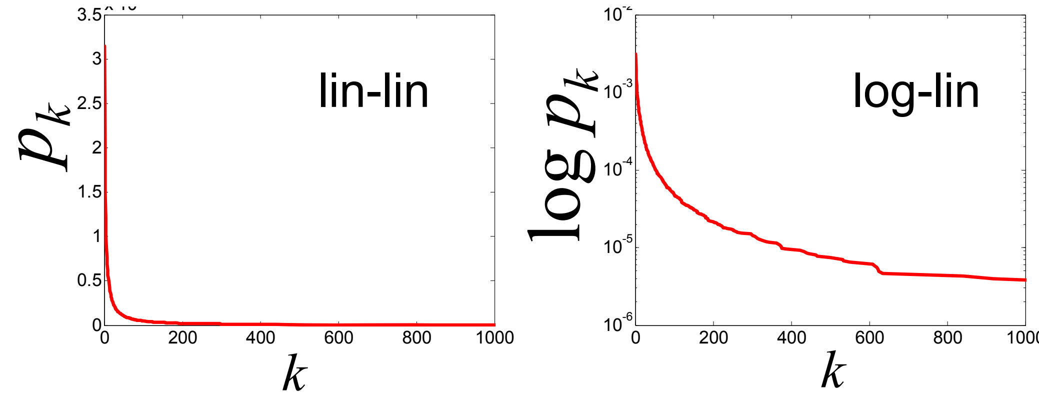
$$p_k \sim k^{-\alpha}$$

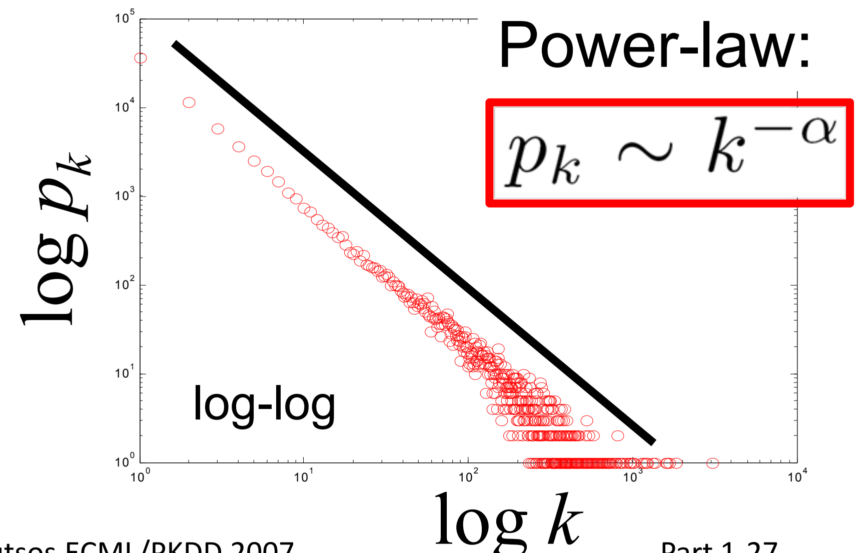


Degree distributions (2)

- Many real world networks contain hubs: highly connected nodes
- We can easily distinguish between exponential and power-law tail by plotting on log-lin and log-log axis
- Power-law is a line on log-log plot

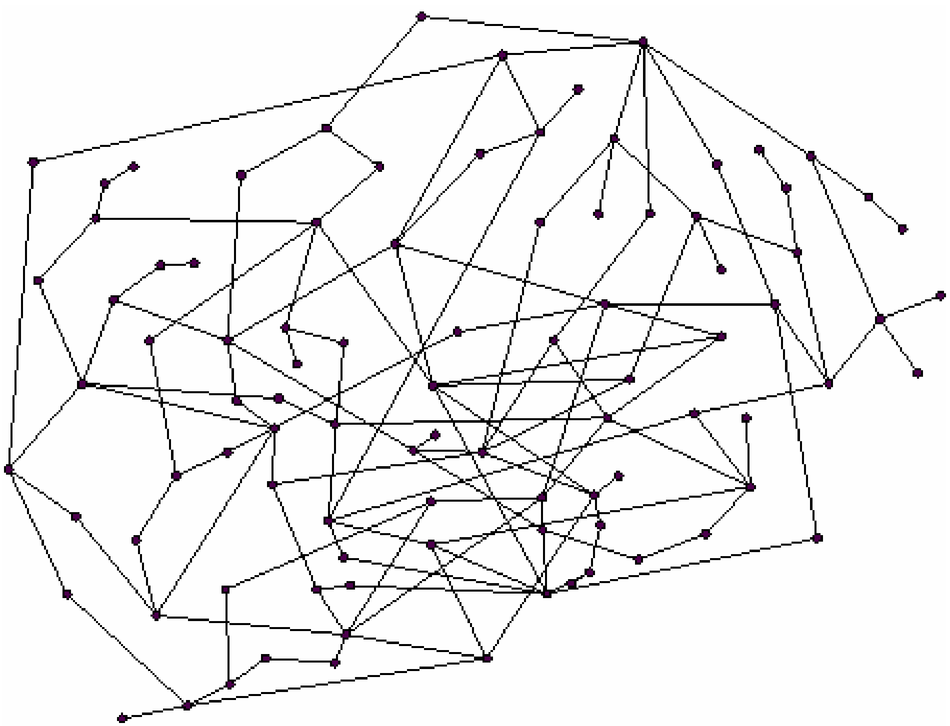
Degree distribution in a blog network
(plot the same data using different scales)



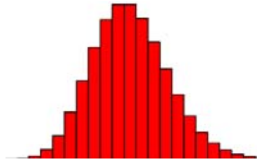


For statistical tests and estimation see Clauset-Shalizi-Newman 2007

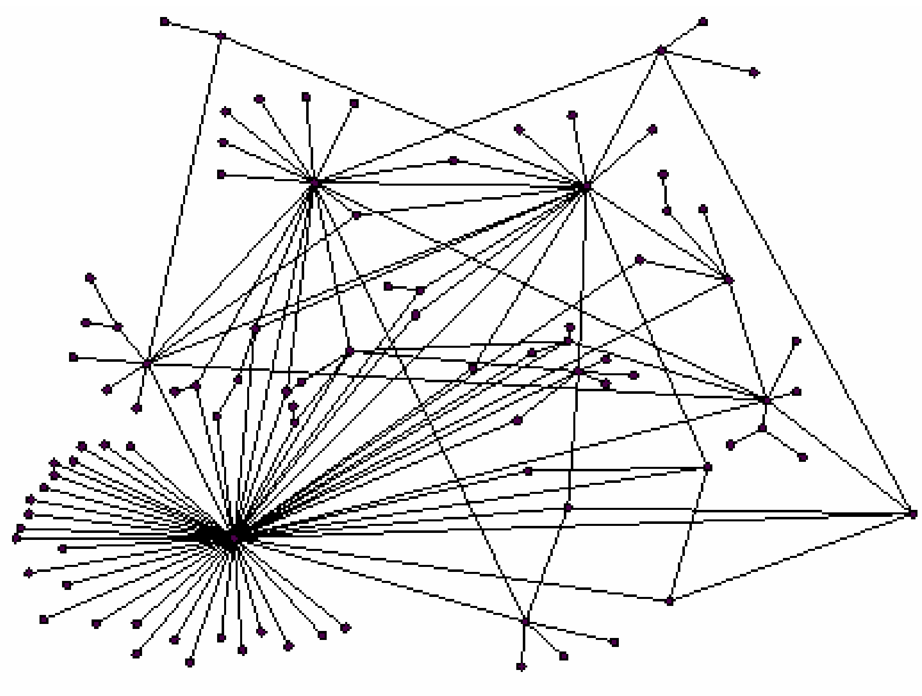
Poisson vs. Scale-free network



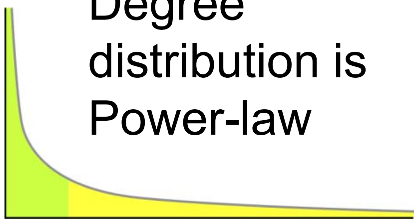
Poisson network
(Erdos-Renyi random graph)



Degree distribution is Poisson



Scale-free (power-law) network

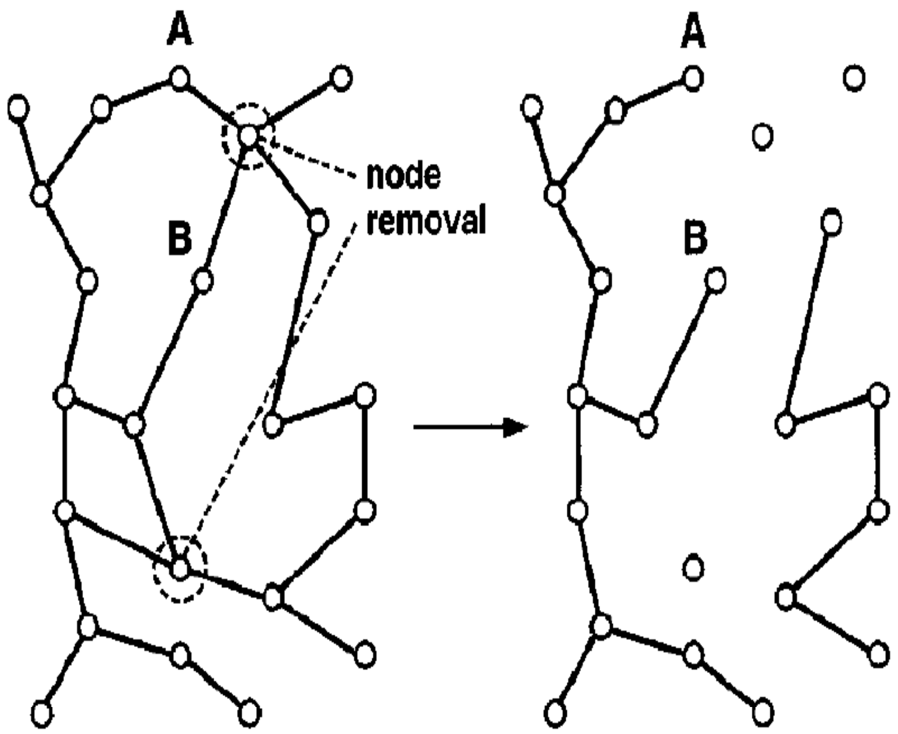


Degree distribution is Power-law

Function is scale free if:
$f(ax) = c\,f(x)$

Network resilience (1)

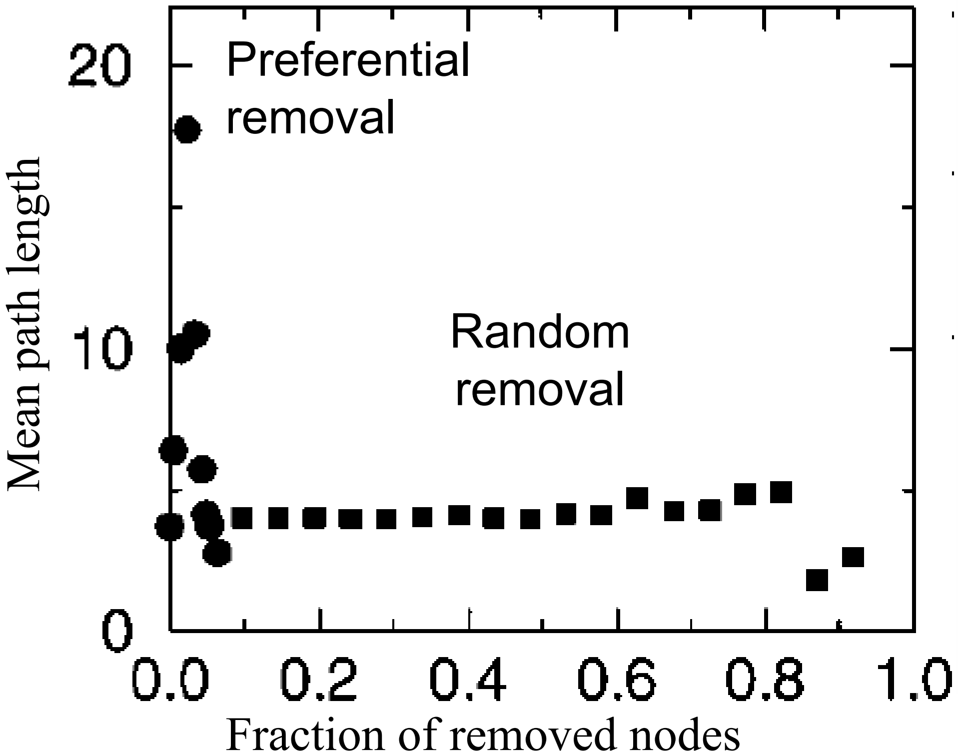
- We observe how the connectivity (length of the paths) of the network changes as the vertices get removed [Albert et al. 00; Palmer et al. 01]
- Vertices can be removed:
  - Uniformly at random
  - In order of decreasing degree
- It is important for epidemiology
  - Removal of vertices corresponds to vaccination



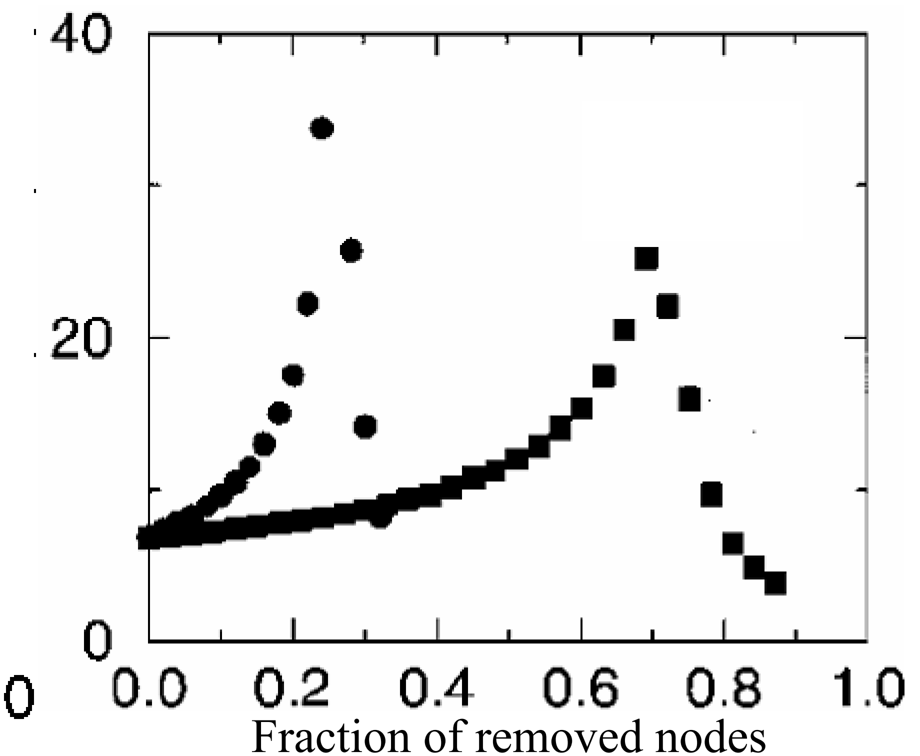
Network resilience (2)

- Real-world networks are resilient to random attacks
  - One has to remove all web-pages of degree > 5 to disconnect the web
  - But this is a very small percentage of web pages
- Random network has better resilience to targeted attacks

Internet (Autonomous systems)

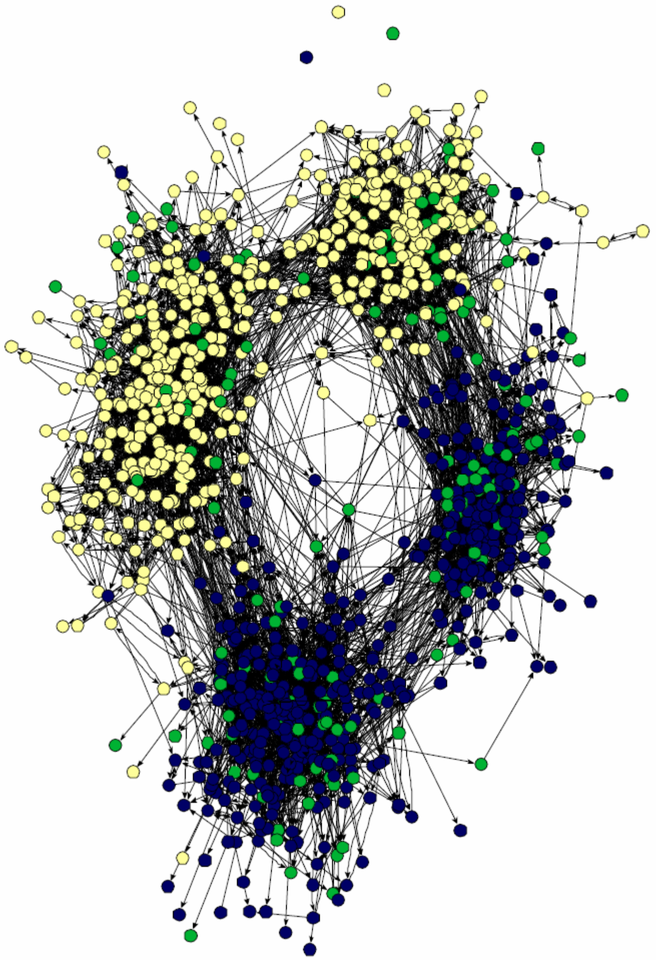


Random network



Community structure

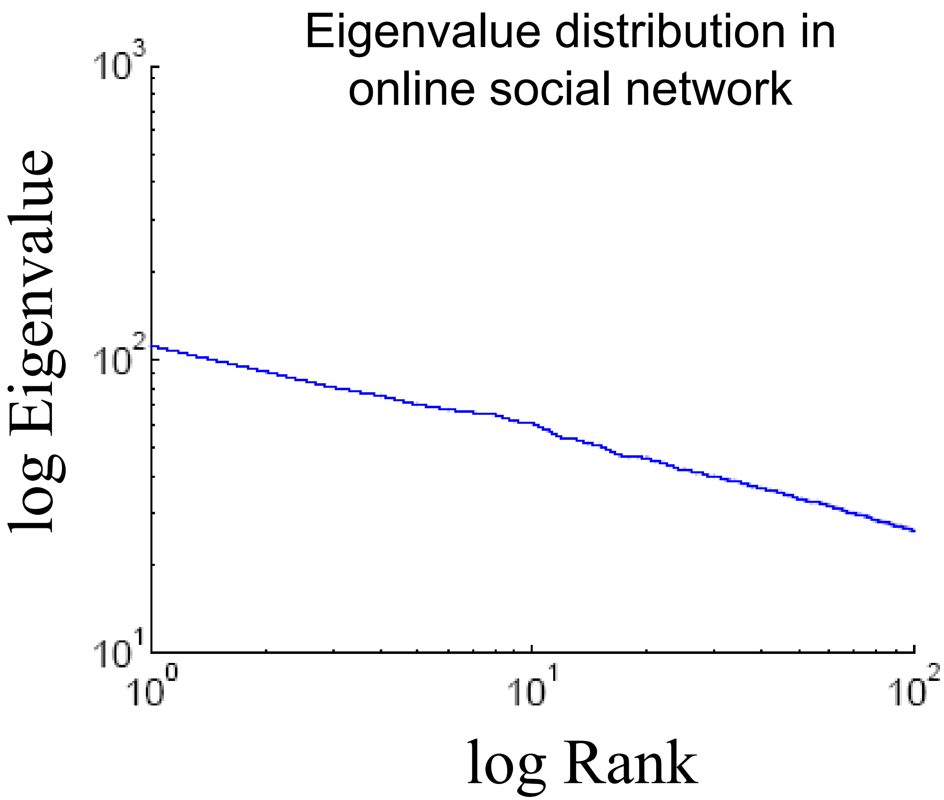
- Most social networks show community structure
  - groups have higher density of edges within than across groups
  - People naturally divide into groups based on interests, age, occupation, …
- How to find communities:
  - Spectral clustering (embedding into a low-dim space)
  - Hierarchical clustering based on connection strength
  - Combinatorial algorithms (min cut style formulations)
  - Block models
  - Diffusion methods



Friendship network of children in a school

Spectral properties

- Eigenvalues of graph adjacency matrix follow a power law
- Network values (components of principal eigenvector) also follow a power-law [Chakrabarti et al]



What about evolving graphs?

- Conventional wisdom/intuition:
  - Constant average degree: the number of edges grows linearly with the number of nodes
  - Slowly growing diameter: as the network grows the distances between nodes grow

Networks over time: Densification

- A simple question: What is the relation between the number of nodes and the number of edges in a network over time?
- Let:
  - $N(t)$ … nodes at time t
  - $E(t)$ … edges at time t
- Suppose that:

$$N(t+1) = 2 * N(t)$$

- Q: what is your guess for

$$E(t+1) =? \ 2 * E(t)$$

- A: over-doubled!
  - But obeying the Densification Power Law [KDD05]

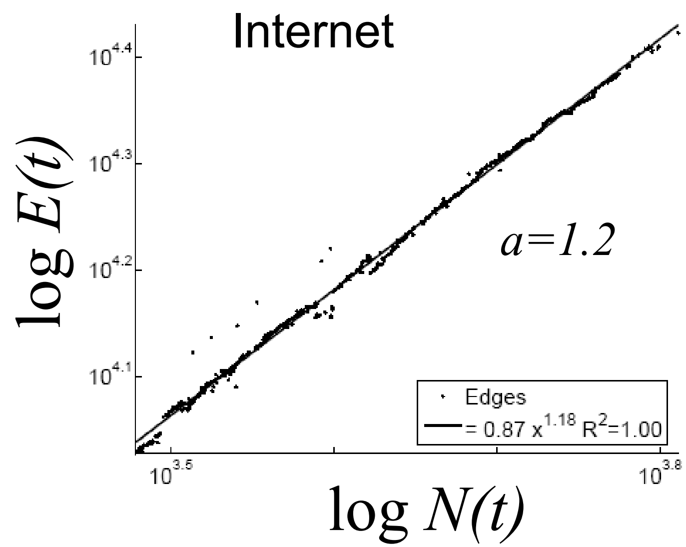
Networks over time: Densification

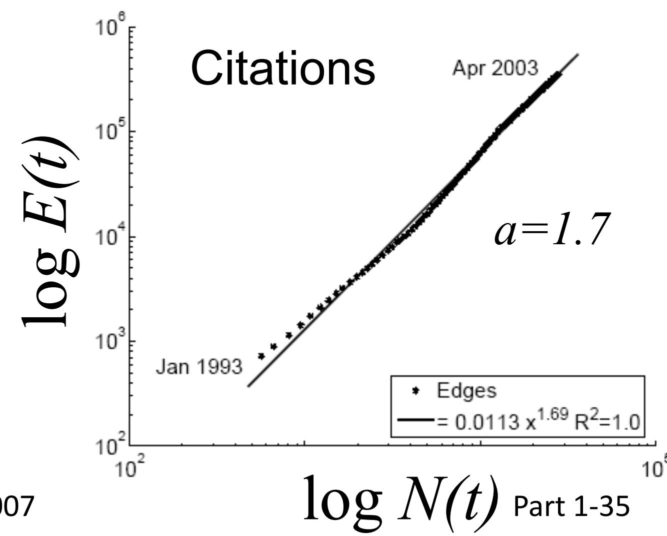
- Networks are denser over time
- The number of edges grows faster than the number of nodes – average degree is increasing

$$E(t) \propto N(t)^a$$

a … densification exponent

- $1 \leq a \leq 2$:
  - $a=1$: linear growth – constant out-degree (assumed in the literature so far)
  - $a=2$: quadratic growth – clique





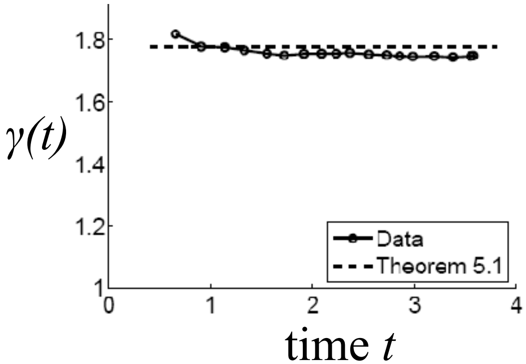
Densification & degree distribution

- How does densification affect degree distribution?
- Densification: $E(t) \propto N(t)^a$
- Degree distribution: $p_k = k^\gamma$
- Given densification exponent a, the degree exponent is [TKDD07]:
  - (a) For $\gamma = const$ over time, we obtain densification only for $1 < \gamma < 2$, and then it holds: $\gamma = a/2$
  - (b) For $\gamma < 2$ degree distribution evolves according to:

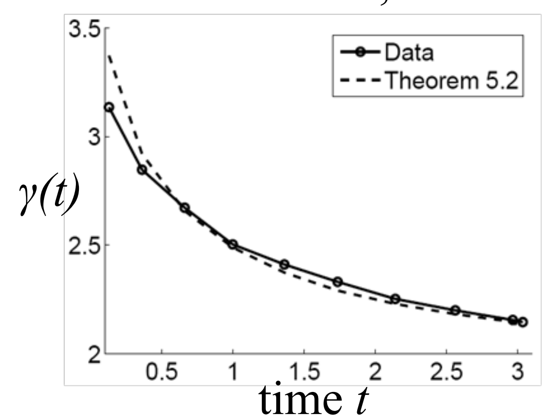
$$\gamma_n = \frac{4n^{a-1} - 1}{2n^{a-1} - 1}$$

Given: densification a, number of nodes n

Case (a): Degree exponent γ is constant over time. The network densifies, $a=1.2$

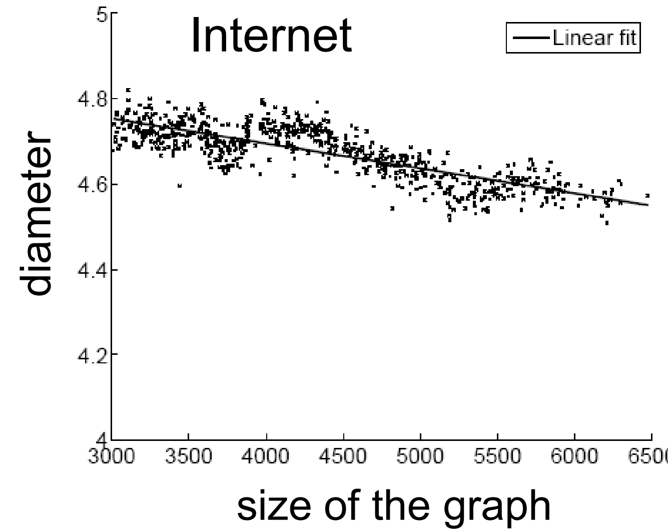


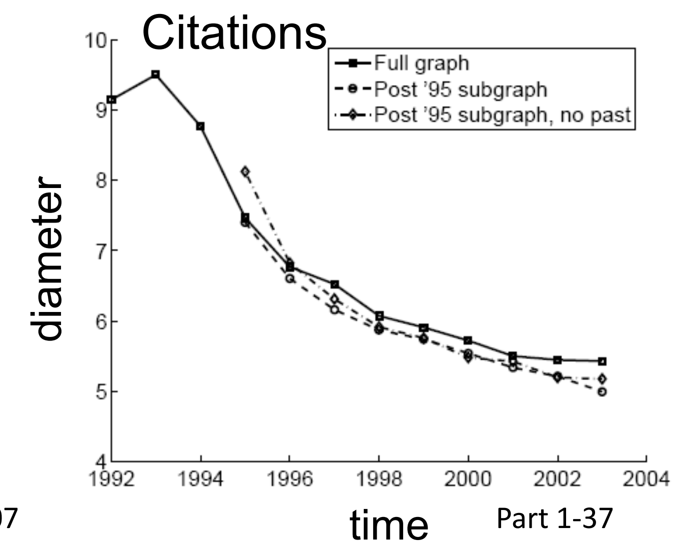
Case (b): Degree exponent γ evolves over time. The network densifies, $a=1.6$



Shrinking diameters

- Intuition and prior work say that distances between the nodes slowly grow as the network grows (like *log n*):
  - $d \sim O(\log N)$
  - $d \sim O(\log \log N)$
- Diameter Shrinks/Stabilizes over time
  - as the network grows the distances between nodes slowly decrease [KDD 05]





Properties hold in many graphs

- These patterns can be observed in many real world networks:
  - World wide web [Barabasi]
  - On-line communities [Holme, Edling, Liljeros]
  - Who call whom telephone networks [Cortes]
  - Internet backbone – routers [Faloutsos, Faloutsos, Faloutsos]
  - Movies to actors network [Barabasi]
  - Science citations [Leskovec, Kleinberg, Faloutsos]
  - Click-streams [Chakrabarti]
  - Autonomous systems [Faloutsos, Faloutsos, Faloutsos]
  - Co-authorship [Leskovec, Kleinberg, Faloutsos]
  - Sexual relationships [Liljeros]

Part 1.2: Models

We saw properties

How do we find models?

1.2 Models: Outline

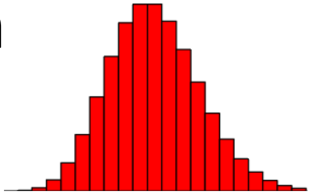
- The timeline of graph models:
  - (Erdos-Renyi) Random graphs (1960s)
  - Exponential random graphs
  - Small-world model
  - Preferential attachment
  - Edge copying model
  - Community guided attachment
  - Forest fire
  - Kronecker graphs (today)

(Erdos-Renyi) Random graph

- Also known as Poisson random graphs or Bernoulli graphs [Erdos&Renyi, 60s]
  - Given n vertices connect each pair i.i.d. with probability p
- Two variants:
  - $G_{n,p}$: graph with m edges appears with probability $p^m(1-p)^{M-m}$, where $M=0.5n(n-1)$ is the max number of edges
  - $G_{n,m}$: graphs with n nodes, *m* edges
- Does not mimic reality
- Very rich mathematical theory: many properties are exactly solvable

Properties of random graphs

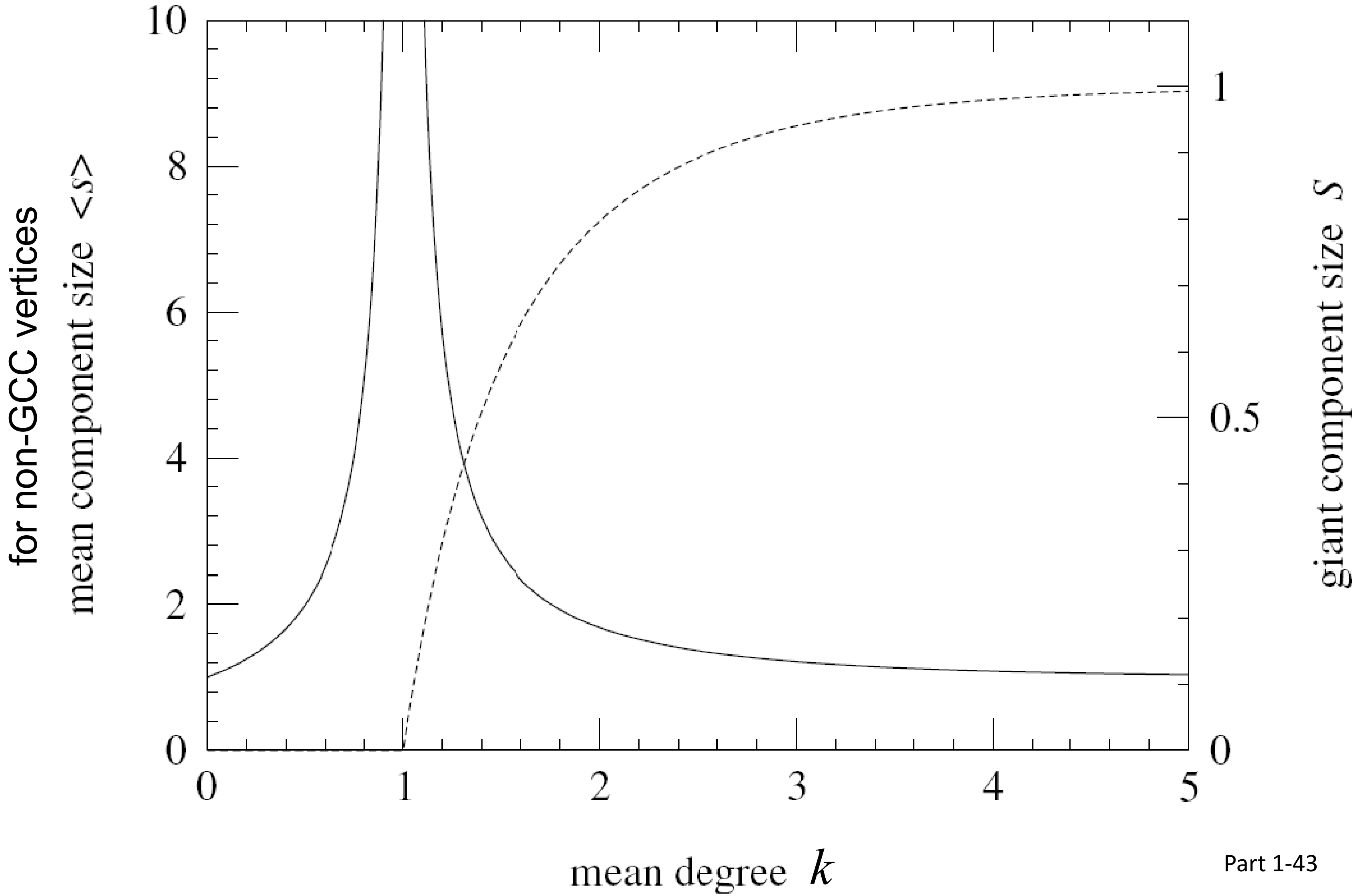
- Degree distribution is Poisson since the presen and absence of edges is independent



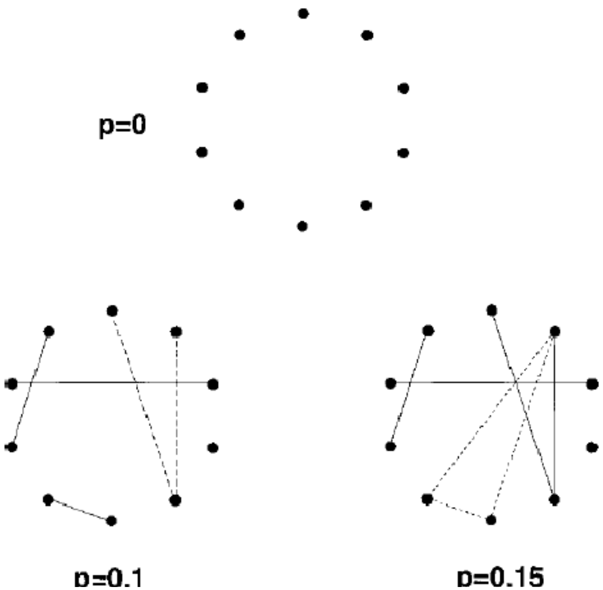
$$p_k = \binom{n}{k} p^k (1-p)^{n-k} \approx \frac{z^k e^{-z}}{k!}$$

- Giant component: average degree $k=2m/n$:
  - $k=1-\varepsilon$: all components are of size $\Omega(\log n)$
  - $k=1+\varepsilon$: there is 1 component of size $\Omega(n)$
    - All others are of size $\Omega(\log n)$
    - They are a tree plus an edge, i.e., cycles
- Diameter: $\log n / \log k$

Evolution of a random graph



Subgraphs in random graphs

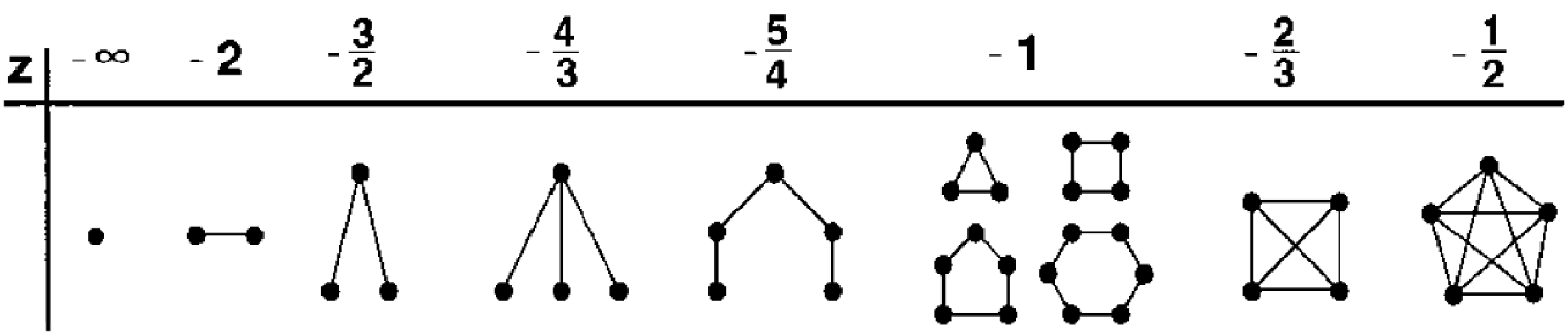


Expected number of subgraphs $H(v,e)$ in $G_{n,p}$ is

$$E(X) = \binom{n}{v} \frac{v!}{a} p^e \approx \frac{n^v p^e}{a}$$

a... # of isomorphic graphs

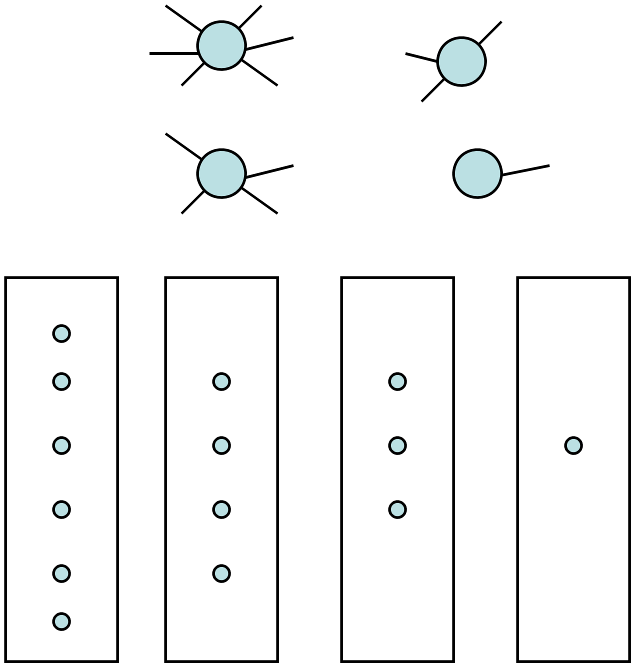
$p \sim N^z$



Random graphs: conclusion

- Pros:
  - Simple and tractable model
  - Phase transitions
  - Giant component
- Cons:
  - Degree distribution
  - No community structure
  - No degree correlations
- Extensions:
  - Configuration model
    - Random graphs with arbitrary degree sequence
    - Excess degree: Degree of a vertex of the end of random edge: $q_k = k\, p_k$

Configuration model

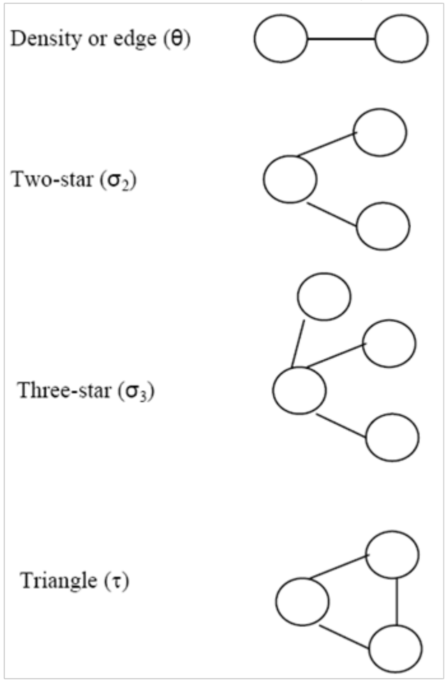


Exponential random graphs (p* models)

- Social sciences thoroughly analyze rather small networks
- Let ε_i set of properties of a graph:
  - E.g., number of edges, number of nodes of a given degree, number of triangles, …
- Exponential random graph model defines a probability distribution over graphs:

$$P(G) = \frac{1}{Z} \exp\left(-\sum_i \beta_i \epsilon_i\right)$$

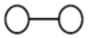
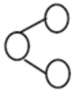
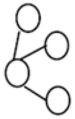
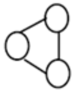
Examples of ε_i



Exponential random graphs

- Includes Erdos-Renyi as a special case
- Assume parameters β_i are specified
  - No analytical solutions for the model
  - But can use simulation to sample the graphs:
    - Define local moves on a graph:
      - Addition/removal of edges
      - Movement of edges
      - Edge swaps
- Parameter estimation:
  - maximum likelihood
- Problem:
  - Can't solve for transitivity (produces
  - Used to analyze small networks

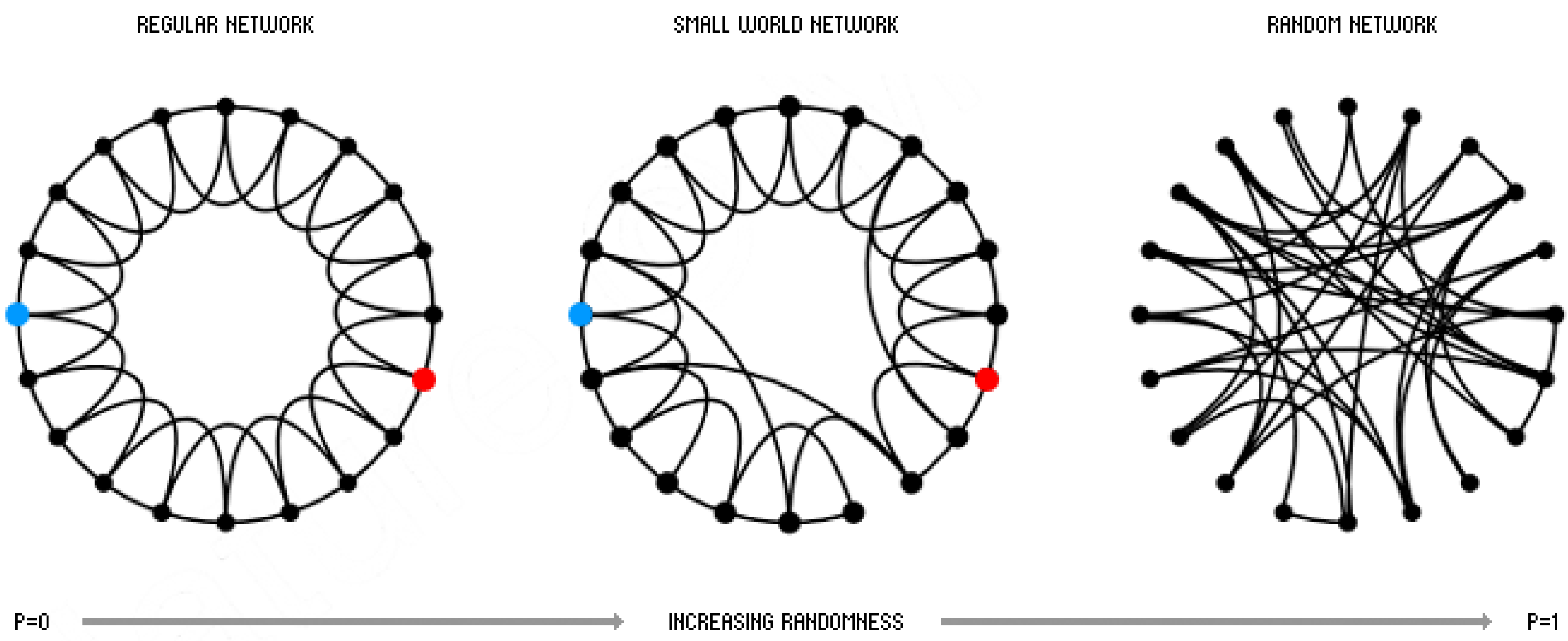
Example of parameter estimates:

Parameter	Configuration	Estimate (standard error)
θ		-4.27 (1.13)
σ_2		1.09 (0.65)
σ_3		-0.67 (0.41)
τ		1.32 (0.65)

Small-world model

- [Watts & Strogatz 1998]
- Used for modeling network transitivity
- Many networks assume some kind of geographical proximity
- Small-world model:
  - Start with a low-dimensional regular lattice
  - Rewire:
    - Add/remove edges to create shortcuts to join remote parts of the lattice
    - For each edge with prob p move the other end to a random vertex

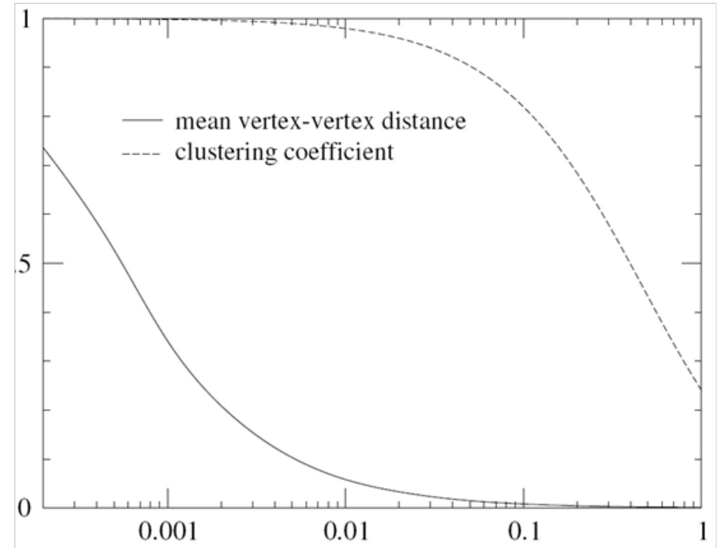
Small-world model



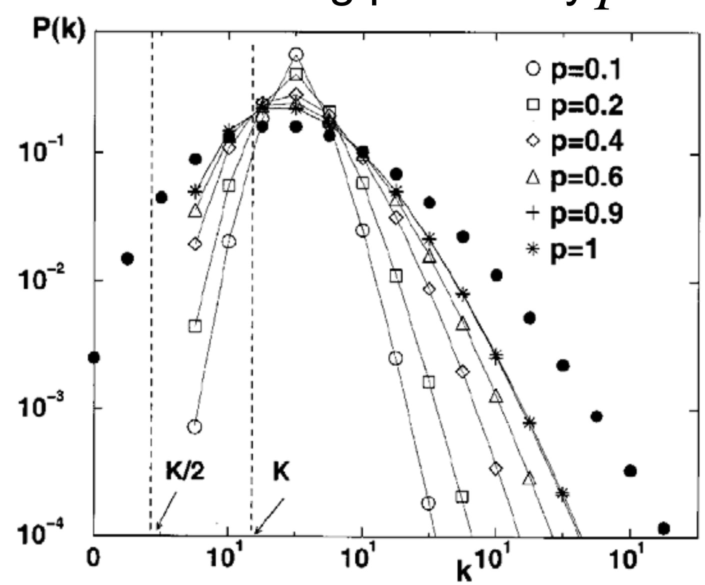
- Rewiring allows to interpolate between regular lattice and random graph

Small-world model

- Regular lattice ($p=0$):
  - Clustering coefficient
    $C=(3k-3)/(4k-2)=3/4$
  - Mean distance $L/4k$
- Almost random graph ($p=1$):
  - Clustering coefficient $C=2k/L$
  - Mean distance $log\ L\ /\ log\ k$
- But, real graphs have power-law degree distribution



Rewiring probability p



Degree distribution

Preferential attachment

- But, random graphs have Poisson degree distribution
- Let's find a better model

- Preferential attachment [Price 1965, Albert & Barabasi 1999]:
  - Add a new node, create m out-links
  - Probability of linking a node k_i is proportional to its degree

$$\Pi(k_i) = \frac{k_i}{\sum_j k_j}$$

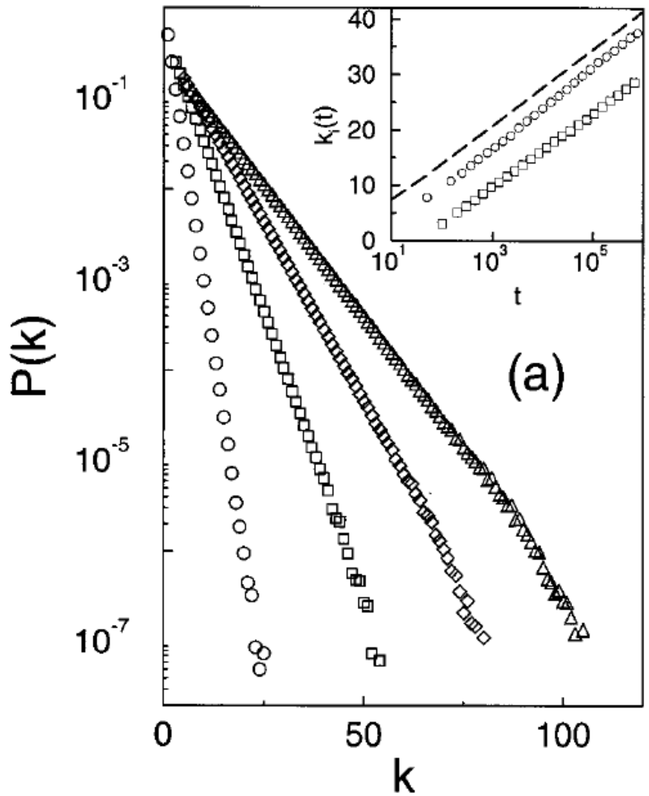
- Based on Herbert Simon's result
  - Power-laws arise from "Rich get richer" (cumulative advantage)
- Examples (Price 1965 for modeling citations):
  - Citations: new citations of a paper are proportional to the number it already has

Preferential attachment

- Leads to power-law degree distributions

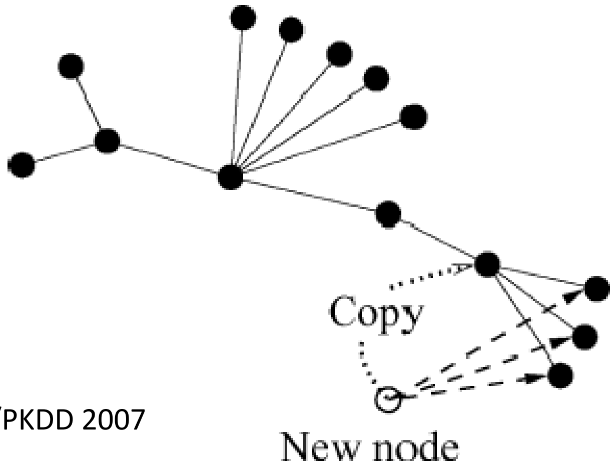
$$p_k \propto k^{-3}$$

- But:
  - all nodes have equal (constant) out-degree
  - one needs a complete knowledge of the network
- There are many generalizations and variants, but the preferential selection is the key ingredient that leads to power-laws



Edge copying model

- But, preferential attachment does not have communities
- Copying model [Kleinberg et al, 99]:
  - Add a node and choose k the number of edges to add
  - With prob. β select k random vertices and link to them
  - Prob. $1\text{-}\beta$ edges are copied from a randomly chosen node
- Generates power-law degree distributions with exponent $1/(1\text{-}\beta)$
- Generates communities

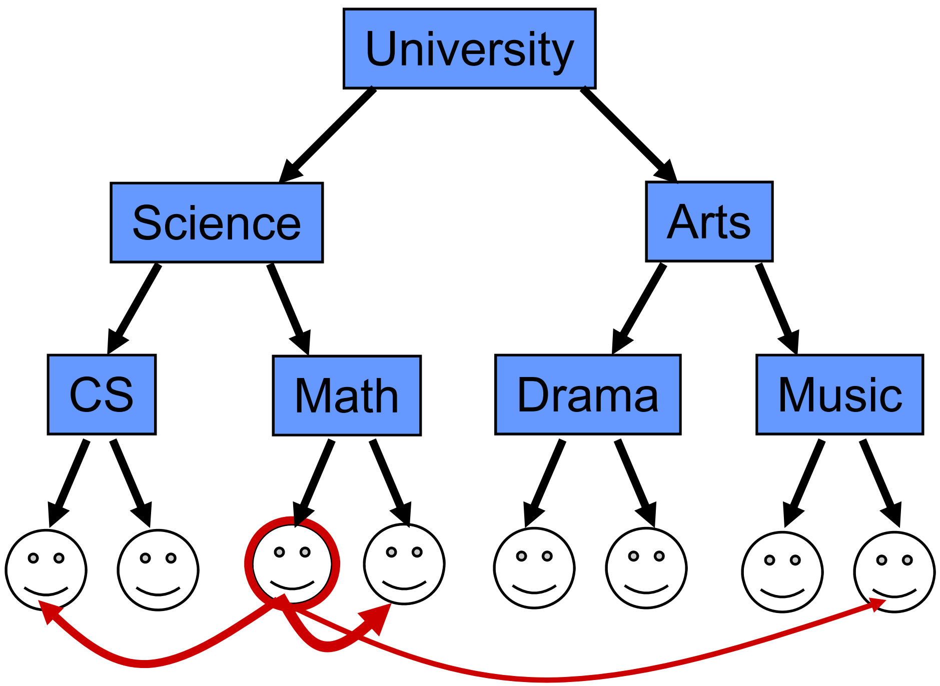


Community guided attachment

- But, we want to model densification in networks

$$E(t) \propto N(t)^a$$

- Assume community structure
- One expects many within-group friendships and fewer cross-group ones



- Community guided attachment [KDD05]

Self-similar university community structure

Community guided attachment

- Assuming cross-community linking probability

$$f(h) = c^{-h}$$

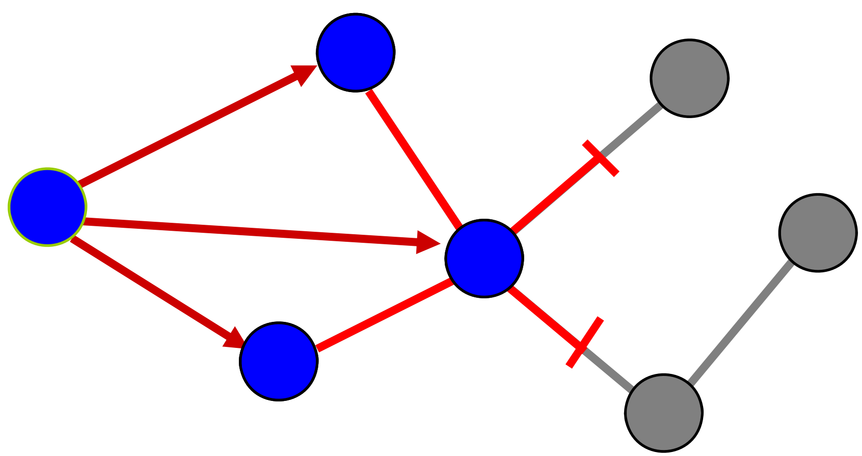
- The Community Guided Attachment leads to Densification Power Law with exponent

$$a = 2 - \log_b(c)$$

  - a … densification exponent
  - b … community tree branching factor
  - c … difficulty constant, $1 \leq c \leq b$
- If $c = 1$: easy to cross communities
  - Then: $a=2$, quadratic growth of edges – near clique
- If $c = b$: hard to cross communities
  - Then: $a=1$, linear growth of edges – constant out-degree

Forest Fire Model

- But, we do not want to have explicit communities
- Want to model graphs that density and have shrinking diameters
- Intuition:
  - How do we meet friends at a party?
  - How do we identify references when writing papers?

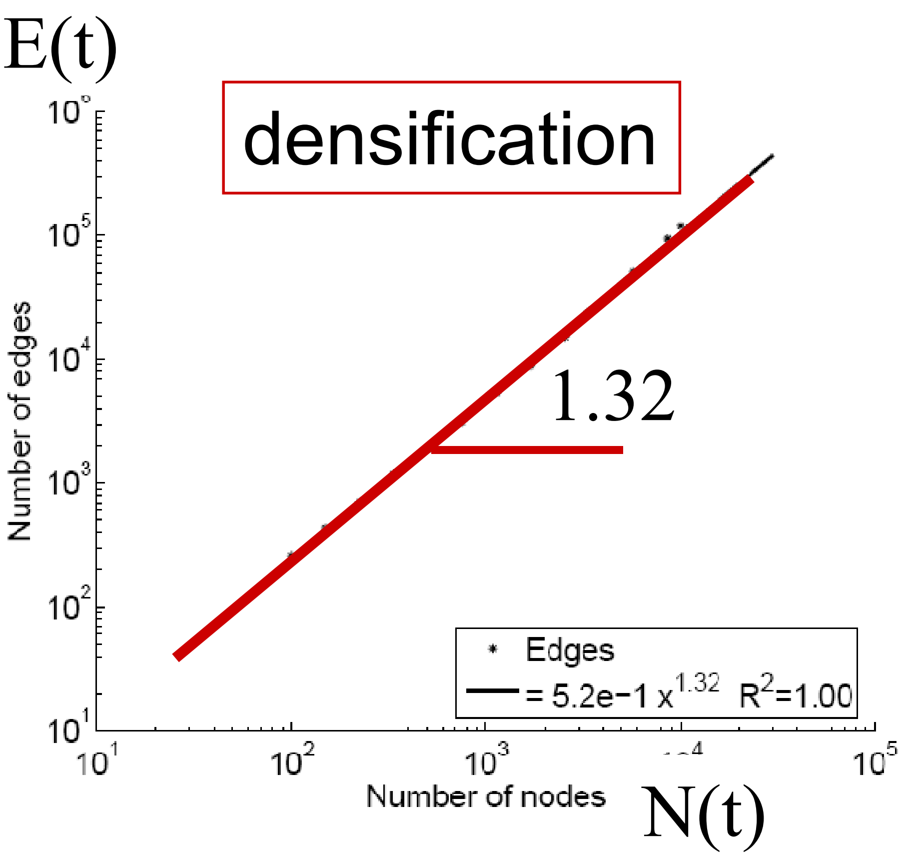


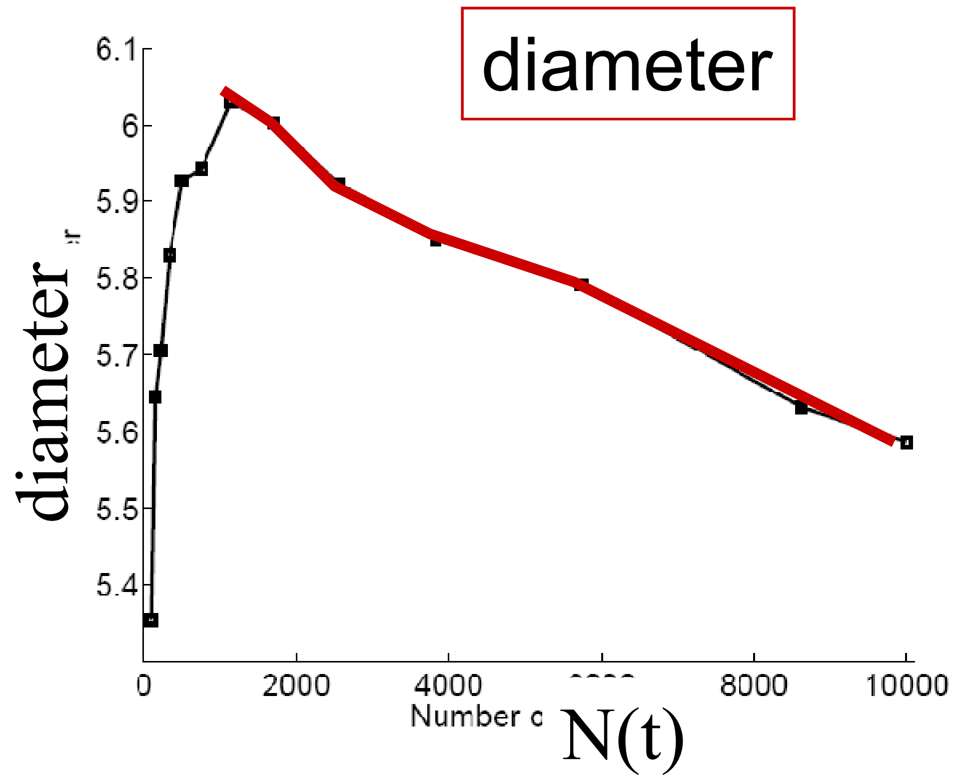
Forest Fire Model

- The Forest Fire model [KDD05] has 2 parameters:
  - p … forward burning probability
  - r … backward burning probability
- The model:
  - Each turn a new node v arrives
  - Uniformly at random chooses an "ambassador" w
  - Flip two geometric coins to determine the number in- and out-links of w to follow (burn)
  - Fire spreads recursively until it dies
  - Node v links to all burned nodes

Forest Fire Model

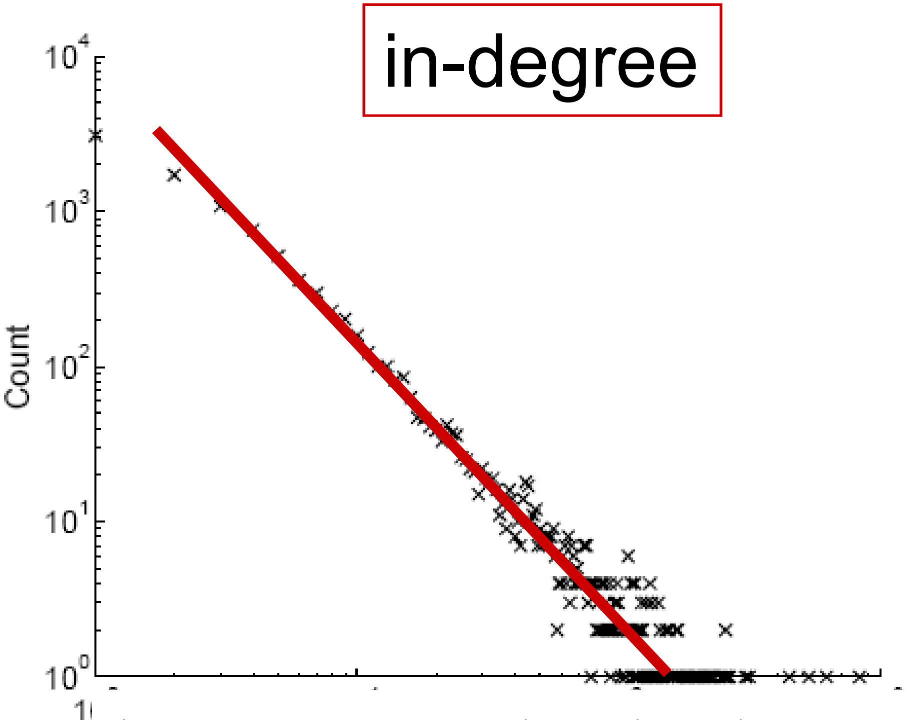
- Forest Fire generates graphs that densify and have shrinking diameter



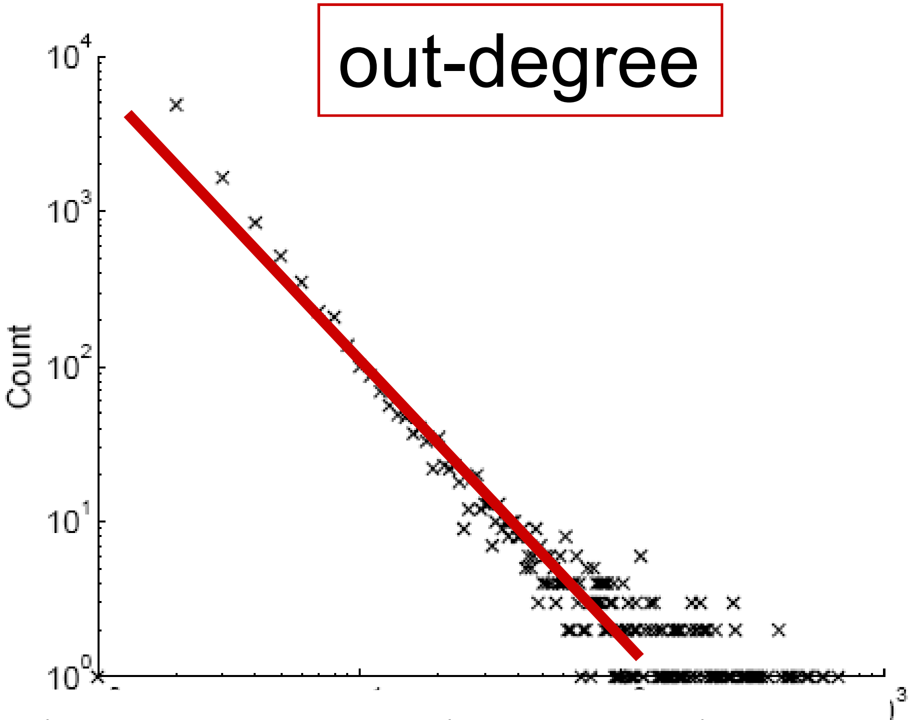


Forest Fire Model

- Forest Fire also generates graphs with Power-Law degree distribution



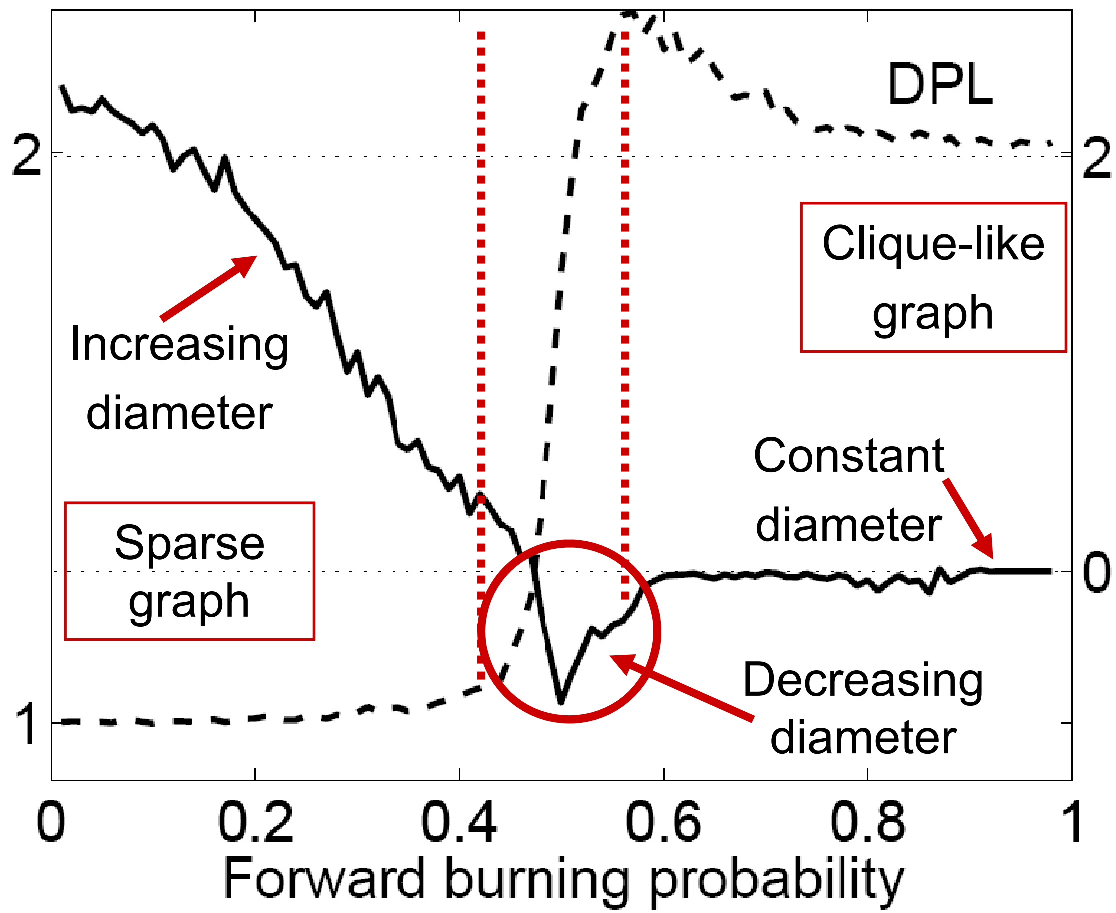
log count vs. log in-degree



log count vs. log out-degree

Forest Fire: Phase transitions

- Fix backward probability r and vary forward burning probability p
- We observe a sharp transition between sparse and clique-like graphs
- Sweet spot is very narrow

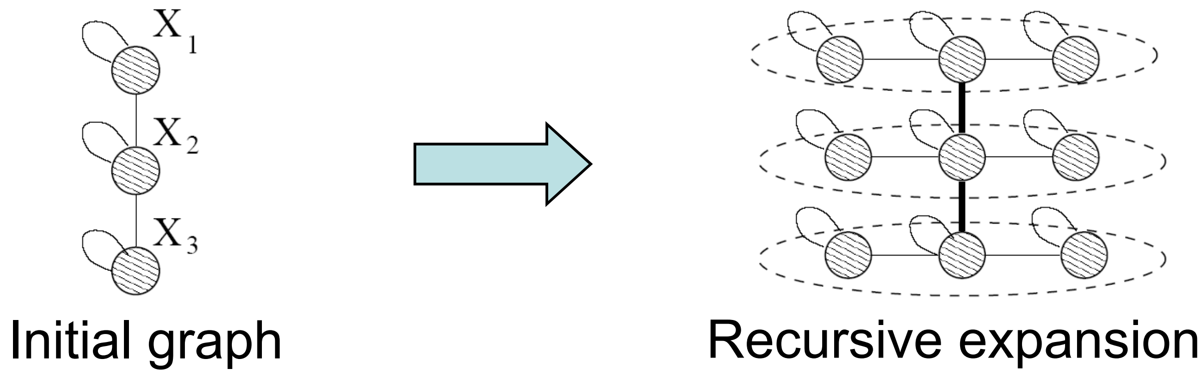


Kronecker graphs

- But, want to have a model that can generate a realistic graph with realistic growth:
  - Static Patterns
    - Power Law Degree Distribution
    - Small Diameter
    - Power Law Eigenvalue and Eigenvector Distribution
  - Temporal Patterns
    - Densification Power Law
    - Shrinking/Constant Diameter
- For Kronecker graphs [PKDD05] all these properties can actually be proven

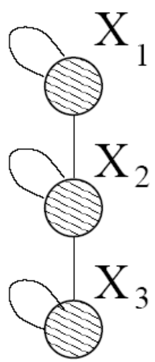
Idea: Recursive graph generation

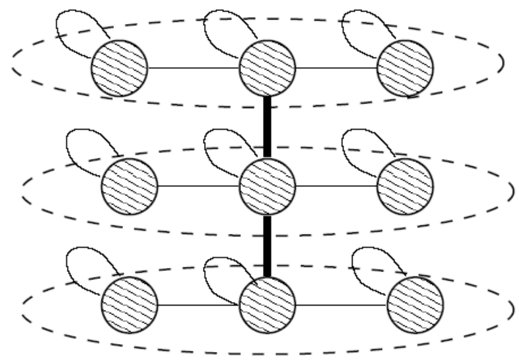
- Starting with our intuitions from densification
- Try to mimic recursive graph/community growth because self similarity leads to power-laws
- There are many obvious (but wrong) ways:



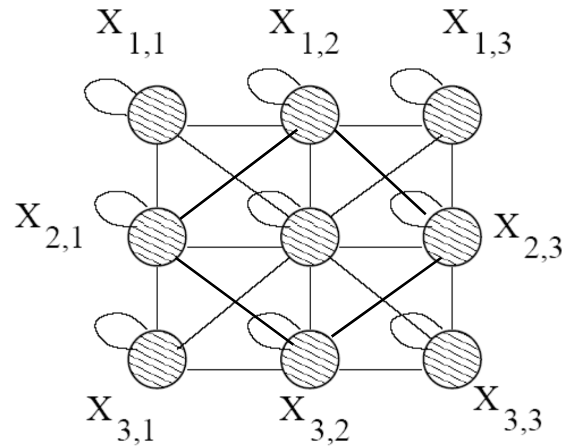
  - Does not densify, has increasing diameter
- Kronecker Product is a way of generating self-similar matrices

Kronecker product: Graph





Intermediate stage



1	1	0
1	1	1
0	1	1

(3x3)

G_1

Adjacency matrix

G_1	G_1	0
G_1	G_1	G_1
0	G_1	G_1

(9x9)

$G_2 = G_1 \otimes G_1$

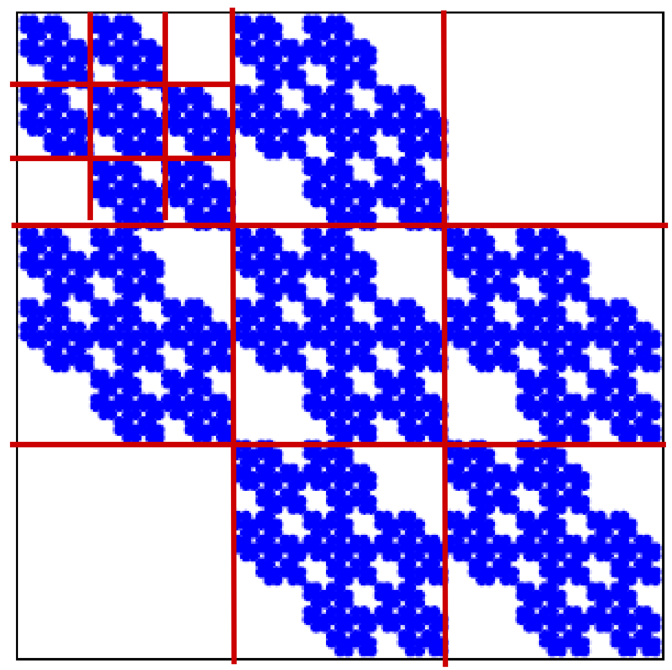
Adjacency matrix

Kronecker product: Graph

- Continuing multypling with G_1 we obtain G_4 and so on …

1	1	0
1	1	1
0	1	1

G_1



G_4 adjacency matrix

Kronecker product: Definition

- The Kronecker product of matrices A and B is given by

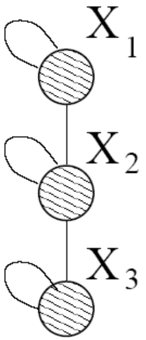
$$\mathbf{C} = \underset{N\,x\,M}{\mathbf{A}} \otimes \underset{K\,x\,L}{\mathbf{B}} \doteq \underbrace{\begin{pmatrix} a_{1,1}\mathbf{B} & a_{1,2}\mathbf{B} & \dots & a_{1,m}\mathbf{B} \\ a_{2,1}\mathbf{B} & a_{2,2}\mathbf{B} & \dots & a_{2,m}\mathbf{B} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n,1}\mathbf{B} & a_{n,2}\mathbf{B} & \dots & a_{n,m}\mathbf{B} \end{pmatrix}}_{N*K\,x\,M*L}$$

- We define a Kronecker product of two graphs as a Kronecker product of their adjacency matrices

Kronecker graphs

- We propose a growing sequence of graphs by iterating the Kronecker product

$$G_k = \underbrace{G_1 \otimes G_1 \otimes \ldots G_1}_{k\ times}$$



G_1

- Each Kronecker multiplication exponentially increases the size of the graph
- G_k has N_1^k nodes and E_1^k edges, so we get densification

Stochastic Kronecker graphs

- But, want a randomized version of Kronecker graphs
- Possible strategies:
  - Randomly add/delete some edges
  - Threshold the matrix, e.g. use only the strongest edges
- Wrong, will destroy the structure of the graph, e.g. diameter, clustering

Stochastic Kronecker graphs

- Create $N_1 \times N_1$ probability matrix P_1
- Compute the k^{th} Kronecker power P_k
- For each entry p_{uv} of P_k include an edge (u,v) with probability p_{uv}

0.5	0.2
0.1	0.3

P_1

Kronecker multiplication

Probability of edge p_{ij}

0.25	0.10	0.10	0.04
0.05	0.15	0.02	0.06
0.05	0.02	0.15	0.06
0.01	0.03	0.03	0.09

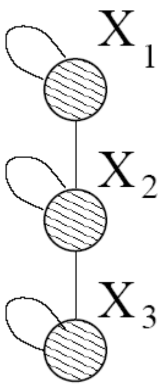
$P_2 = P_1 \otimes P_1$

Instance matrix K_2

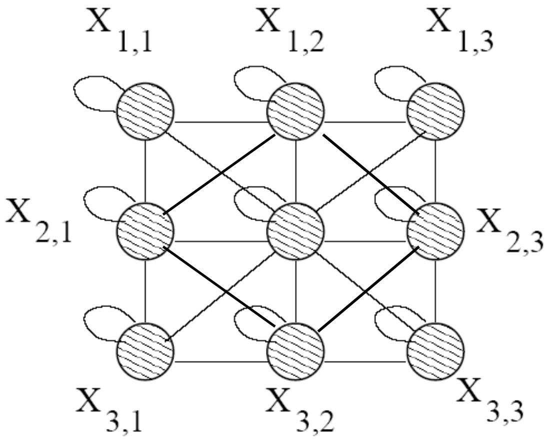
flip biased coins

Kronecker graphs: Intuition (1)

- Intuition:
  - Recursive growth of graph communities
  - Nodes get expanded to micro communities
  - Nodes in sub-community link among themselves and to nodes from different communities



G_1



$G_2 = G_1 \otimes G_1$

Kronecker graphs: Intuition (2)

- Node attribute representation
  - Nodes are described by (binary) features [likes ice cream, likes chocolate]
  - *E.g.*, u=[1,0],   v=[1, 1]
  - Parameter matrix gives linking probability:
    p(u,v) = 0.1 * 0.5 = 0.15

	1	0
1	0.5	0.2
0	0.1	0.3

Kronecker multiplication →

	11	10	01	00
11	0.25	0.10	0.10	0.04
10	0.05	0.15	0.02	0.06
01	0.05	0.02	0.15	0.06
00	0.01	0.03	0.03	0.09

Properties of Kronecker graphs

- We can show [PKDD05] that Kronecker multiplication generates graphs that have:
  - Properties of static networks
    - ✓ Power Law Degree Distribution
    - ✓ Power Law eigenvalue and eigenvector distribution
    - ✓ Small Diameter
  - Properties of dynamic networks
    - ✓ Densification Power Law
    - ✓ Shrinking/Stabilizing Diameter

Mahdian and Xu '07 show that these properties also hold for Stochastic Kronecker Graphs

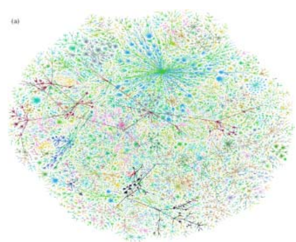
1.3: Fitting the models to real graphs

We saw the models.

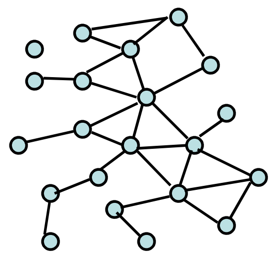
Want to fit a model to a large real graph?

The problem

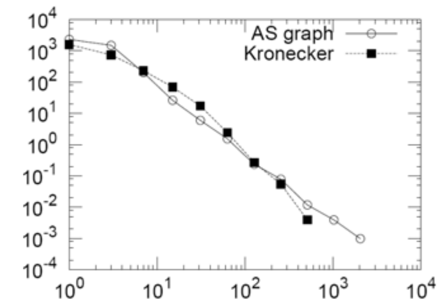
- We want to generate realistic networks:



Given a
real network



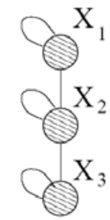
Generate a
synthetic network



Some statistical property,
e.g., degree distribution

  - P1) What are the relevant properties?
  - P2) What is a good analytically tractable model?
  - P3) How can we fit the model (find parameters)?

Model estimation: approach



- Maximum likelihood estimation
  - Given real graph G
  - Estimate Kronecker initiator graph Θ (e.g., $\begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 1 \end{bmatrix}$) which

$$\arg\max_{\Theta} P(G \mid \Theta)$$

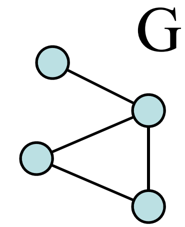
- We need to (efficiently) calculate

$$P(G \mid \Theta)$$

- And maximize over Θ (e.g., using gradient descent)

Fitting Kronecker graphs

- Given a graph G and Kronecker matrix Θ we calculate probability that Θ generated G $P(G|\Theta)$



0.5	0.2
0.1	0.3

Θ

0.25	0.10	0.10	0.04
0.05	0.15	0.02	0.06
0.05	0.02	0.15	0.06
0.01	0.03	0.03	0.09

Θ_k

1	0	1	1
0	1	0	1
1	0	1	1
1	1	1	1

G

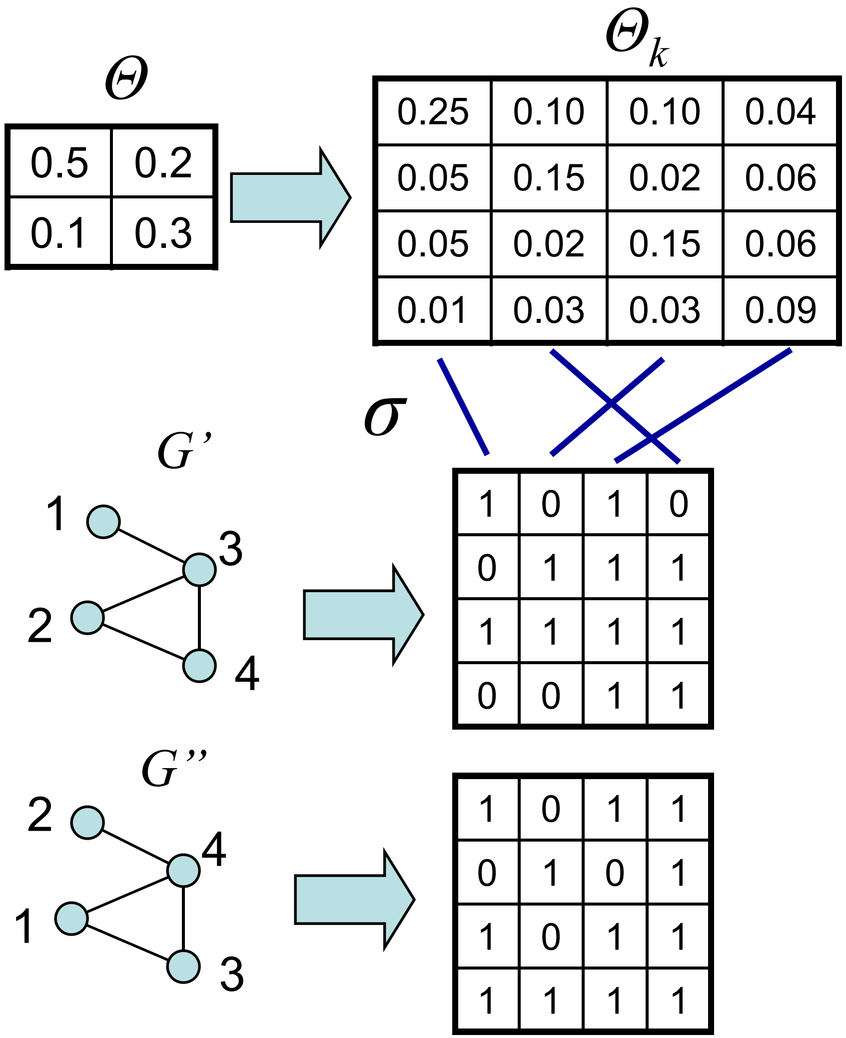
$\boldsymbol{P}(G|\Theta)$

$$P(G \mid \Theta) = \prod_{(u,v)\in G} \Theta_k[u,v] \prod_{(u,v)\notin G} (1-\Theta_k[u,v])$$

Challenges

- Challenge 1: Node correspondence problem
  - How the map the nodes of the real graph to the nodes of the synthetic graph?
- Challenge 2: Scalability
  - For large graphs $O(N^2)$ is too slow
  - Scaling to large graphs – performing the calculations quickly

Challenge 1: Node correspondence



$P(G'|\Theta) = P(G''|\Theta)$

- Nodes are unlabeled
- Graphs G' and G'' should have the same probability

  $P(G'|\Theta) = P(G''|\Theta)$

- One needs to consider all node correspondences σ

$$P(G \mid \Theta) = \sum_{\sigma} P(G \mid \Theta, \sigma) P(\sigma)$$

- All correspondences are a priori equally likely
- There are ***O(N!)*** correspondences

Challenge 2: calculating $P(G|\Theta,\sigma)$

- Assume we solved the correspondence problem
- Calculating

$$P(G|\Theta) = \prod_{(u,v)\in G} \Theta_k[\sigma_u, \sigma_v] \prod_{(u,v)\notin G} (1-\Theta_k[\sigma_u, \sigma_v])$$

σ… node labeling

- Takes $O(N^2)$ time
- Infeasible for large graphs (N ~ 10^5)

0.25	0.10	0.10	0.04
0.05	0.15	0.02	0.06
0.05	0.02	0.15	0.06
0.01	0.03	0.03	0.09

Θ_{kc}

σ

1	0	1	1
0	1	0	1
1	0	1	1
0	0	1	1

G

$P(G|\Theta, \sigma)$

Model estimation: solution

- Naïvely estimating the Kronecker initiator takes $O(N!N^2)$ time:
  - $N!$ for graph isomorphism
    - Metropolis sampling: $N! \rightarrow const$
  - N^2 for traversing the graph adjacency matrix
    - Properties of Kronecker product and sparsity ($E << N^2$): $N^2 \rightarrow E$
- We can estimate the parameters of Kronecker graph in linear time $O(E)$
- For details see [Leskovec-Faloutsos 2007]

Solution 1: Node correspondence

- Log-likelihood

$$l(\Theta) = \log \sum_{\sigma} P(G|\Theta,\sigma)P(\sigma)$$

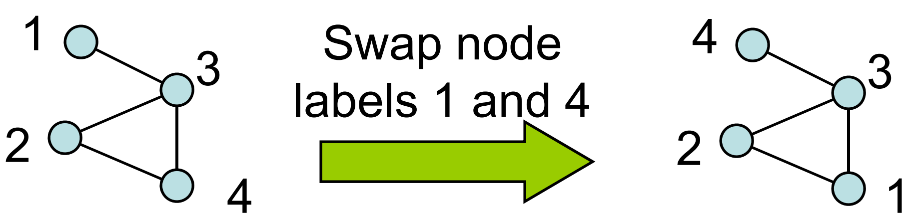
- Gradient of log-likelihood

$$\frac{\partial}{\partial\Theta} l(\Theta) = \sum_{\sigma} \frac{\partial \log P(G|\sigma,\Theta)}{\partial\Theta} \boxed{P(\sigma|G,\Theta)}$$

- Sample the permutations from $P(\sigma|G,\Theta)$ and average the gradients

Sampling node correspondences

- Metropolis sampling:
  - Start with a random permutation
  - Do local moves on the permutation
  - Accept the new permutation
    - If new permutation is better (gives higher likelihood)
    - If new is worse accept with probability proportional to the ratio of likelihoods



1	1	0	1	0
2	0	1	1	1
3	1	1	1	1
4	0	1	1	1

1	1	1	1	0
2	1	1	1	0
3	1	1	1	1
4	0	0	1	1

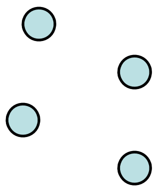
Can compute efficiently: Only need to account for changes in 2 rows / columns

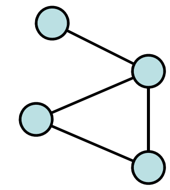
Solution 2: Calculating $P(G|\Theta,\sigma)$

- Calculating naively $P(G|\Theta,\sigma)$ takes $O(N^2)$
- Idea:
  - First calculate likelihood of empty graph, a graph with 0 edges
  - Correct the likelihood for edges that we observe in the graph
- By exploiting the structure of Kronecker product we obtain closed form for likelihood of an empty graph

Solution 2: Calculating *P(G|Θ,σ)*

- We approximate the likelihood:





$$l(\Theta) \approx \underbrace{l_e(\Theta)}_{\text{Empty graph}} + \sum_{(u,v)\in G} \underbrace{-\log(1-\Theta_k[\sigma_u,\sigma_v])}_{\text{No-edge likelihood}} + \underbrace{\log(\Theta_k[\sigma_u,\sigma_v])}_{\text{Edge likelihood}}$$

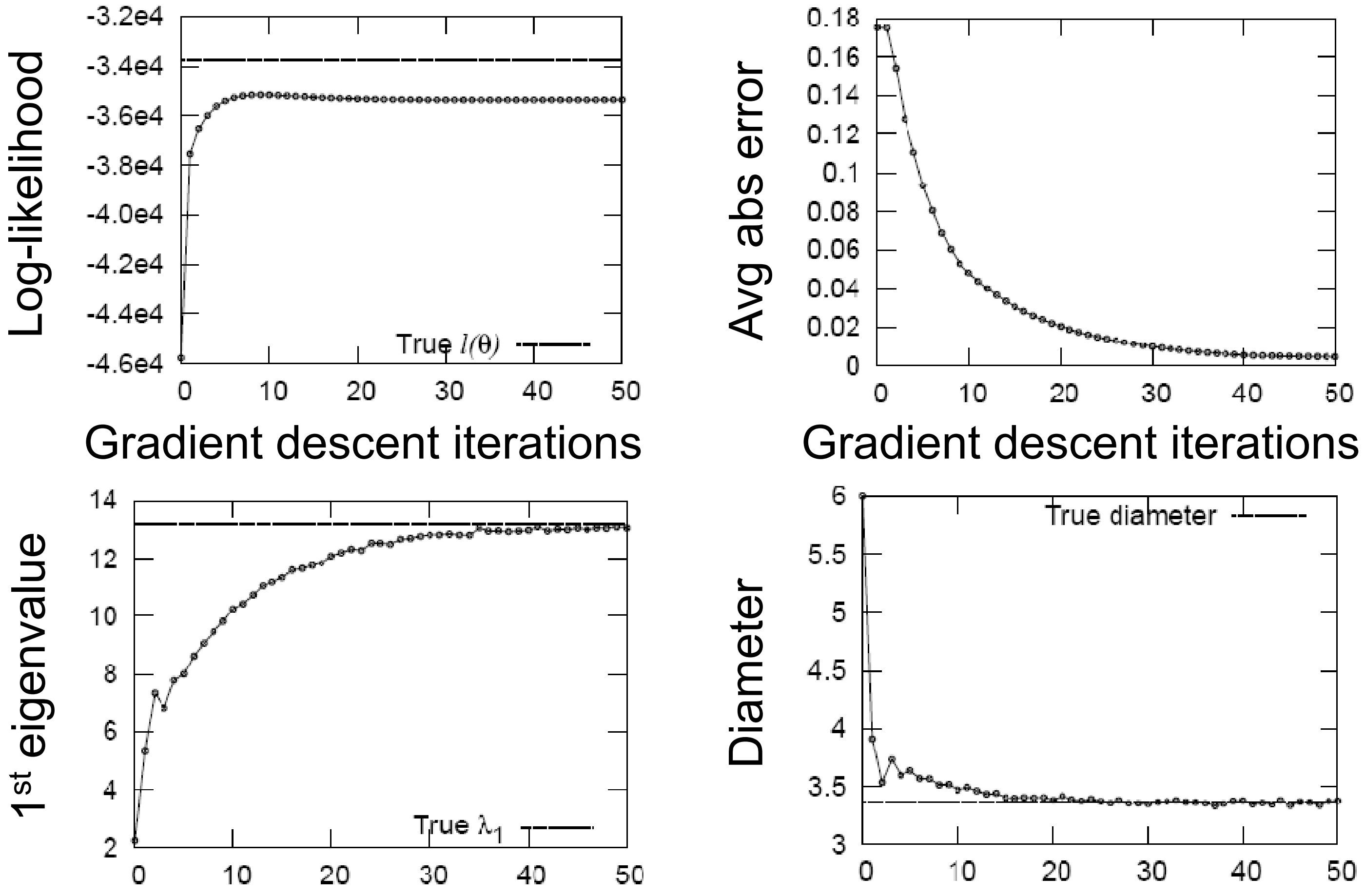
- The sum goes only over the edges
- Evaluating $P(G|\Theta,\sigma)$ takes $O(E)$ time
- Real graphs are sparse, $E << N^2$

Experiments: Synthetic data

- Can gradient descent recover true parameters?
- Optimization problem is not convex
- How nice (without local minima) is optimization space?
  - Generate a graph from random parameters
  - Start at random point and use gradient descent
  - We recover true parameters 98% of the times

Convergence of properties

- How does algorithm converge to true parameters with gradient descent iterations?



Experiments: real networks

- Experimental setup:
  - Given real graph
  - Stochastic gradient descent from random initial point
  - Obtain estimated parameters
  - Generate synthetic graphs
  - Compare properties of both graphs
- We do not fit the properties themselves
- We fit the likelihood and then compare the graph properties

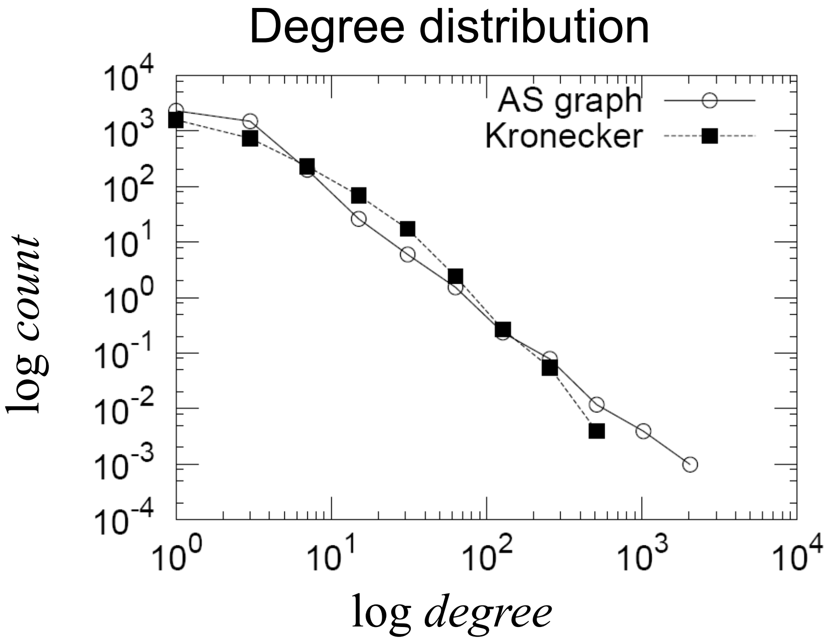
AS graph (N=6500, E=26500)

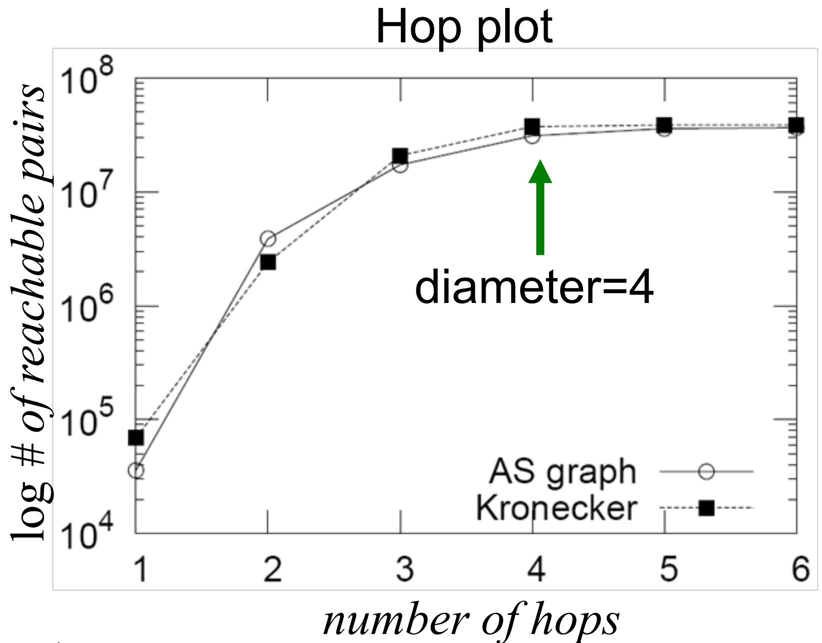
- Autonomous systems (internet)
- We search the space of ~$10^{50,000}$ permutations
- Fitting takes 20 minutes
- AS graph is undirected and estimated parameter matrix is symmetric:

0.98	0.58
0.58	0.06

AS: comparing graph properties

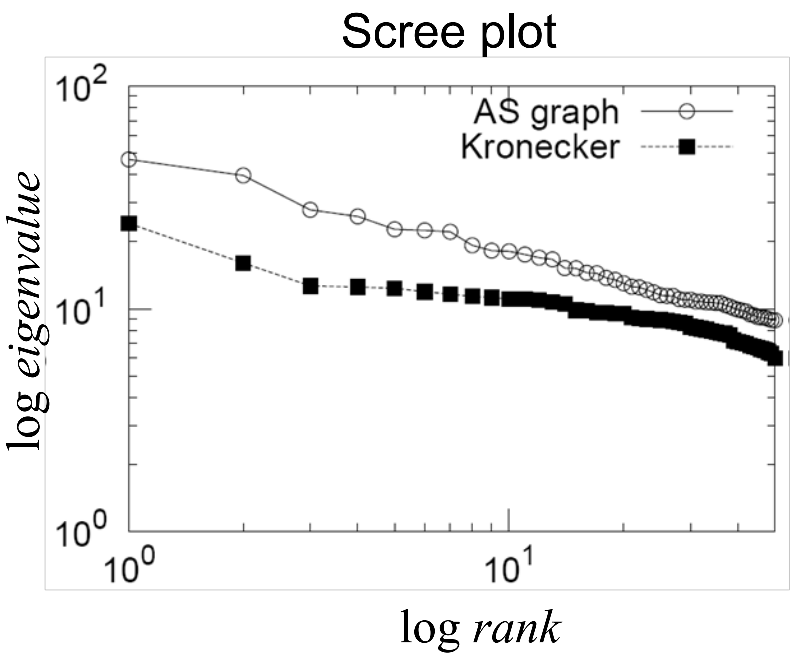
- Generate synthetic graph using estimated parameters
- Compare the properties of two graphs

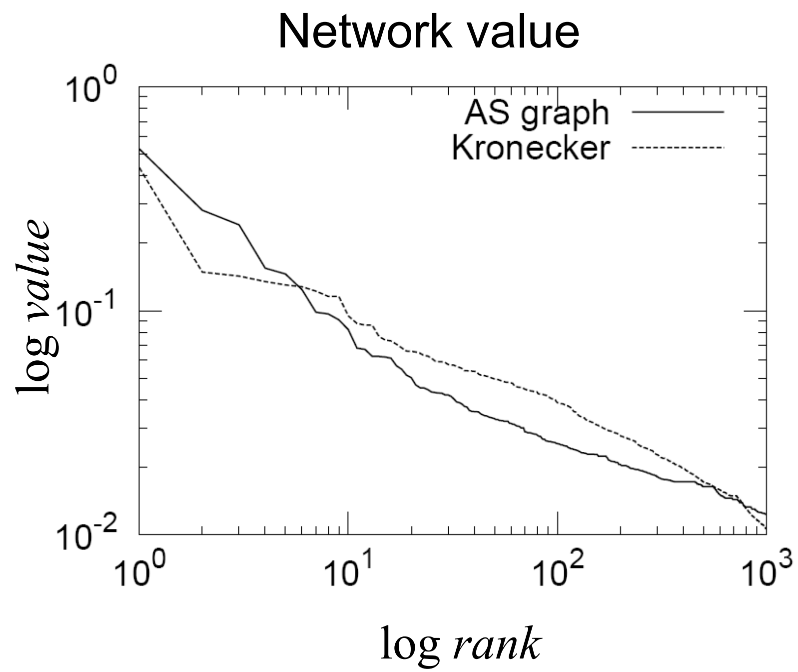




AS: comparing graph properties

- Spectral properties of graph adjacency matrices

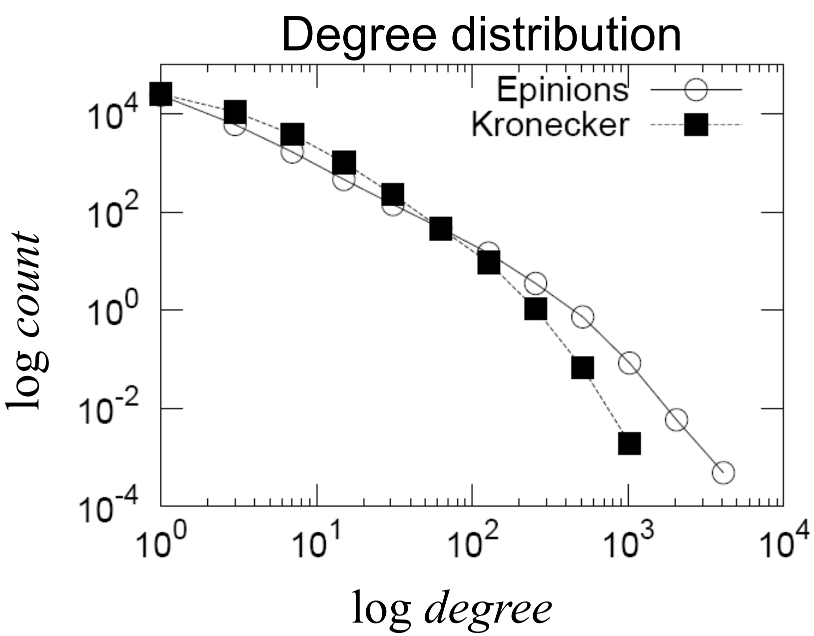


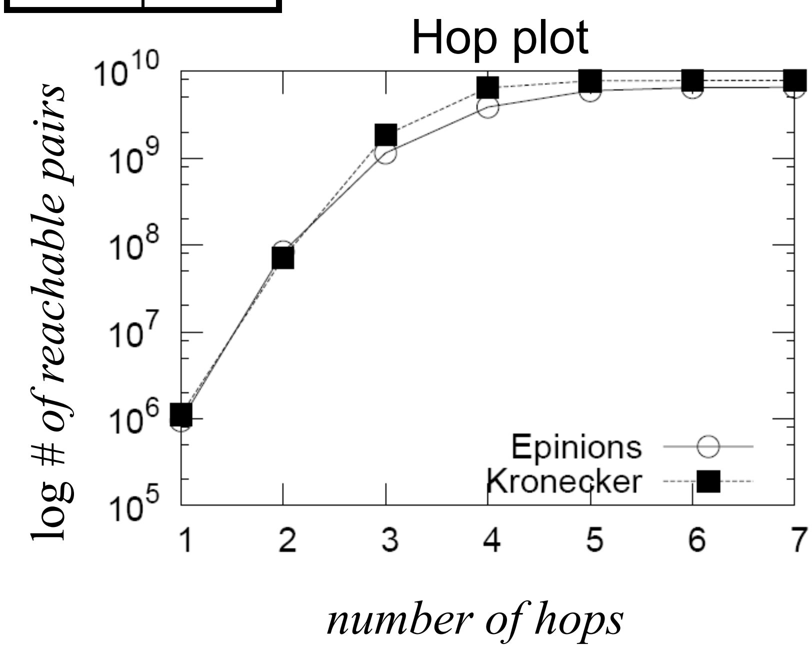


Epinions graph (N=76k, E=510k)

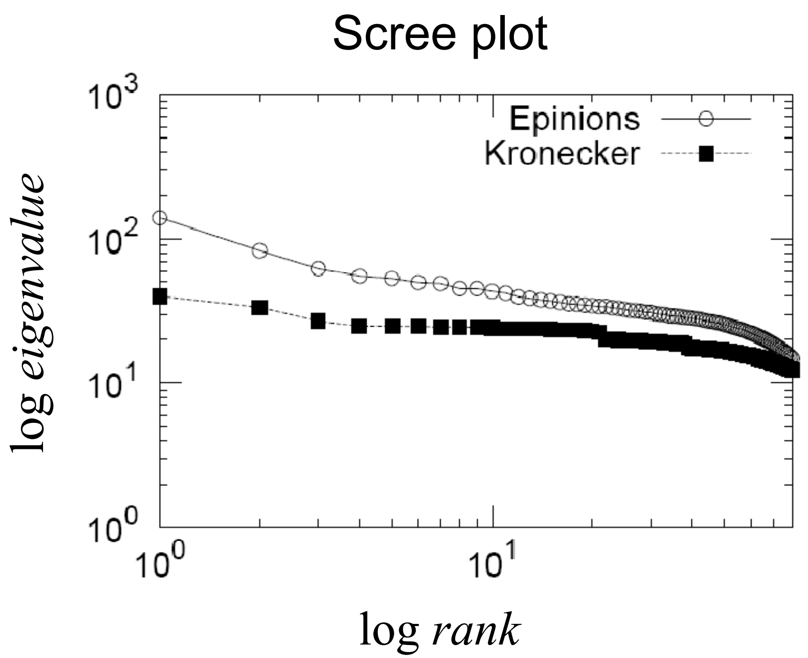
- We search the space of ~$10^{1,000,000}$ permutations
- Fitting takes 2 hours
- The structure of the estimated parameter gives insight into the structure of the graph

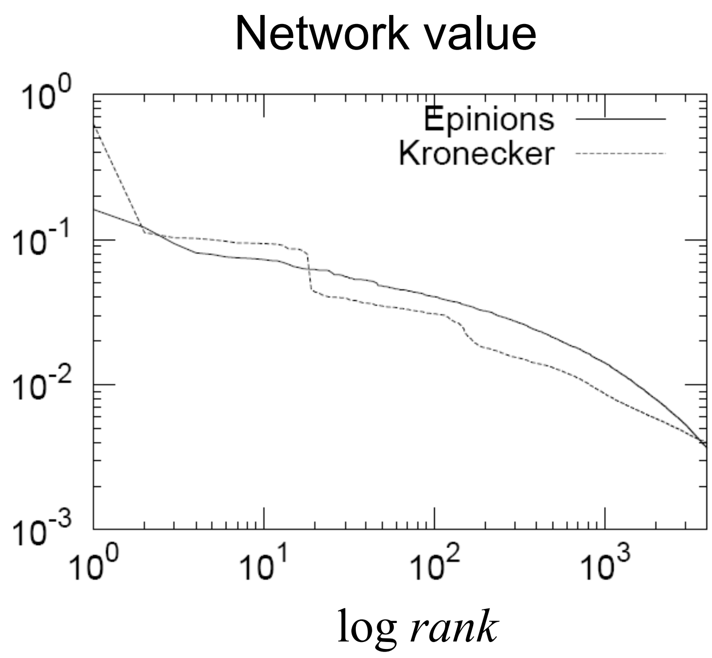
0.99	0.54
0.49	0.13





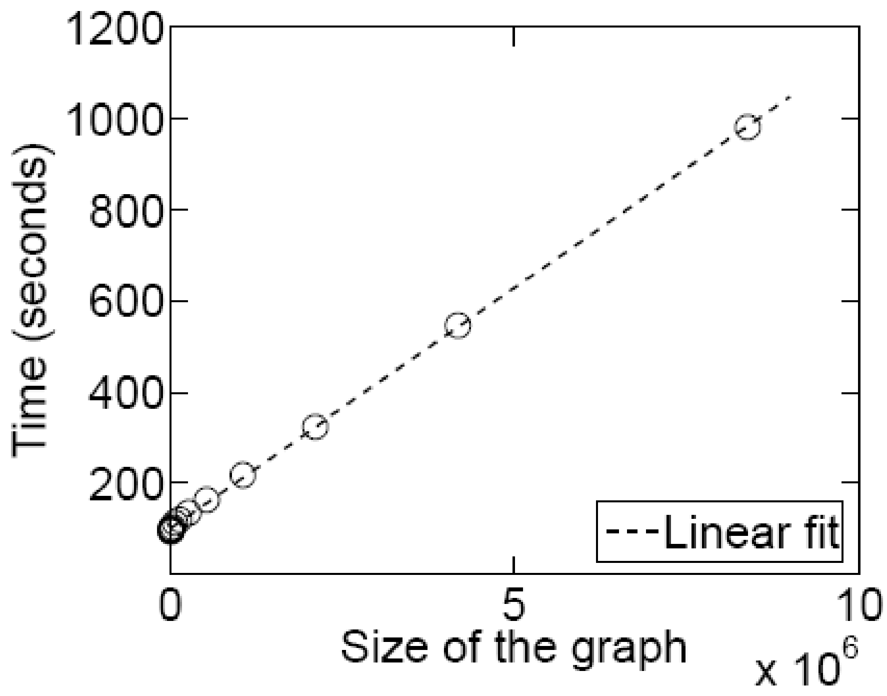
Epinions graph (N=76k, E=510k)





Scalability

- Fitting scales linearly with the number of edges



Conclusion

- Kronecker Graph model has
  - provable properties
  - small number of parameters
- Scalable algorithms for fitting Kronecker Graphs
- Efficiently search large space ($\sim 10^{1,000,000}$) of permutations
- Kronecker graphs fit well real networks using few parameters
- Kronecker graphs match graph properties without a priori deciding on which ones to fit

Conclusion

- Statistical properties of networks across various domains
  - Key to understanding the behavior of many "independent" nodes
- Models of network structure and growth
  - Help explain, think and reason about properties
- Prediction, understanding of the structure
  - Fitting the models

Why should we care?

- Gives insight into the graph formation process:
  - *Anomaly detection* – abnormal behavior, evolution
  - *Predictions* – predicting future from the past
  - *Simulations* of new algorithms where real graphs are hard/impossible to collect
  - *Graph sampling* – many real world graphs are too large to deal with
  - "*What if*" scenarios

Reflections

- How to systematically characterize the network structure?
- How do properties relate to one another?
- Is there something else we should measure?

Reflections

- Design systems (networks) that will
  - Be robust to node failures
  - Support local search (navigation): P2P networks
- Why are networks the way they are?
- Predict the future of the network?
- How should one be taking care of the network for it to grow organically?

References

- *Graphs over Time: Densification Laws, Shrinking Diameters and Possible Explanations,* by J. Leskovec, J. Kleinberg, C. Faloutsos, ACM KDD 2005
- *Graph Evolution: Densification and Shrinking Diameters*, by J. Leskovec, J. Kleinberg and C. Faloutsos, *ACM TKDD 2007*
- *Realistic, Mathematically Tractable Graph Generation and Evolution, Using Kronecker Multiplication*, by J. Leskovec, D. Chakrabarti, J. Kleinberg and C. Faloutsos, *PKDD 2005*
- *Scalable Modeling of Real Graphs using Kronecker Multiplication*, by J. Leskovec and C. Faloutsos, *ICML 2007*
- *The Dynamics of Viral Marketing, by* J. Leskovec, L. Adamic, B. Huberman, *ACM Electronic Commerce 2006*
- *Collective dynamics of 'small-world' networks*, by D. J. Watts and S. H. Strogatz, Nature 1998
- *Emergence of scaling in random networks*, by R. Albert and A.-L. Barabasi, Science 1999
- *On the evolution of random graphs*, by P. Erdos and A. Renyi, Publication of the Mathematical Institute of the Hungarian Acadamy of Science, 1960
- *Power-law distributions in empirical data*, by A. Clauset, C. Shalizi, M. Newman, 2007

References

- *The structure and function of complex networks*, M. Newman, SIAM Review 2003
- *Hierarchical Organization in Complex Networks*, Ravasz and Barabasi, Physical Review E 2003
- *A random graph model for massive graphs*, W. Aiello, F. Chung and L. Lu, STOC 2000
- *Community structure in social and biological networks*, by Girvan and Newman, PNAS 2002
- *On Power-law Relationships of the Internet Topology,* by Faloutsos, Faloutsos, and Faloutsos, SIGCOM 1999
- *Power laws, Pareto distributions and Zipf's law,* by M. Newman, Contemporary Physics 2005
- *Social Network Analysis : Methods and Applications*, by Wasserman, Cambridge University Press 1994
- *The web as a graph: Measurements, models and methods*, by J. Kleinberg and S. R. Kumar, P. Raghavan, S. Rajagopalan and A. Tomkins, COCOON 1998
- *Stochastic Kronecker Graphs,* by M. Mahdian and Y. Xu, Workshop on Algorithms and Models for the Web-Graph (WAW) 2007

Some slides and plots borrowed from L. Adamic, M. Newman, M. Joseph, A. Barabasi, J. Kleinberg, D. Lieben-Nowell, S. Valverde, and R. Sole

Coming up next…

Diffusion and cascading behavior in networks

- Viral Marketing: How do people make recommendations?
- How does information and viruses propagate in networks?
- How to detect cascades and find influential nodes?