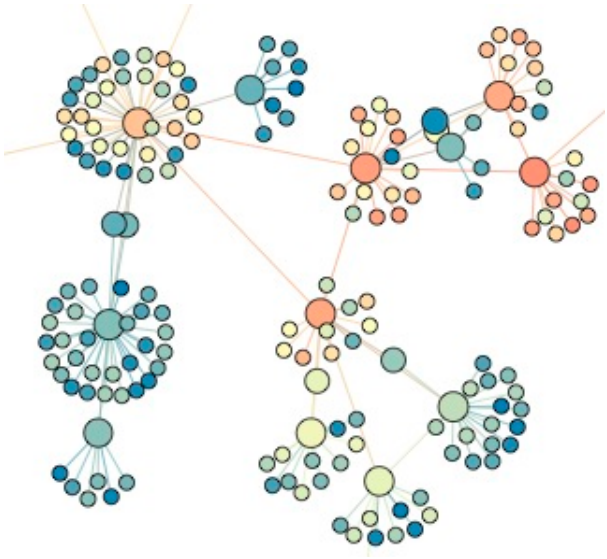


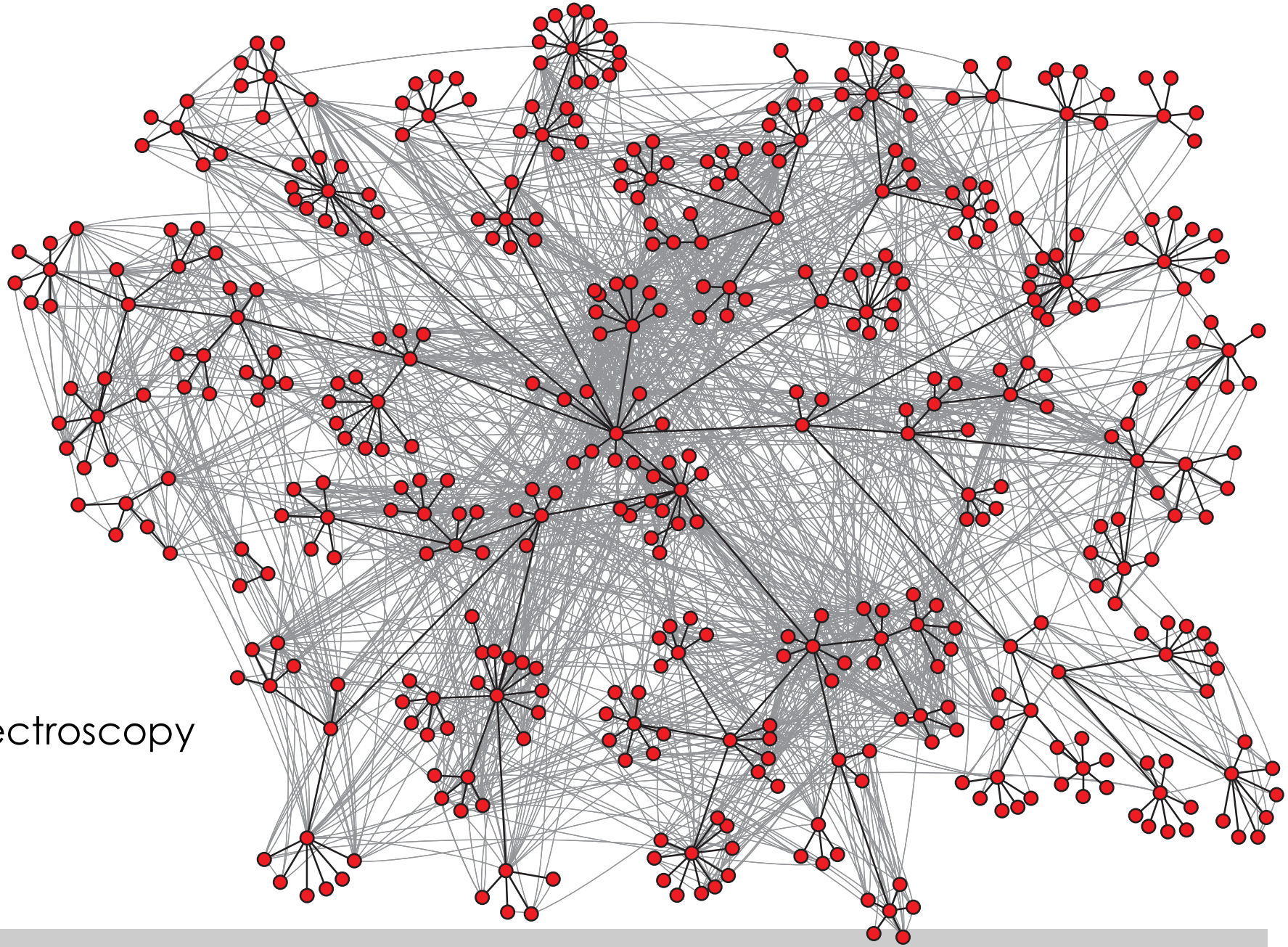
Community Finding

CS224W

Outline

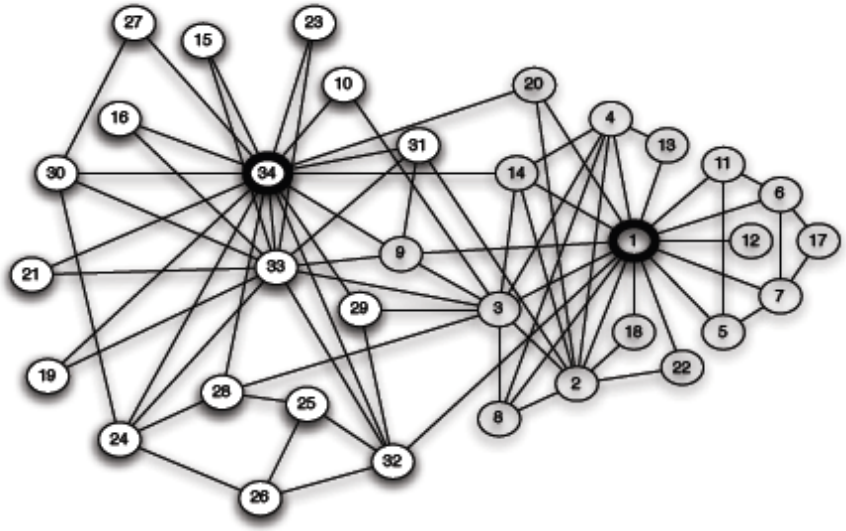
- why do we look for community structure?
- we need to define it in order to find it
- approaches to finding it

Why look for community structure?

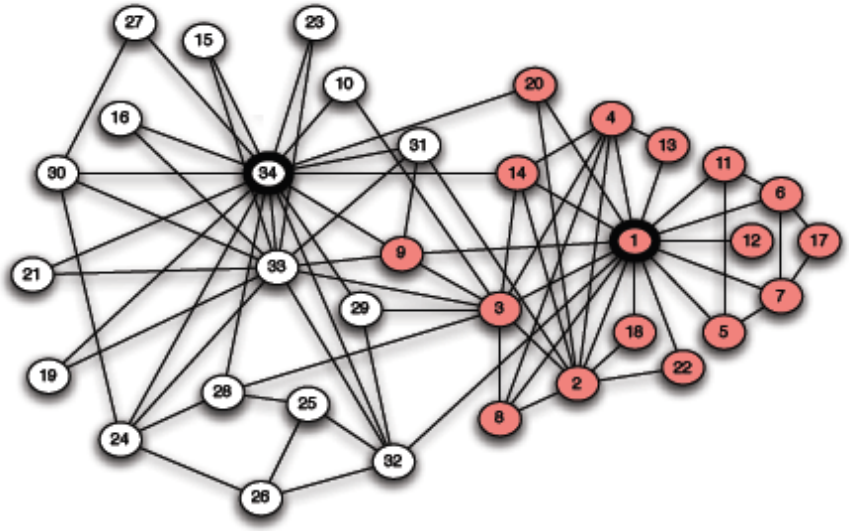


example:
email spectroscopy

Zachary Karate Club



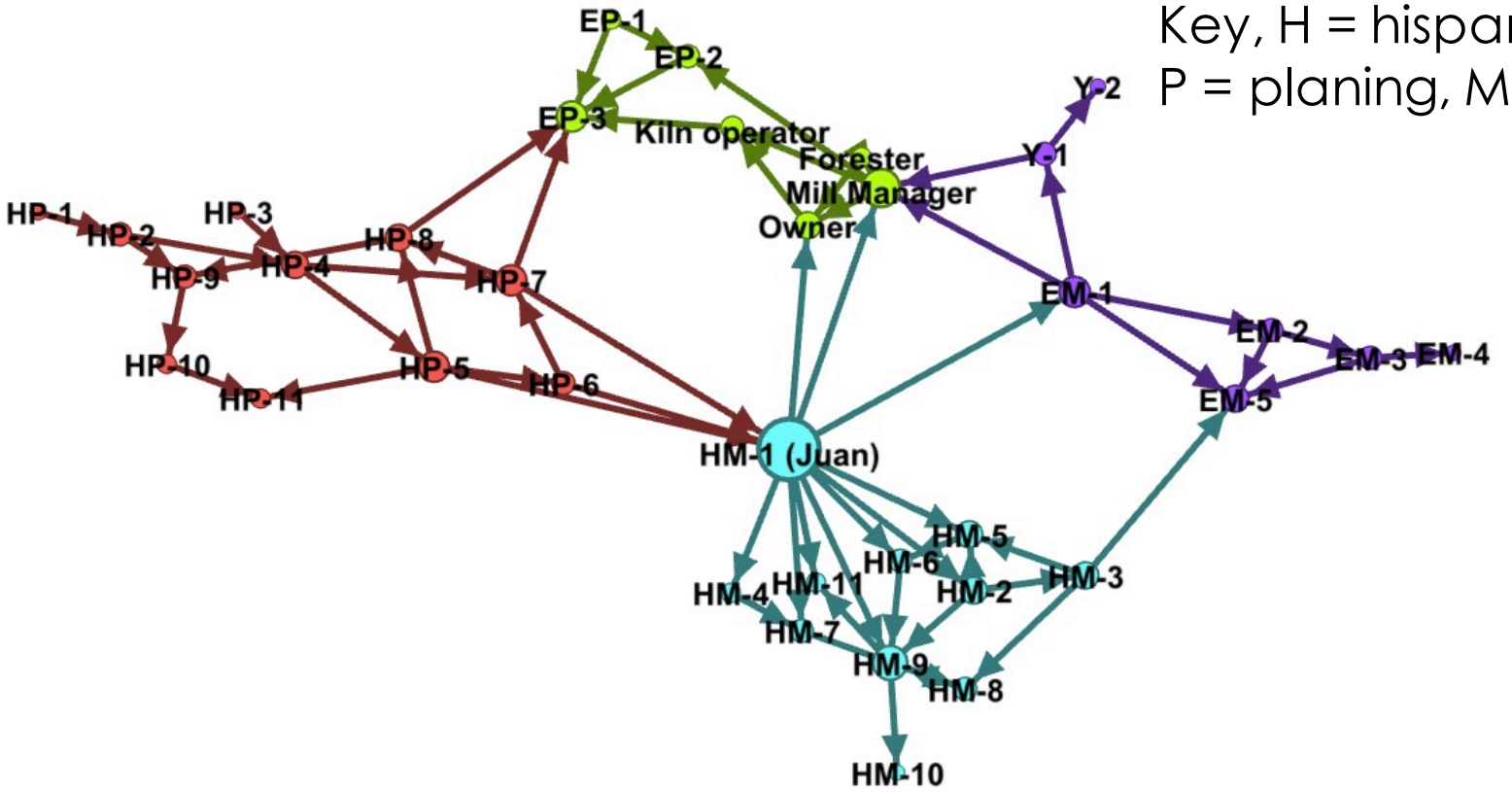
(a) *Karate club network*



(b) *After a split into two clubs*

source:Easley/Kleinberg

Why look for community structure?



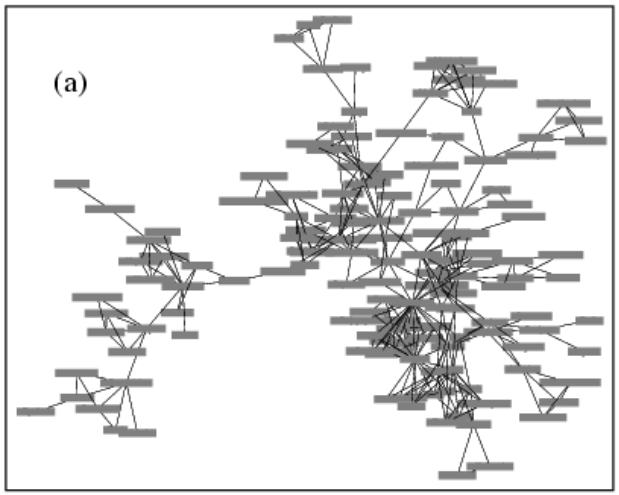
Key, H = hispanic, E = english
P = planing, M = milling, Y = yard

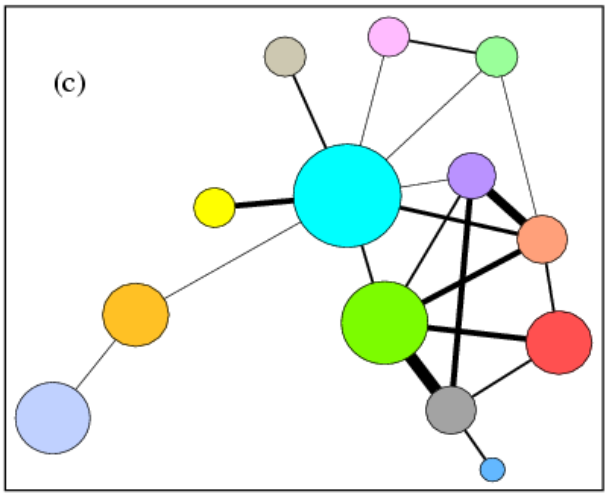
- The management at the sawmill was having difficulty persuading the workers to adopt a new plan, even though everyone would benefit. In particular the Hispanic workers (H) were reluctant to agree. The management called in a sociologist who mapped out who talked to whom regularly. Then they suggested that the management talk to Juan and have him talk to the Hispanic workers. It was a success, promptly everyone was on board with the new plan. Why?

Sawmill network: source Exploratory Social Network Analysis with Pajek

Why do it: gain understanding

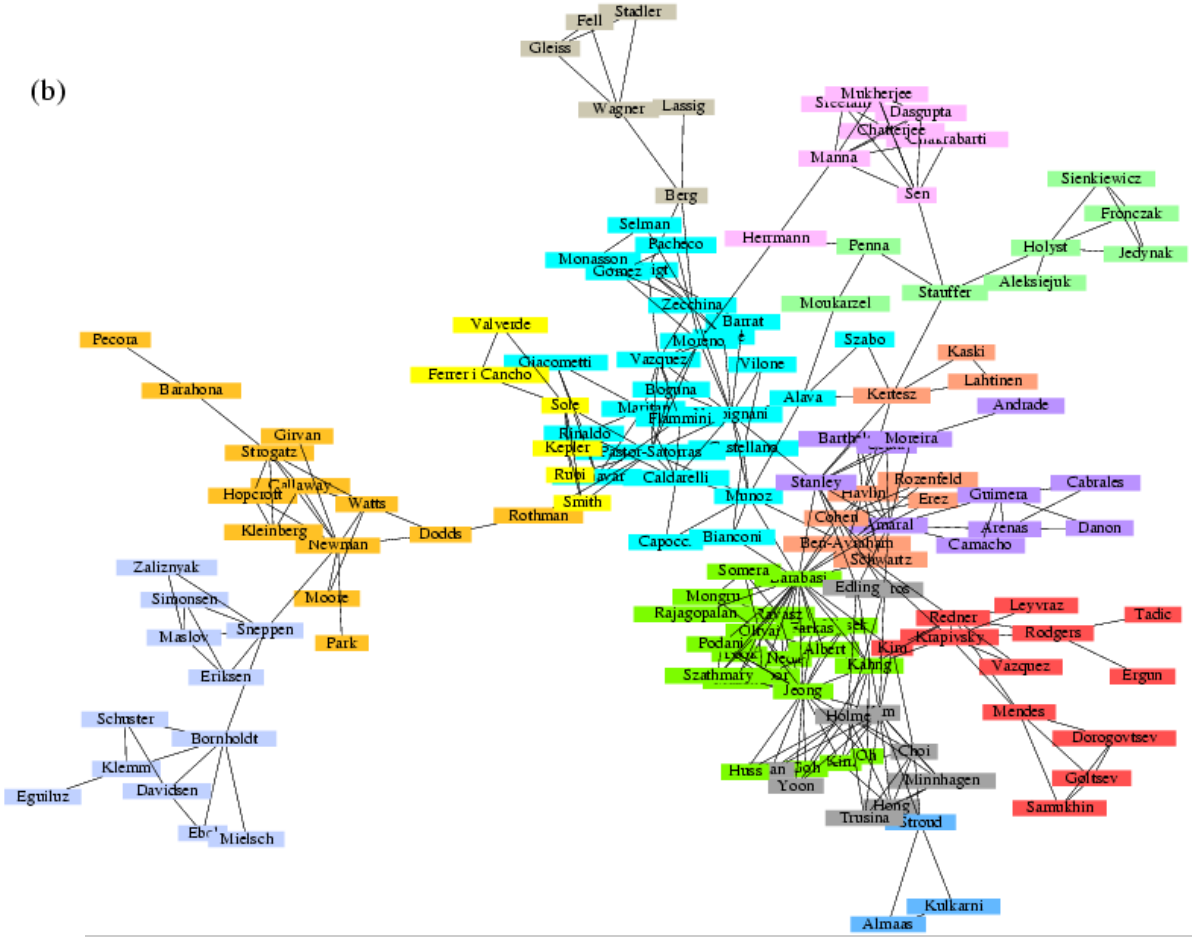
- Gain understanding of networks
  - Discover communities of practice
  - Measure isolation of groups
  - Understand opinion dynamics / adoption





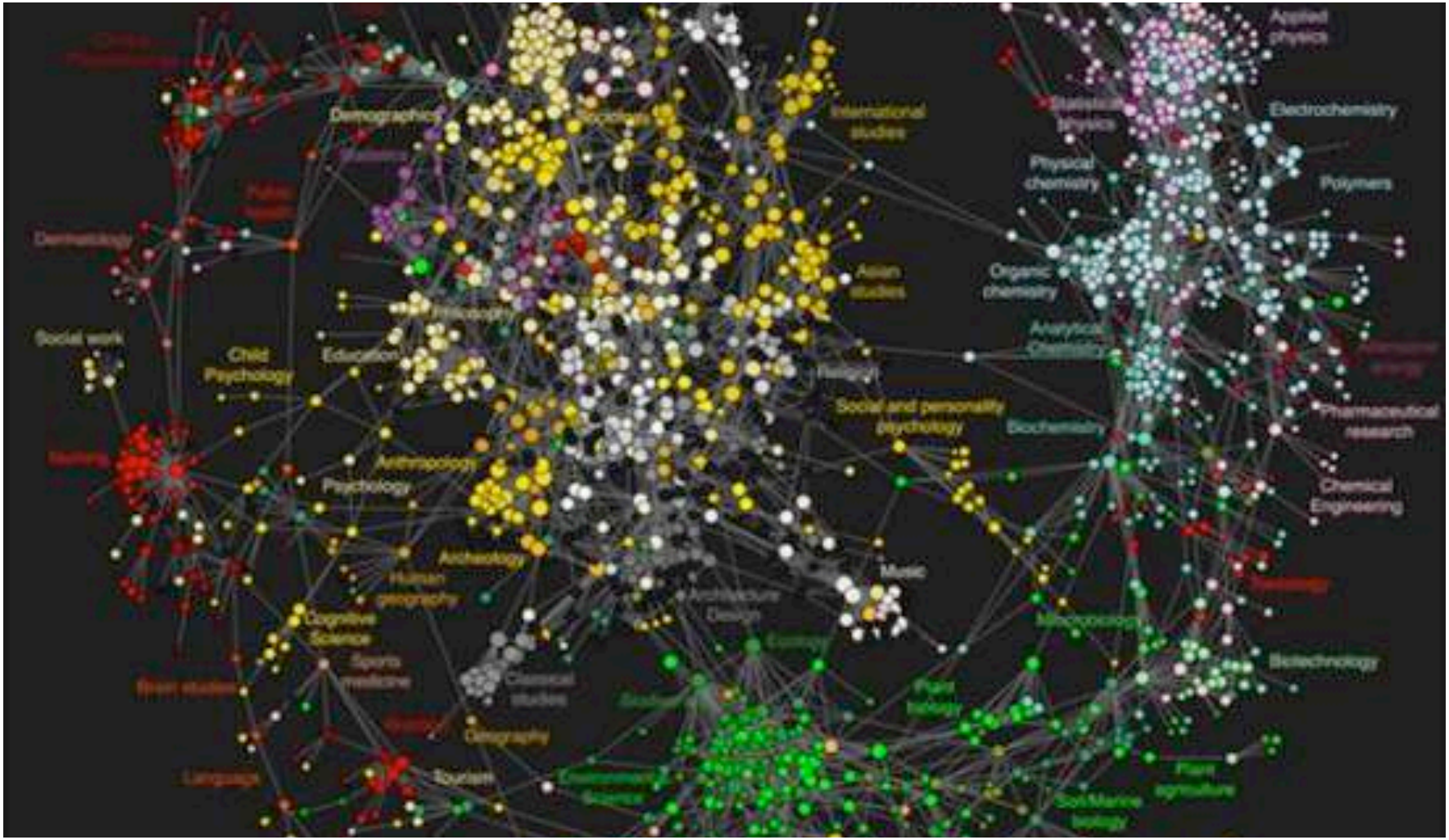
Why do it: visualize

- Communities help to "aggregate" network data



high-res maps of science

http://www.plosone.org/article/info%3Adoi%2F10.1371%2Fjournal.pone.0004803

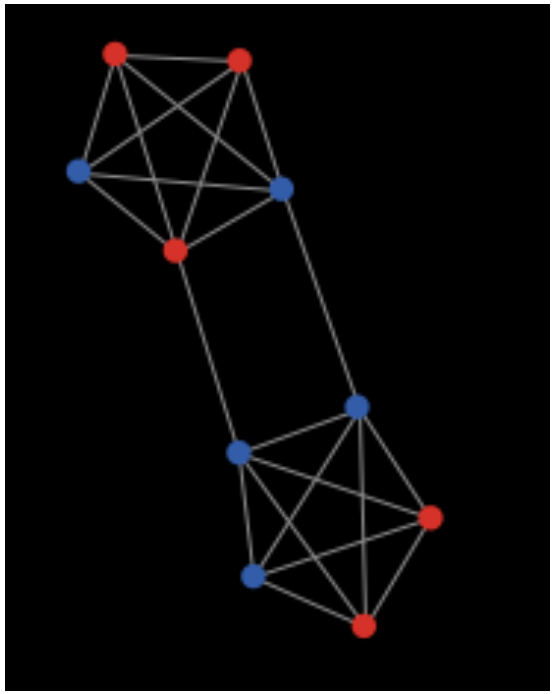


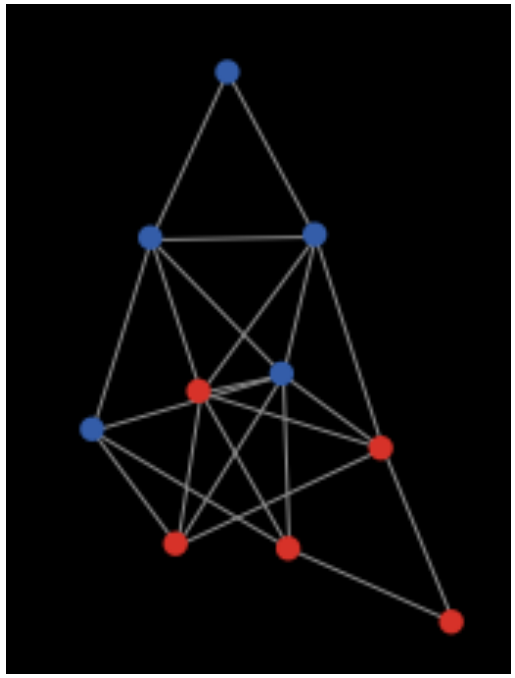
Why do it: store & compute

- If your network needs to be distributed accross the world or over many machines, you want to minimize the number of edges in-between

reminder: opinion formation and community structure

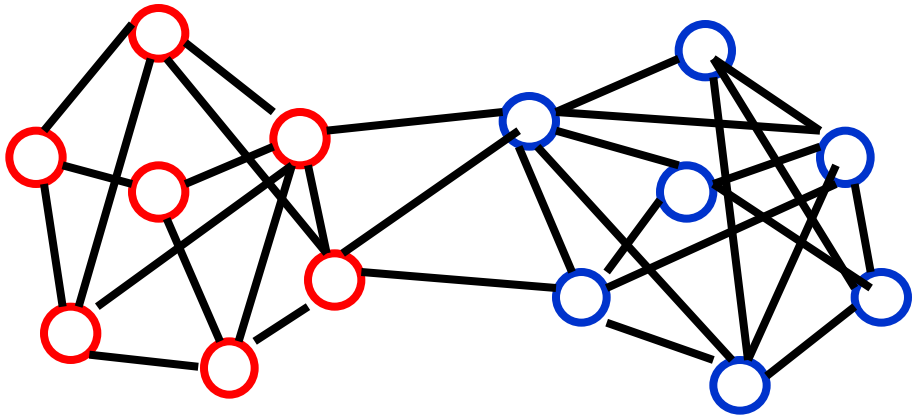
- each node adopts the majority opinion of its neighbors (flips a coin if it's a tie)





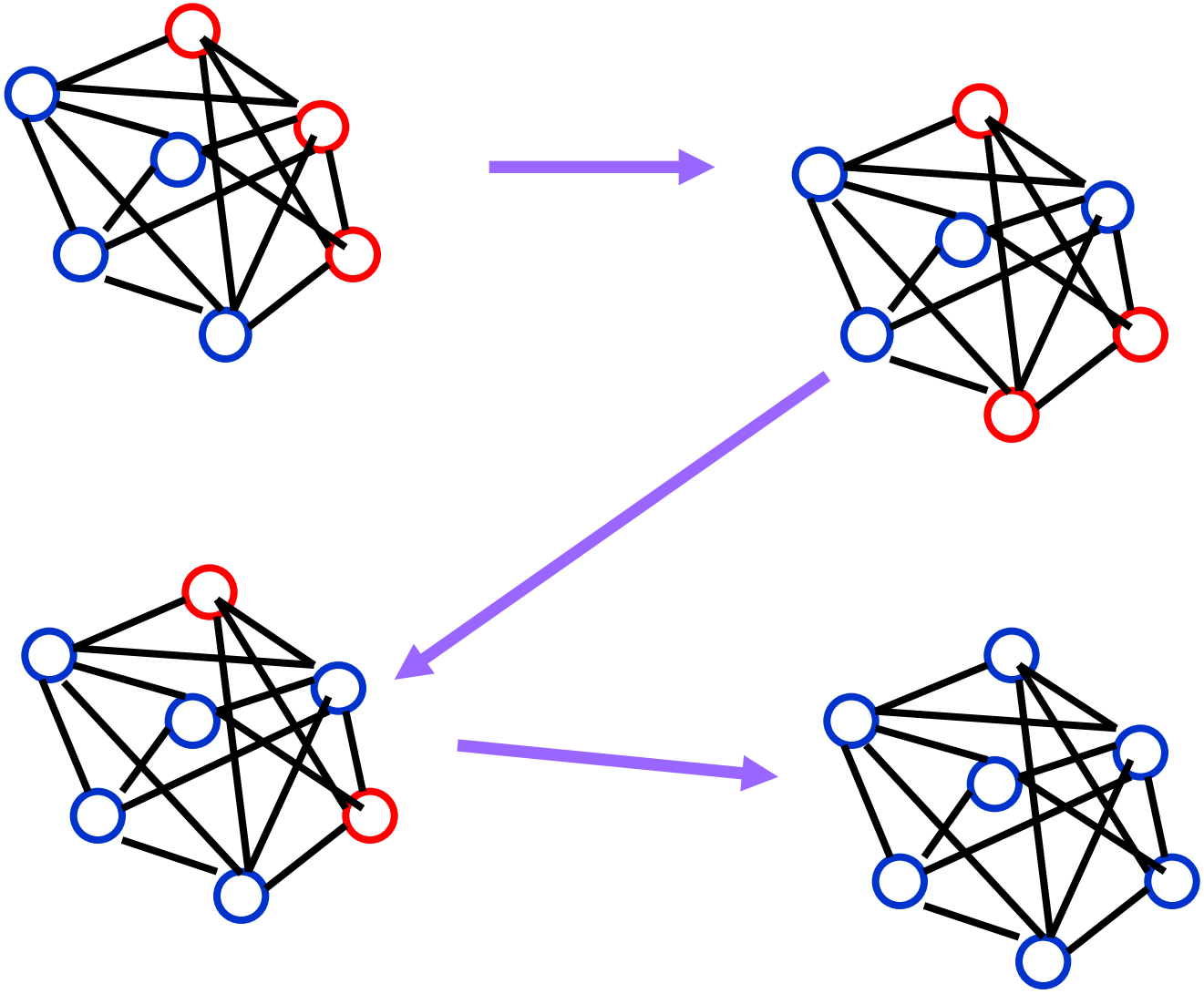
Why care about group cohesion?

- opinion formation and uniformity



- if each node adopts the opinion of the majority of its neighbors, it is possible to have different opinions in different cohesive subgroups

within a cohesive subgroup – greater uniformity

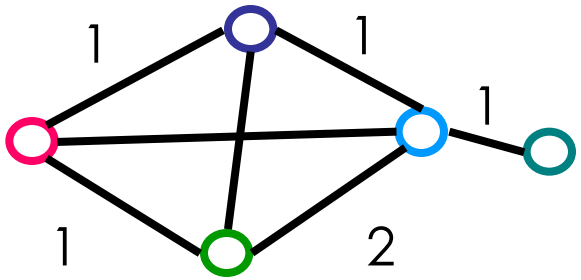


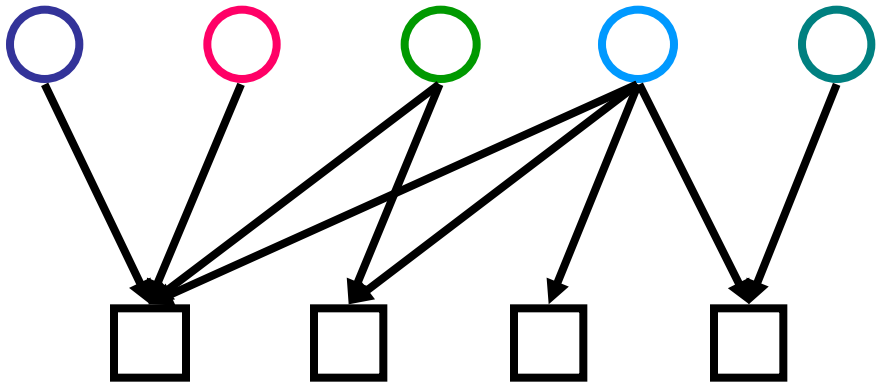
What makes a community?

- mutuality of ties
  - everybody in the group knows everybody else
- frequency of ties among members
  - everybody in the group has links to at least k others in the group
- closeness or reachability of subgroup members
  - individuals are separated by at most n hops
- relative frequency of ties among subgroup members compared to nonmembers

Affiliation networks

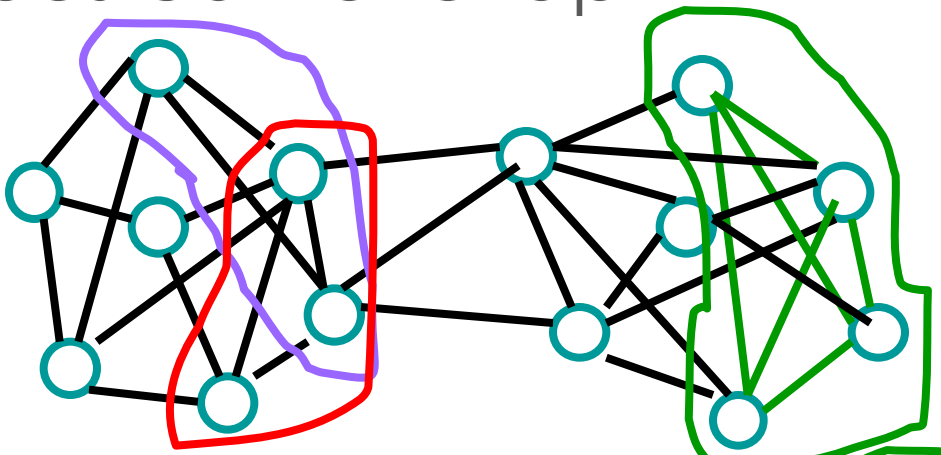
- otherwise known as
  - membership network
    - e.g. board of directors
  - hypernetwork or hypergraph
  - bipartite graphs
  - interlocks

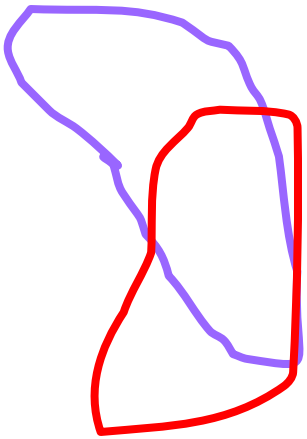




Cliques

- Every member of the group has links to every other member
- Cliques can overlap





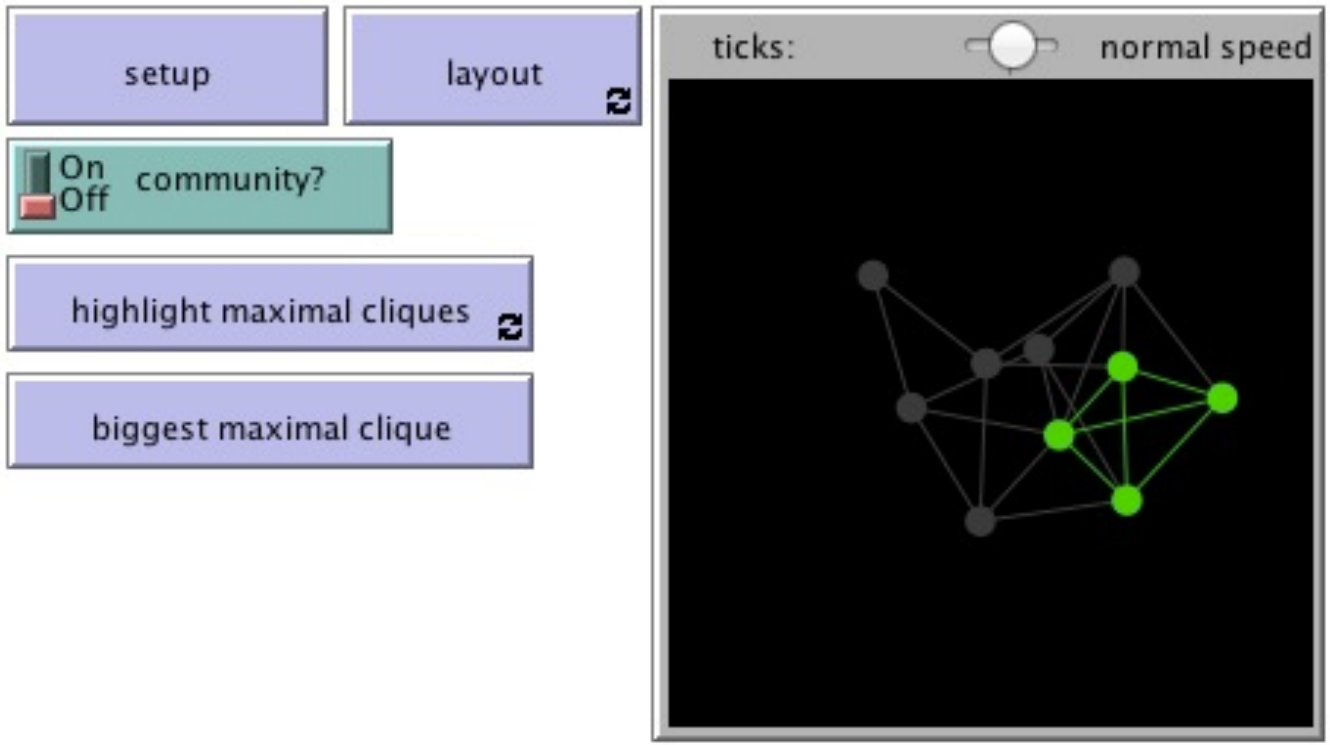
overlapping cliques of size 3

clique of size 4

Cliques & community structure

- Go to http://web.stanford.edu/class/cs224w/NetLogo/Cliques.nlogo

- Try the ER vs. community structure setup (they are the same as for the opinion formation model)



Quiz question

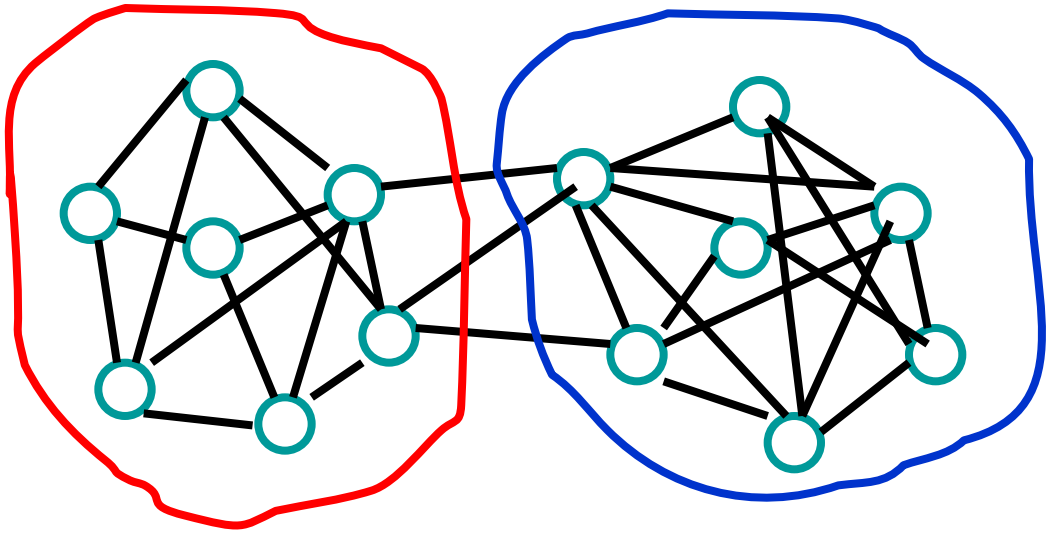
- Which has a larger maximal clique?
  - network with community structure
  - the equivalent ER random graph

Meaningfulness of cliques

- Not robust
  - one missing link can disqualify a clique
- Not interesting
  - everybody is connected to everybody else
  - no core-periphery structure
  - no centrality measures apply
- How cliques overlap can be more interesting than that they exist

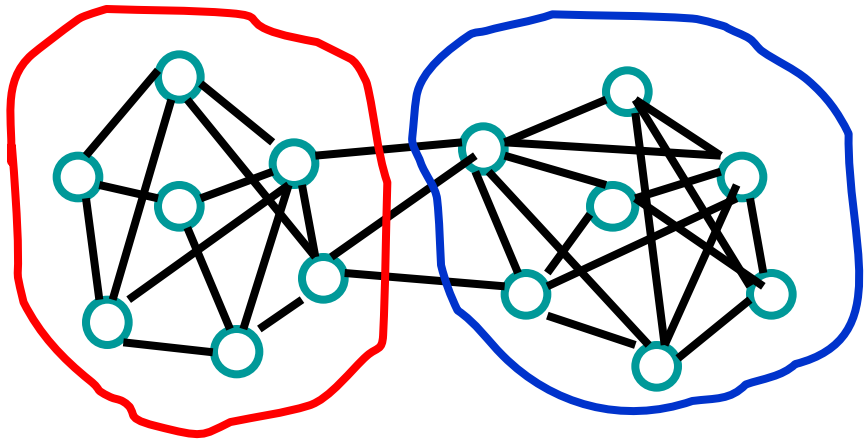
k-cores: similar idea, less stringent

- Each node within a group is connected to k other nodes in the group



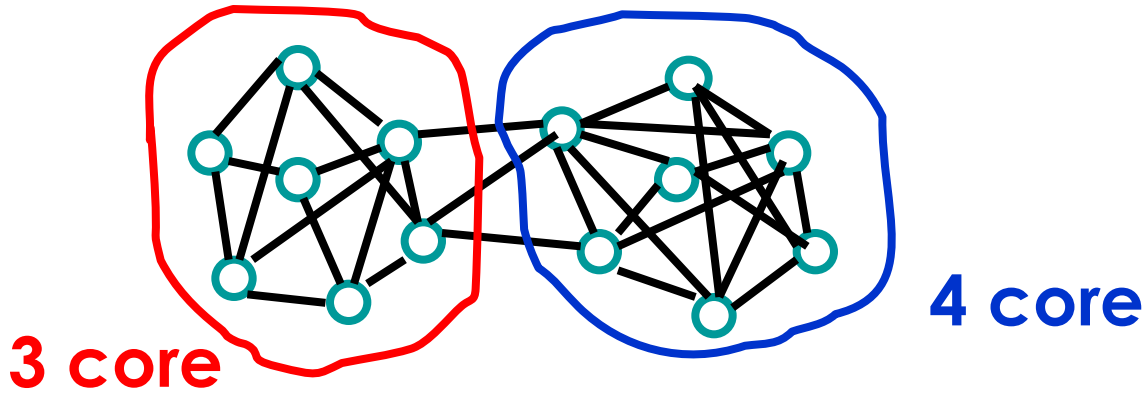
Quiz Question

- What is the "k" for the core circled in red?
- What is the "k" for the core circled in blue?

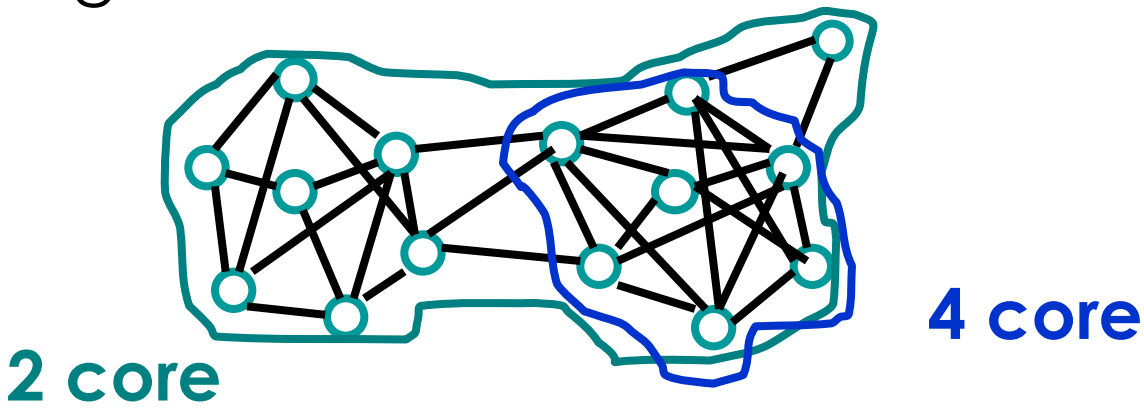


k-cores

- Each node within a group is connected to k other nodes in the group

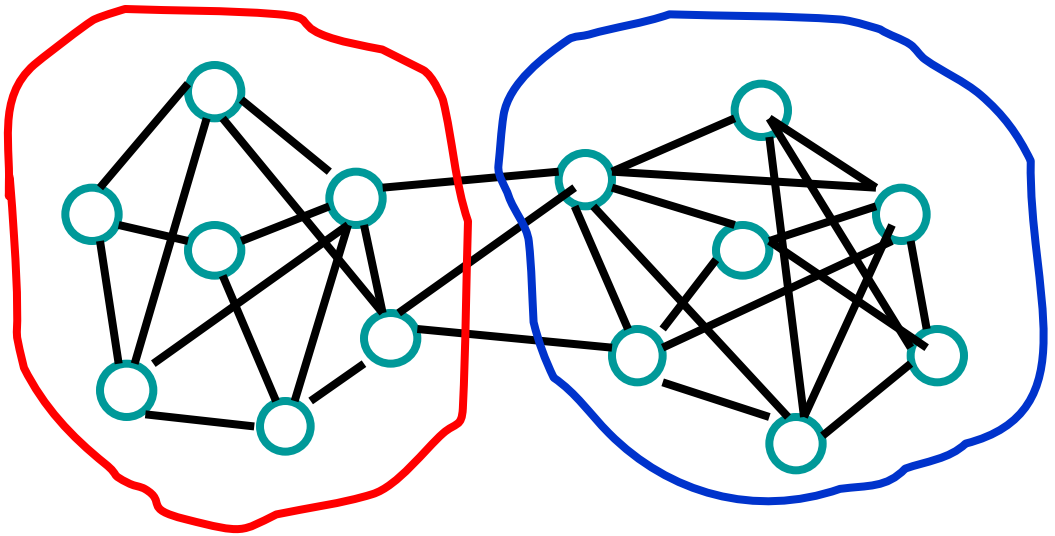


- but even this is too stringent of a requirement for identifying natural communities



subgroups based on reachability and diameter

- n – cliques
  - maximal distance between any two nodes in subgroup is n

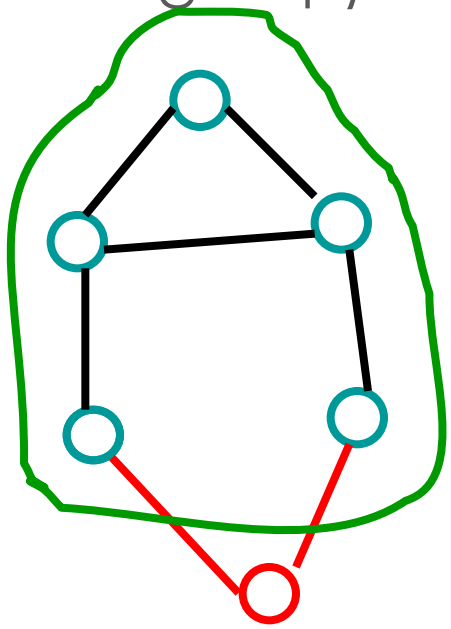


2-cliques

- theoretical justification
  - information flow through intermediaries

considerations with n-cliques

- problem
  - diameter may be greater than n
  - n-clique may be disconnected (paths go through nodes not in subgroup)



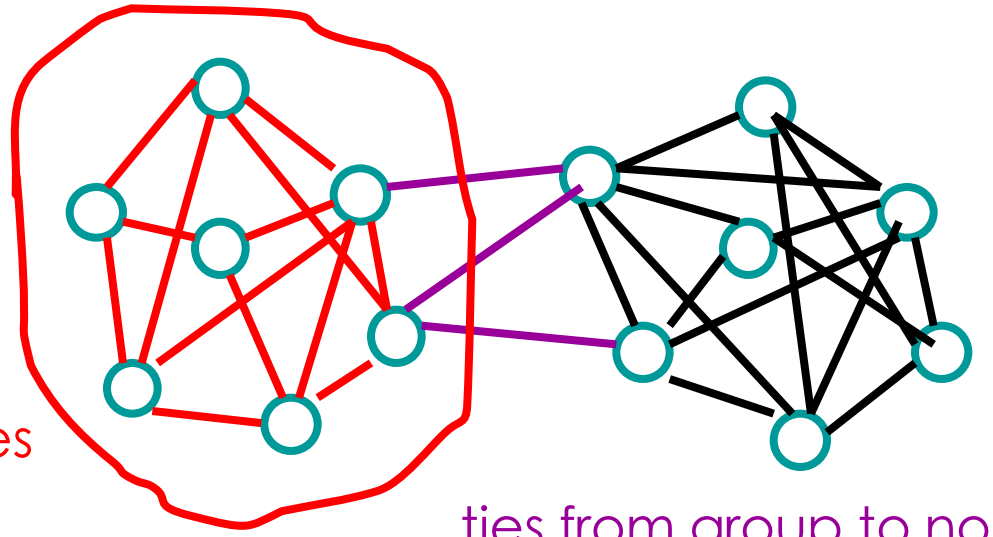
2 – clique
diameter = 3

path outside the 2-clique

- fix
  - n-club: maximal subgraph of diameter 2

p-cliques: frequency of in group ties

- partition the network into clusters where vertices have at least a proportion p (number between 0 and 1) of neighbors inside the cluster.



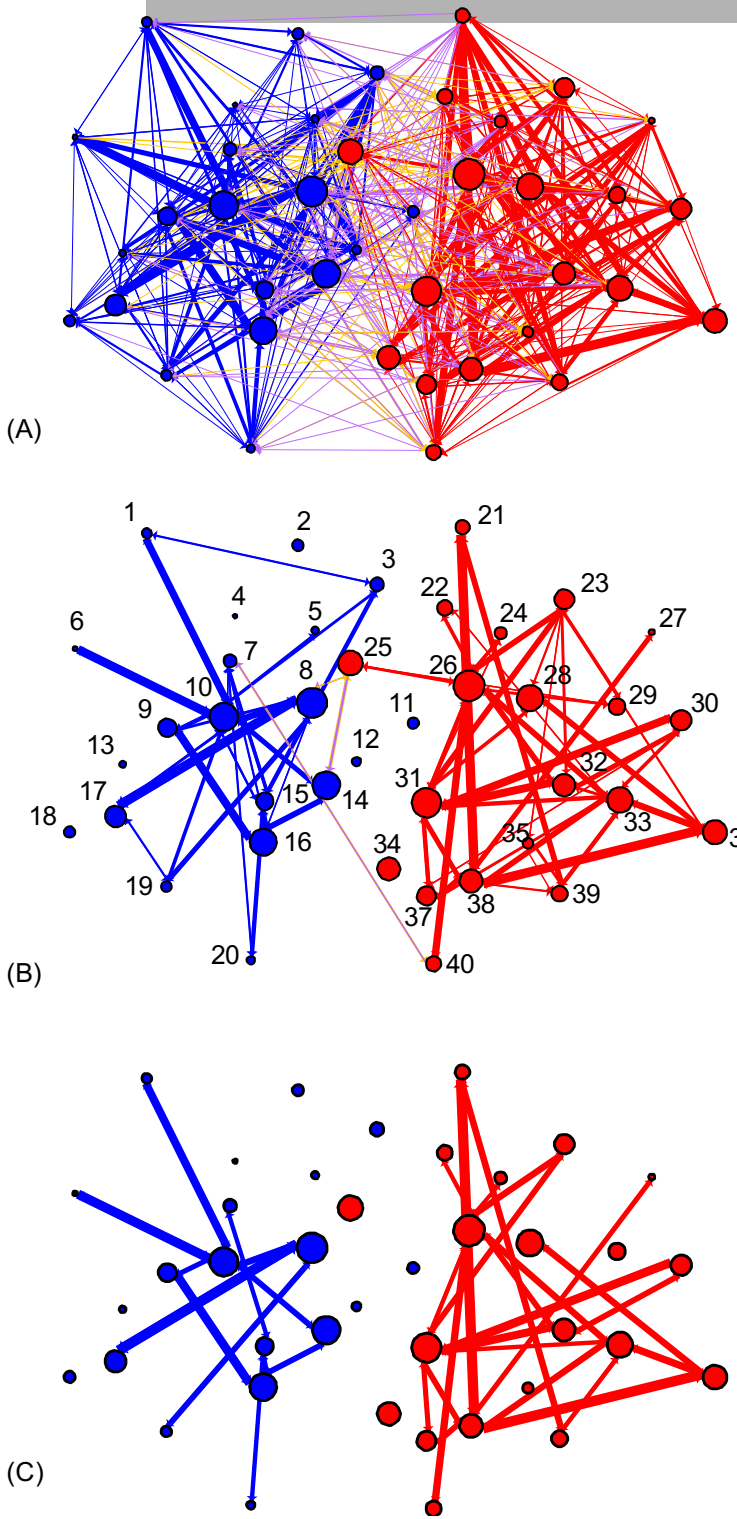
within-group ties

ties from group to nodes external to the group

cohesion in directed & weighted networks

- something we've already learned how to do:
  - find strongly connected components
- keep only a subset of ties before finding connected components
  - reciprocal ties
  - edge weight above a threshold

Example: political blogs
(Aug 29th – Nov 15th, 2004)



1 Digbys Blog
2 JamesWalcott
3 Pandagon
4 blog.johnkerry.com
5 Oliver Willis
6 America Blog
7 Crooked Timber
8 Daily Kos
9 American Prospect
10 Eschaton
11 Wonkette
12 Talk Left
13 Political Wire
14 Talking Points Memo
15 Matthew Yglesias
16 Washington Monthly
17 MyDD
18 Juan Cole
19 Left Coaster
20 Bradford DeLong

21 JawaReport
22 Voka Pundit
23 Roger LSimon
24 Tim Blair
25 Andrew Sullivan
26 Instapundit
27 Blogs for Bush
28 LittleGreen Footballs
29 Belmont Club
30 Captain's Quarters
31 Powerline
32 HughHewitt
33 INDCJournal
34 Real Clear Politics
35 Winds of Change
36 Allahpundit
37 Michelle Malkin
38 WizBang
39 Dean's World
40 Volokh

A) all citations between A-list blogs in 2 months preceding the 2004 election

B) citations between A-list blogs with at least 5 citations in both directions

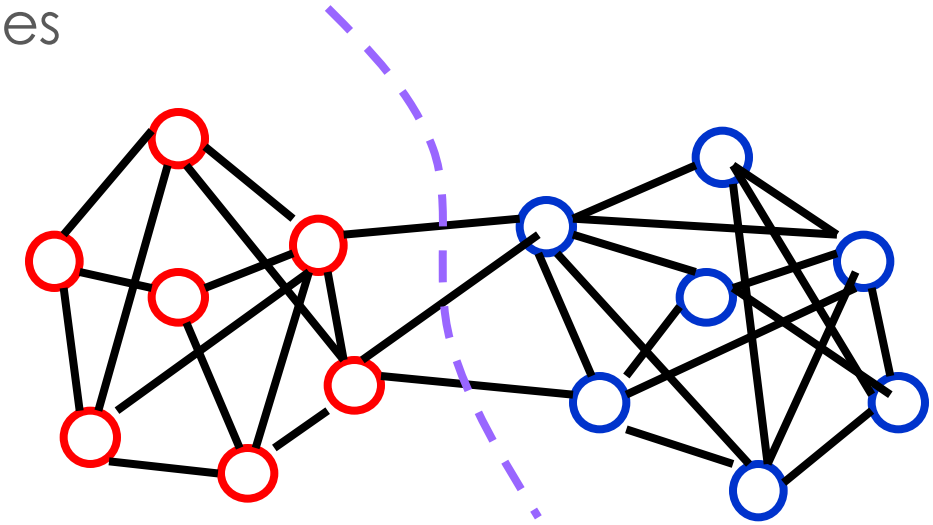
C) edges further limited to those exceeding 25 combined citations

only 15% of the citations bridge communities

source: Adamic & Glance, LinkKDD2005

Community finding vs. other approaches

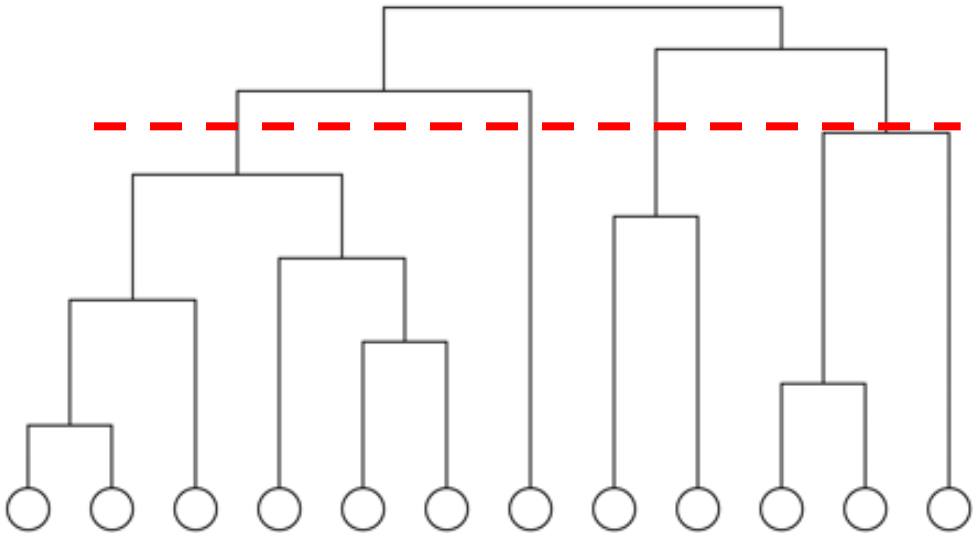
- Social and other networks have a natural community structure
- We want to discover this structure rather than impose a certain size of community or fix the number of communities



- Without "looking", can we discover community structure in an automated way?

Hierarchical clustering

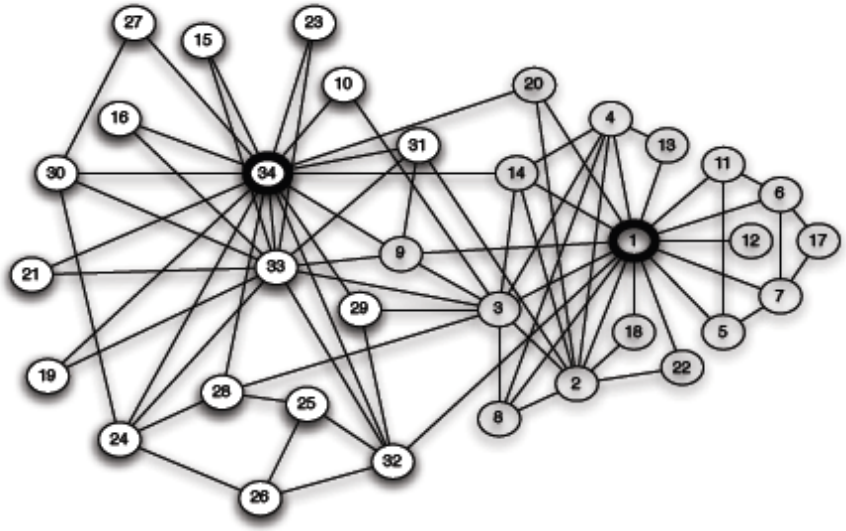
- Process:
  - after calculating the "distances" for all pairs of vertices
  - start with all n vertices disconnected
  - add edges between pairs one by one in order of decreasing weight
  - result: nested components, where one can take a 'slice' at any level of the tree



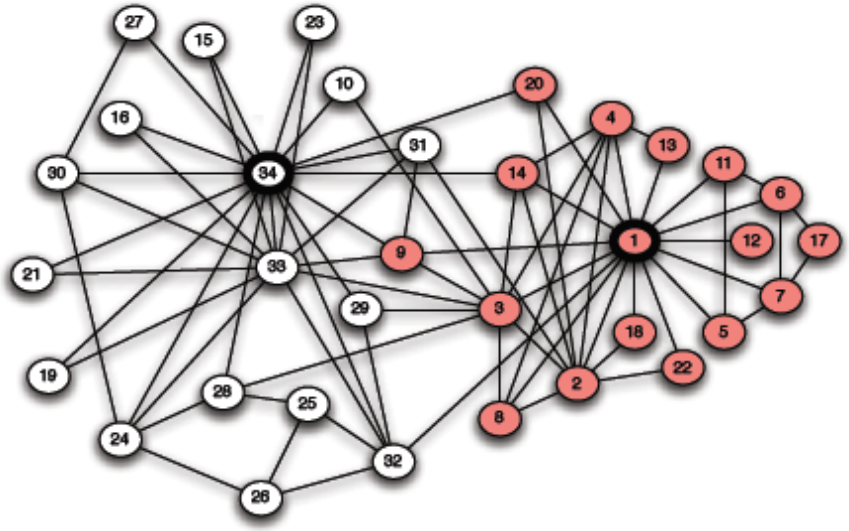
Hierarchical clustering

- Process:
  - after calculating the weights W for all pairs of vertices
  - start with all n vertices disconnected
  - add edges between pairs one by one in order of decreasing weight

Zachary Karate Club

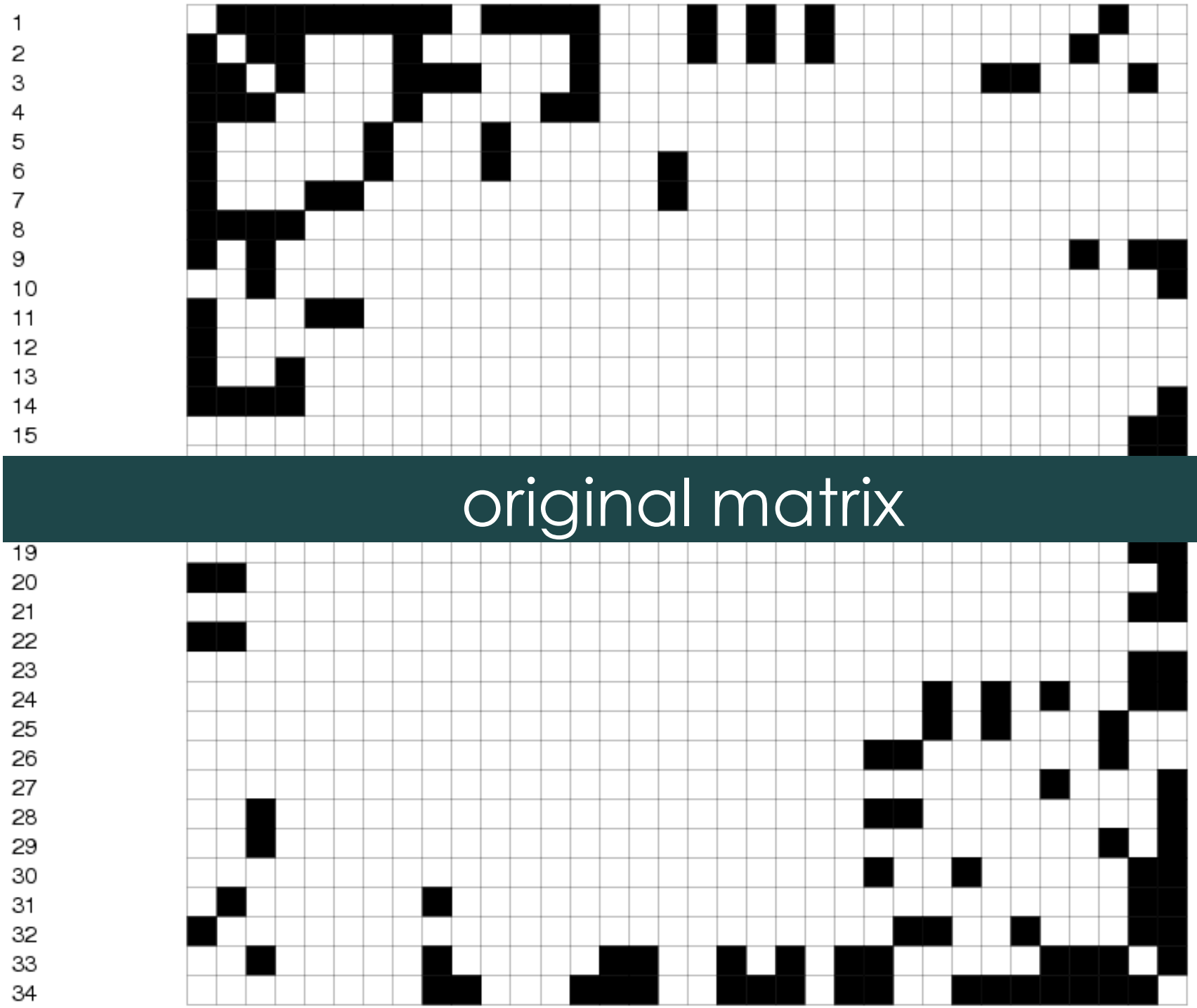


(a) *Karate club network*



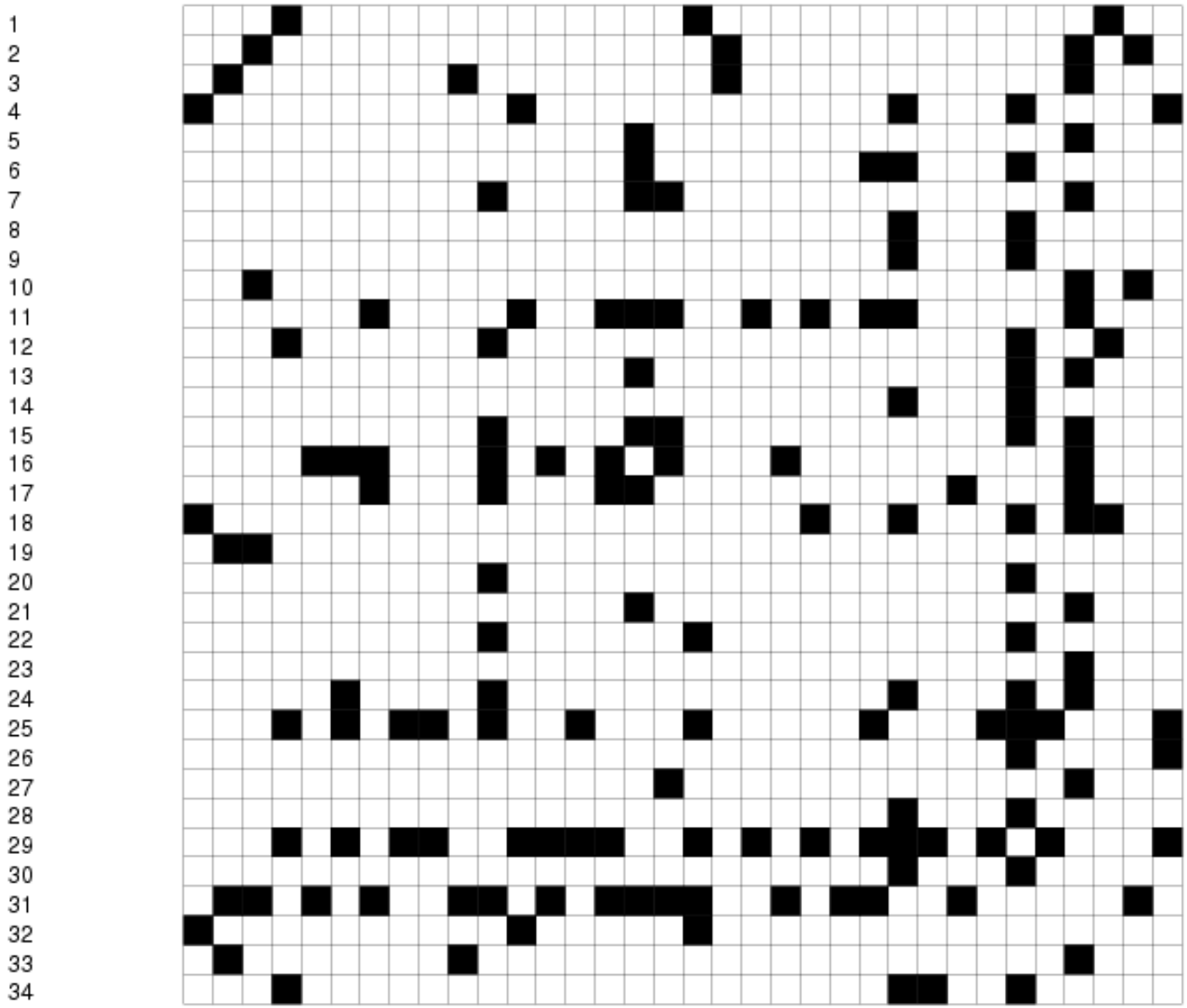
(b) *After a split into two clubs*

source:Easley/Kleinberg

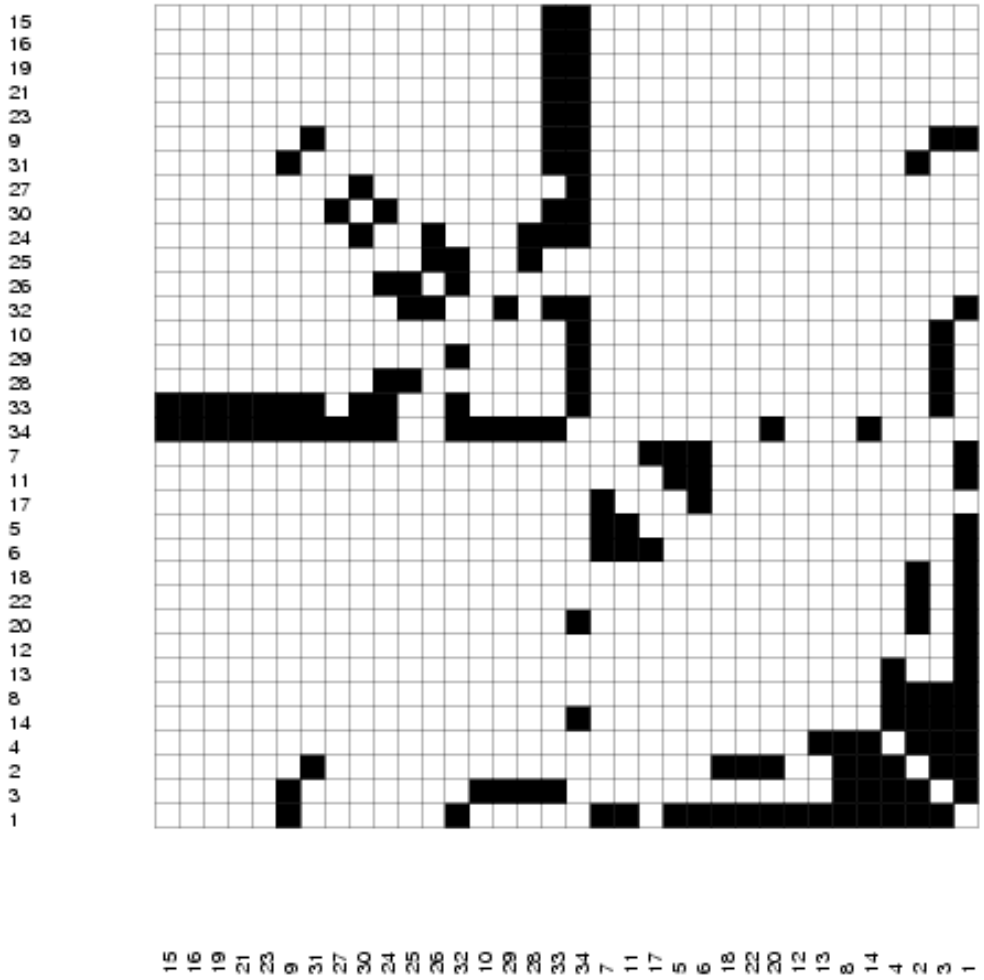


original matrix

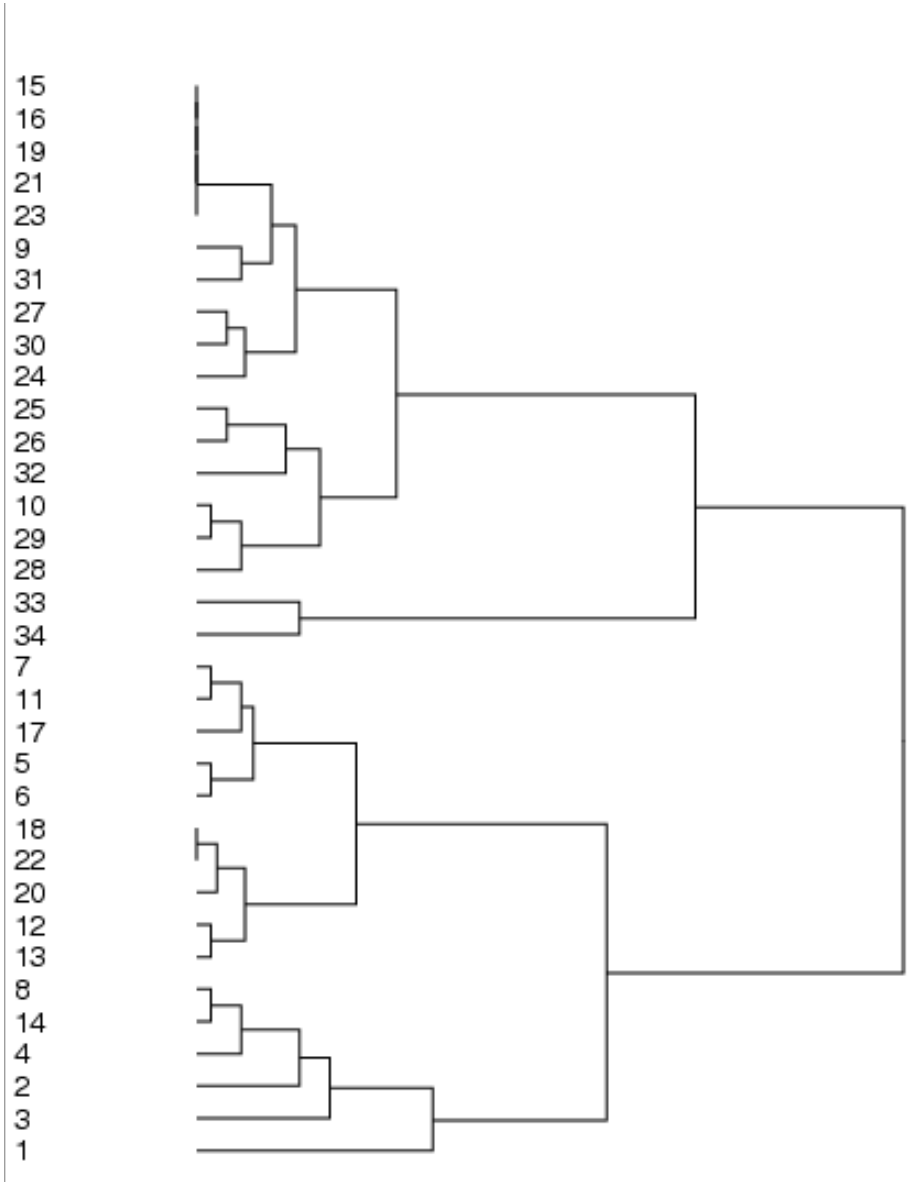
randomized karate club matrix



permuted matrix



dendrogram

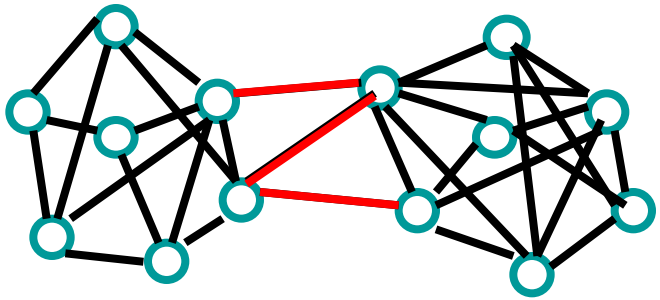

15
16
19
21
23
9
31
27
30
24
25
26
32
10
29
28
33
34
7
11
17
5
6
18
22
20
12
13
8
14
4
2
3
1

betweenness clustering

- Algorithm
  - compute the betweenness of all edges
  - while (betweenness of any edge > threshold):
    - remove edge with highest betweenness
    - recalculate betweenness

- Betweenness needs to be recalculated at each step
  - removal of an edge can impact the betweenness of another edge
  - very expensive: all pairs shortest path – $O(N^3)$
  - may need to repeat up to N times
  - does not scale to more than a few hundred nodes, even with the fastest algorithms

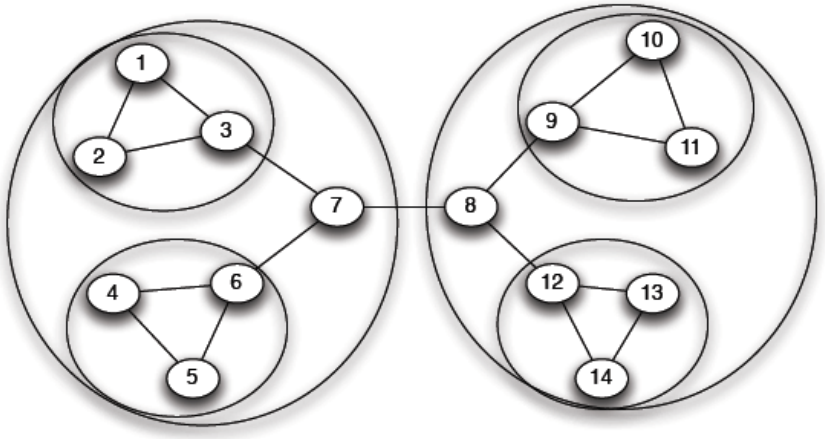
betweenness clustering algorithm

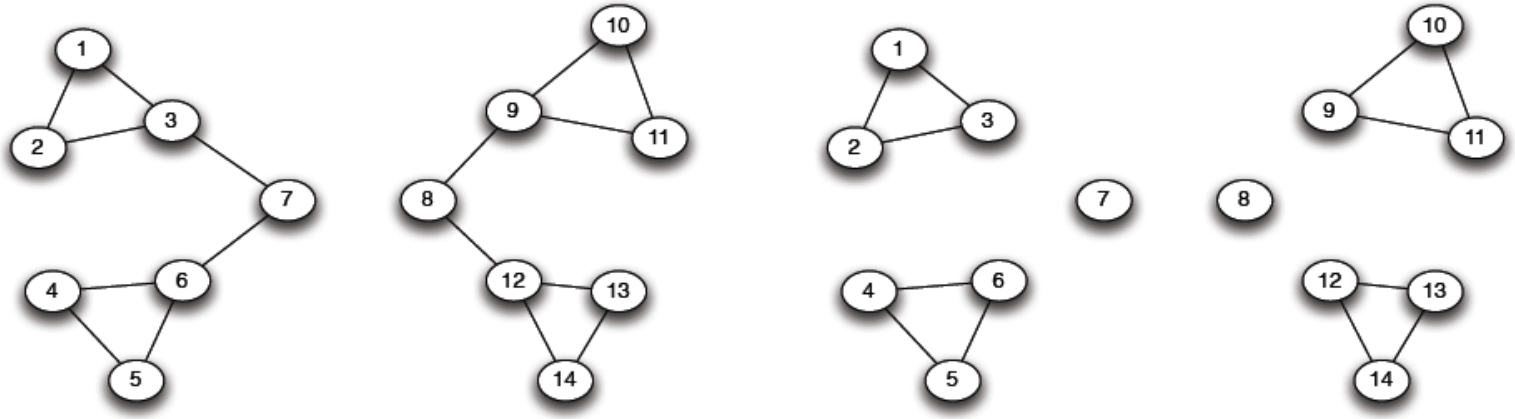


http://web.stanford.edu/class/cs224w/NetLogo/GirvanNewman.nlogo

betweenness clustering:

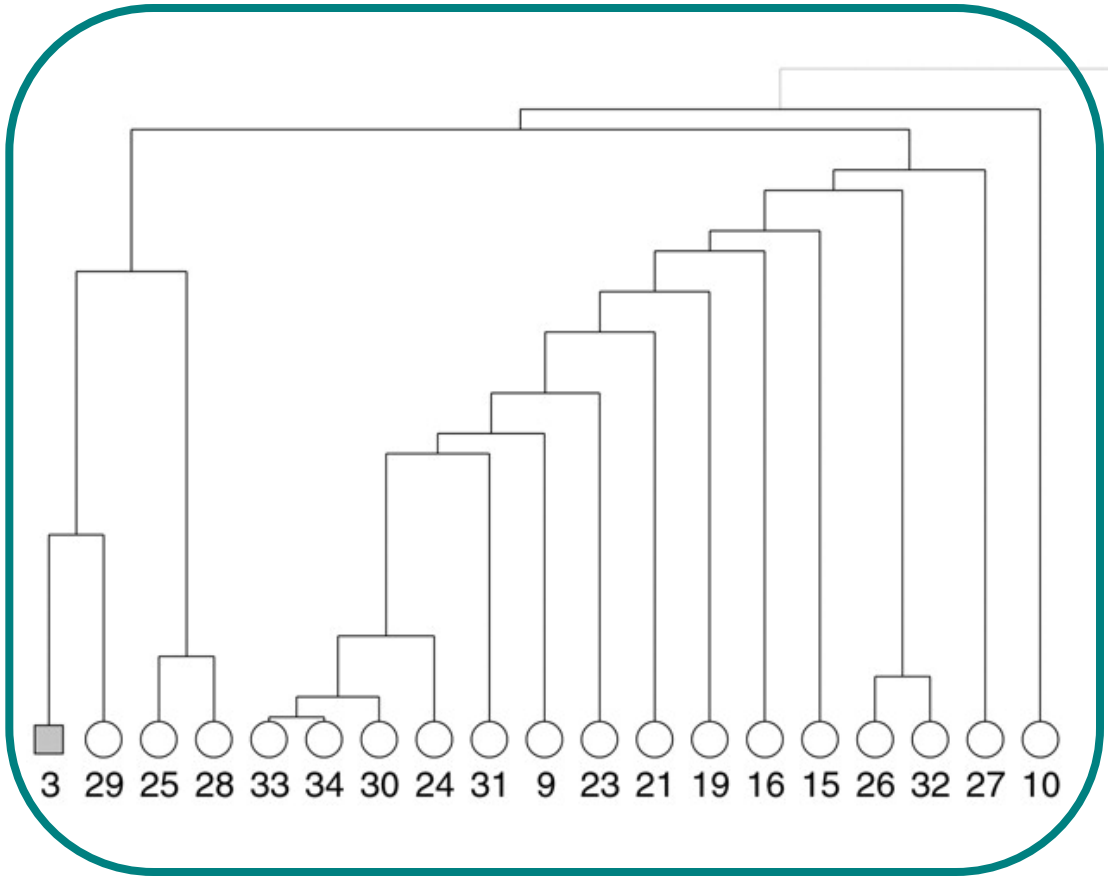
- successively remove edges of highest betweenness (the bridges, or local bridges), breaking up the network into separate components

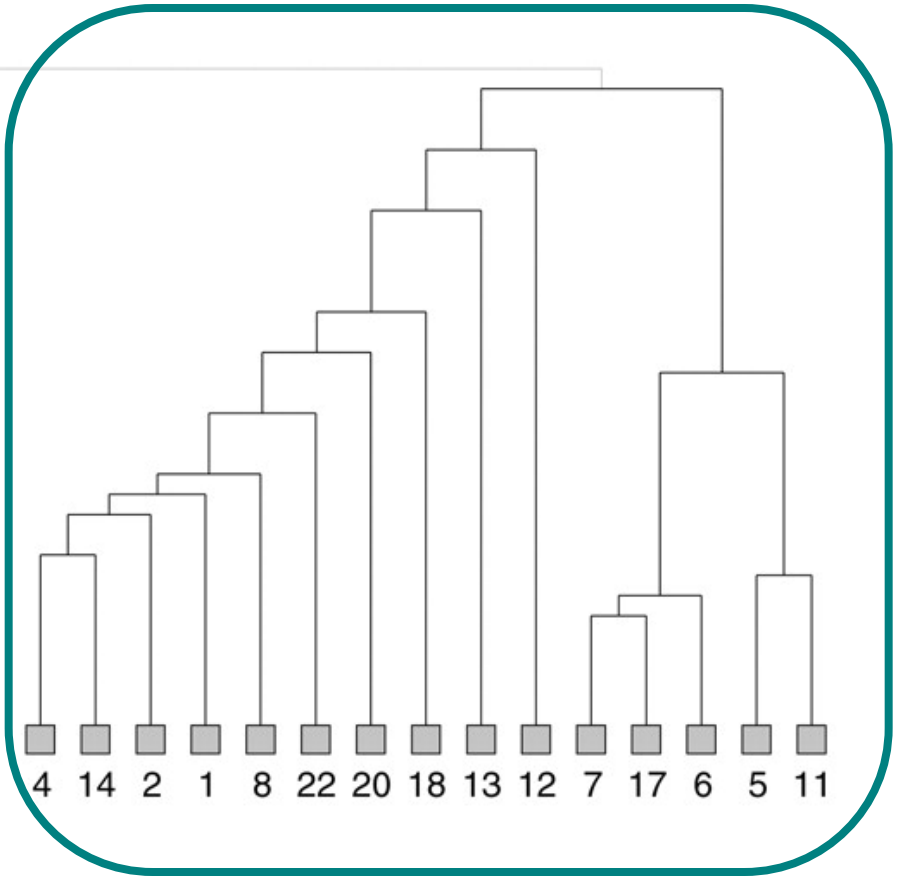




(a) *Step 1* (b) *Step 2*

betweenness clustering algorithm & the karate club data set





source: Girvan and Newman, PNAS June 11, 2002 99(12):7821-7826

Modularity

- Consider edges that fall within a community or between a community and the rest of the network
- Define modularity:

$$Q = \frac{1}{2m} \sum_{vw} \left[A_{vw} - \frac{k_v k_w}{2m} \right] \delta(c_v, c_w)$$

if vertices are in the same community

adjacency matrix

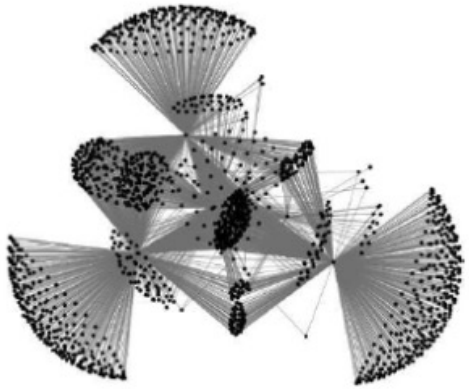
probability of an edge between two vertices is proportional to their degrees

- For a random network, Q = 0
  - the number of edges within a community is no different from what you would expect

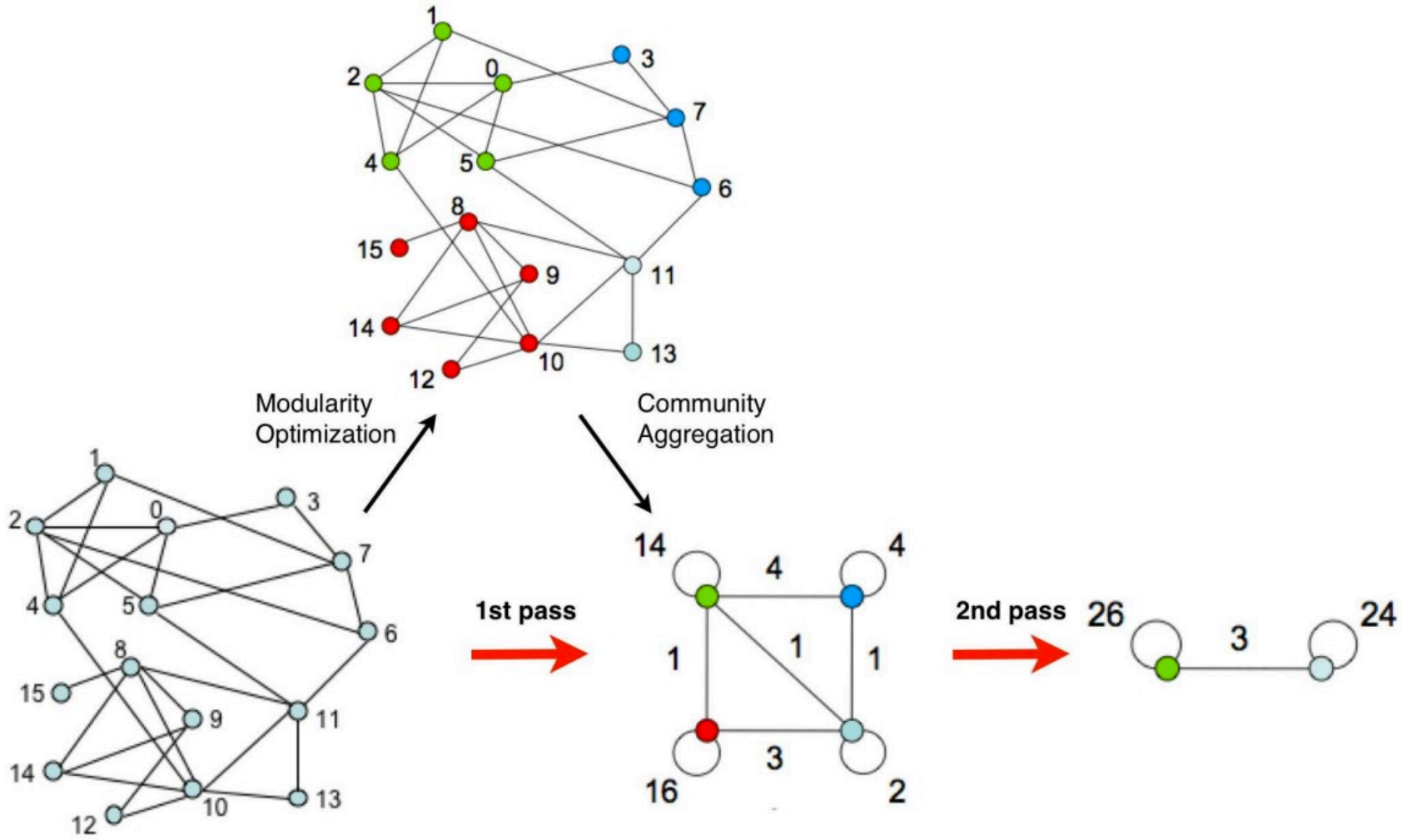
Finding community structure in very large networks
Authors: Aaron Clauset, M. E. J. Newman, Cristopher Moore 2004

Modularity

- Algorithm
  - start with all vertices as isolates
  - follow a greedy strategy:
    - successively join clusters with the greatest increase ΔQ in modularity
    - stop when the maximum possible $\Delta Q <= 0$ from joining any two
  - successfully used to find community structure in a graph with > 400,000 nodes with > 2 million edges
    - Amazon's people who bought this also bought that…
  - alternatives to achieving optimum ΔQ:
    - simulated annealing rather than greedy search



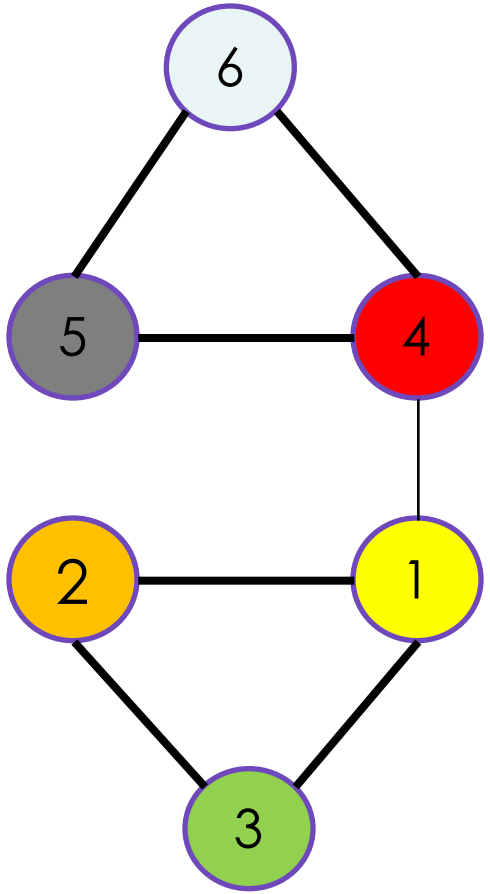
Louvain algorithm (Modularity optimization)



Louvain Algorithm

Maximize Modularity

$$Q = \frac{1}{2m}\Sigma_{ij}\left[A_{ij} - \frac{k_i k_j}{2m}\right]\delta(c_i, c_j)$$



Σ_{in} = sum of all the weights of the links inside the community ***i*** is moving into

Σ_{tot} = sum of all the weights of the links to the community

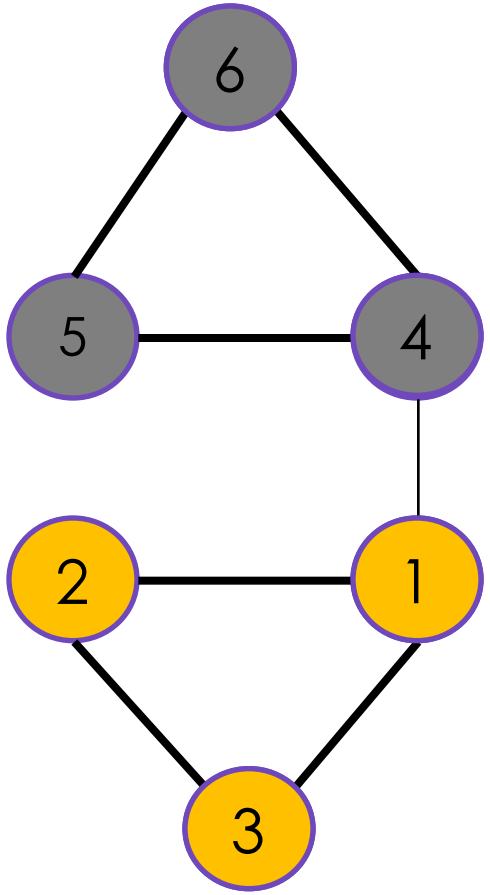
Change in modularity for node i,
if moved to neighbor community j

$$\Delta Q = \left[\frac{\Sigma_{in} + k_{i,in}}{2m} - \left(\frac{\Sigma_{tot} + k_i}{2m}\right)^2\right] - \left[\frac{\Sigma_{in}}{2m} - \left(\frac{\Sigma_{tot}}{2m}\right)^2 - \left(\frac{k_i}{2m}\right)^2\right]$$

Louvain Algorithm

Maximize Modularity

$$Q = \frac{1}{2m}\Sigma_{ij}\left[A_{ij} - \frac{k_i k_j}{2m}\right]\delta(c_i, c_j)$$



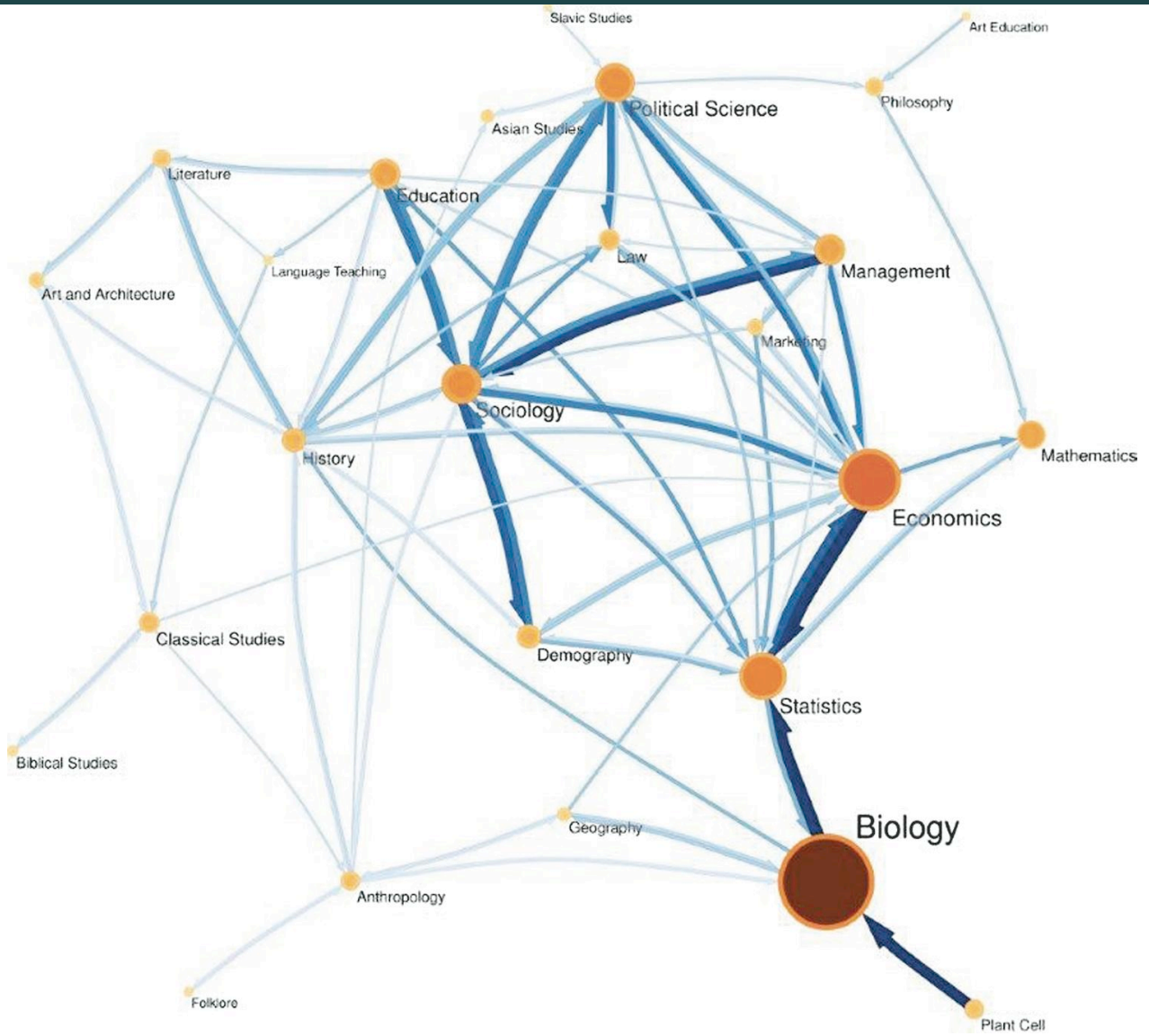
animation credit: Karthik Subbian

An information theoretic approach

- InfoMap, MapEquation
- How to most concisely describe a random walk on the network using huffman codes? Prefixes become communities…
- http://www.***pnas***.org/content/105/4/1118
  http://www.mapequation.org/mapgenerator/index.html

high-res maps of science

http://www.plosone.org/article/info%3Adoi%2F10.1371%2Fjournal.pone.0004803

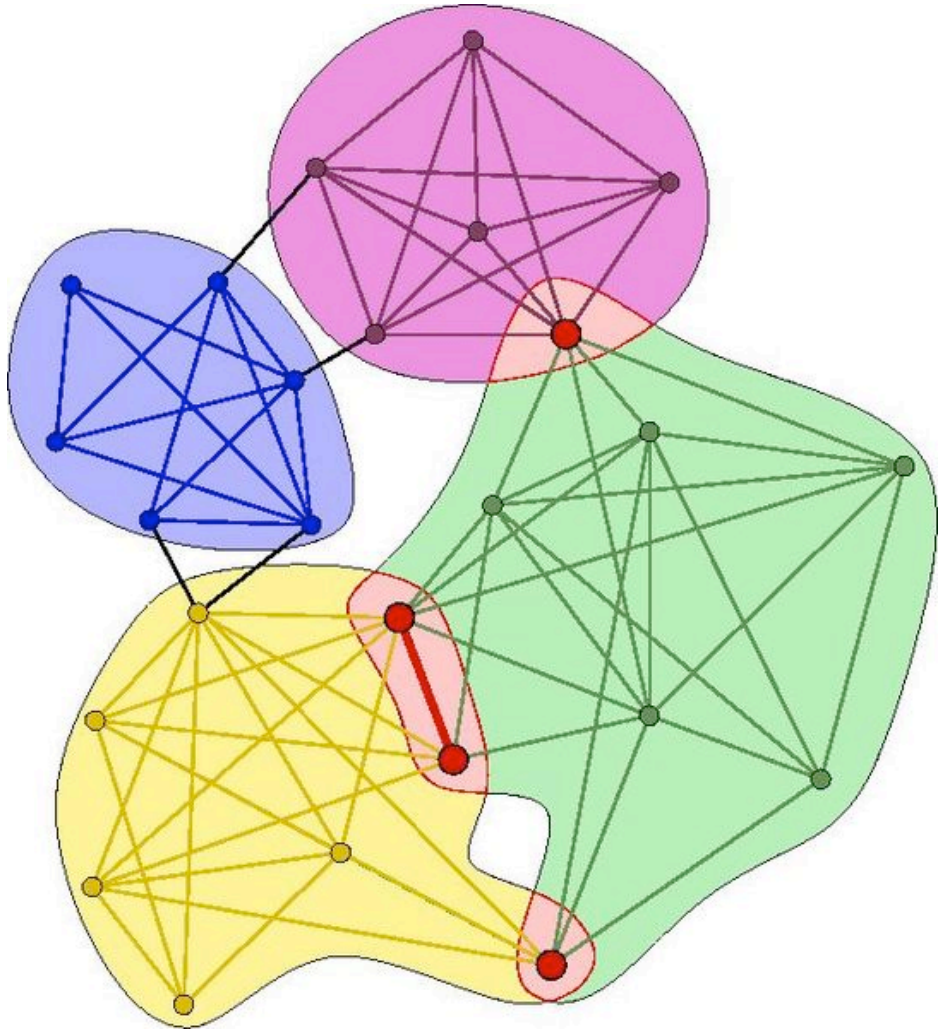


Performs best on benchmark tests of many community algorithms

Solution 1 for overlapping communities

- Clique finder
- http://cfinder.org

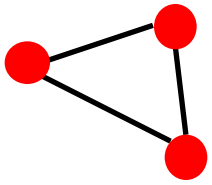
Uncovering the overlapping community structure of complex networks in nature and society G. Palla, I. Derényi, I. Farkas, and T. Vicsek: Nature 435, 814–818 (2005)



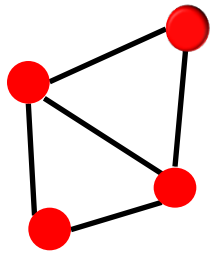
[Palla et al., '05]

Clique Percolation Method (CPM)

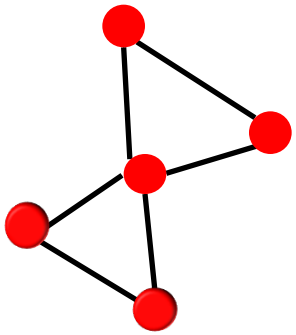
- **Two nodes belong to the same community if they can be connected through adjacent k-cliques:**
  - **k-clique:**
    - Fully connected graph on k nodes
  - **Adjacent k-cliques:**
    - overlap in $k-1$ nodes



3-clique

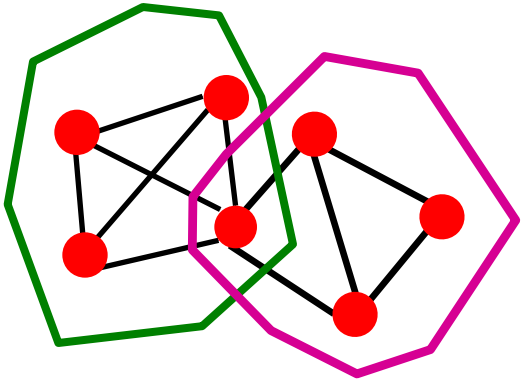


Adjacent 3-cliques



Non-adjacent 3-cliques

- **k-clique community**
  - Set of nodes that can be reached through a sequence of adjacent k-cliques

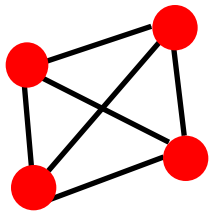


Two overlapping 3-clique communities

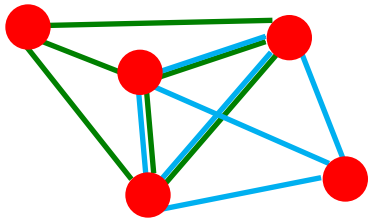
[Palla et al., '05]

Clique Percolation Method (CPM)

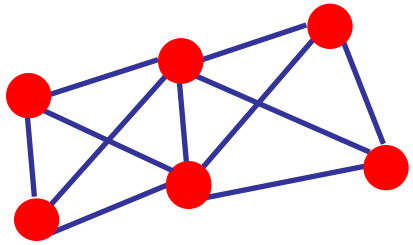
- Two nodes belong to the same community if they can be connected through adjacent k-cliques:



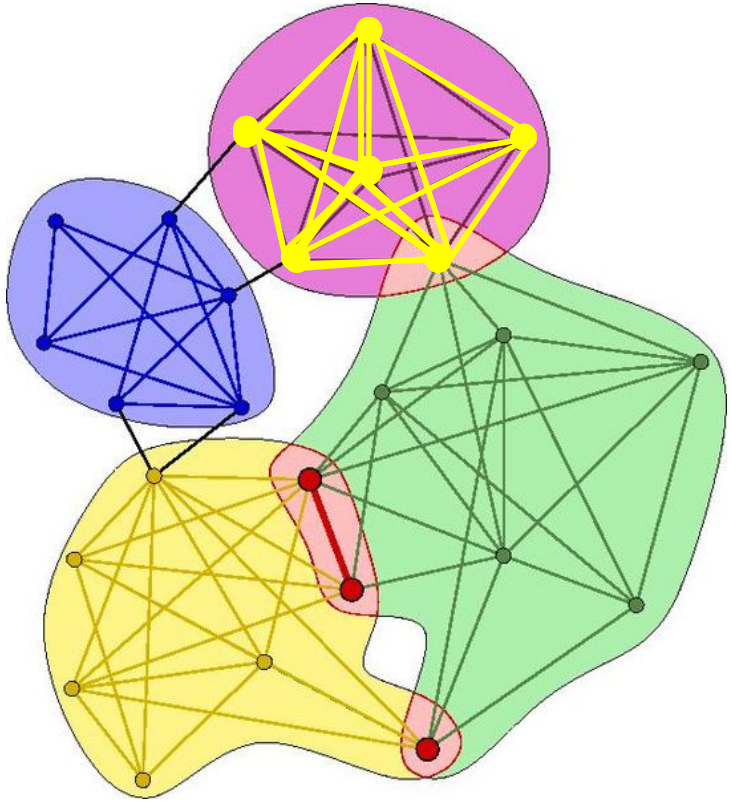
4-clique



Adjacent 4-cliques



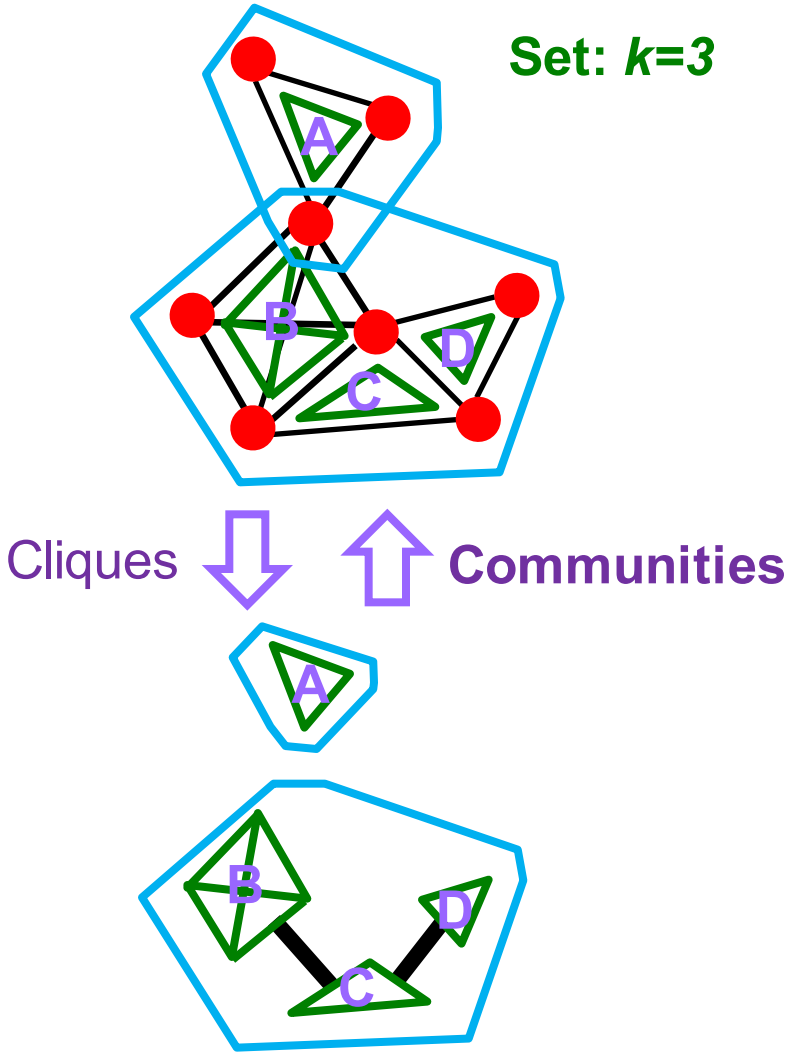
Non-adjacent 4-cliques



Communities for k=4

CPM: Steps

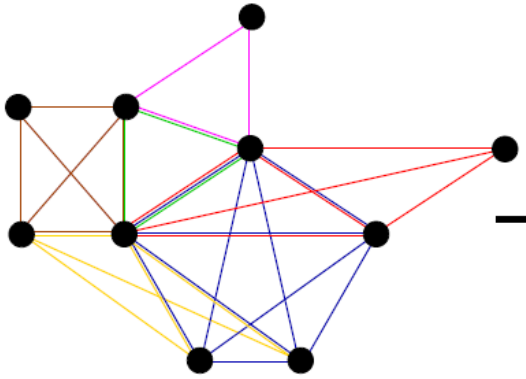
- **Clique Percolation Method:**
  - **Find maximal-cliques**
    - Def: Clique is maximal if no superset is a clique
  - **Clique overlap super-graph:**
    - Each clique is a super-node
    - Connect two cliques if they overlap in at least $k-1$ nodes
  - **Communities:**
    - Connected components of the clique overlap matrix



- **How to set k?**
  - Set k so that we get the "richest" (most widely distributed cluster sizes) community structure

CPM method: Example

- Start with graph
- Find maximal cliques
- Create clique overlap matrix
- Threshold the matrix at value ***k-1***
  - If $a_{ij} < k - 1$ *set* 0
- Communities are the connected components of the thresholded matrix



(1) Graph

Cliques

Overlap size

Cliques

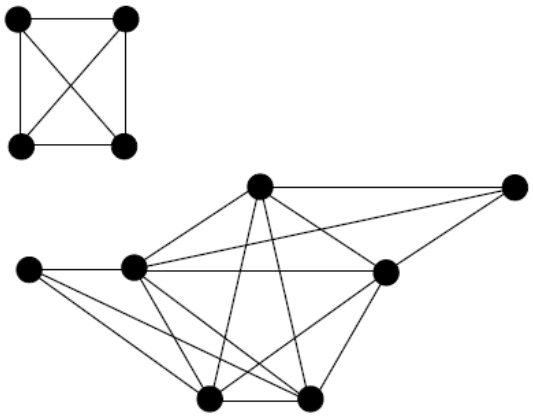
	5	3	2	1	3	1
	3	4	2	1	1	1
	2	2	3	2	1	2
	1	1	2	3	0	1
	3	1	1	0	4	2
	1	1	2	1	2	4

(2) Clique overlap matrix

k=4

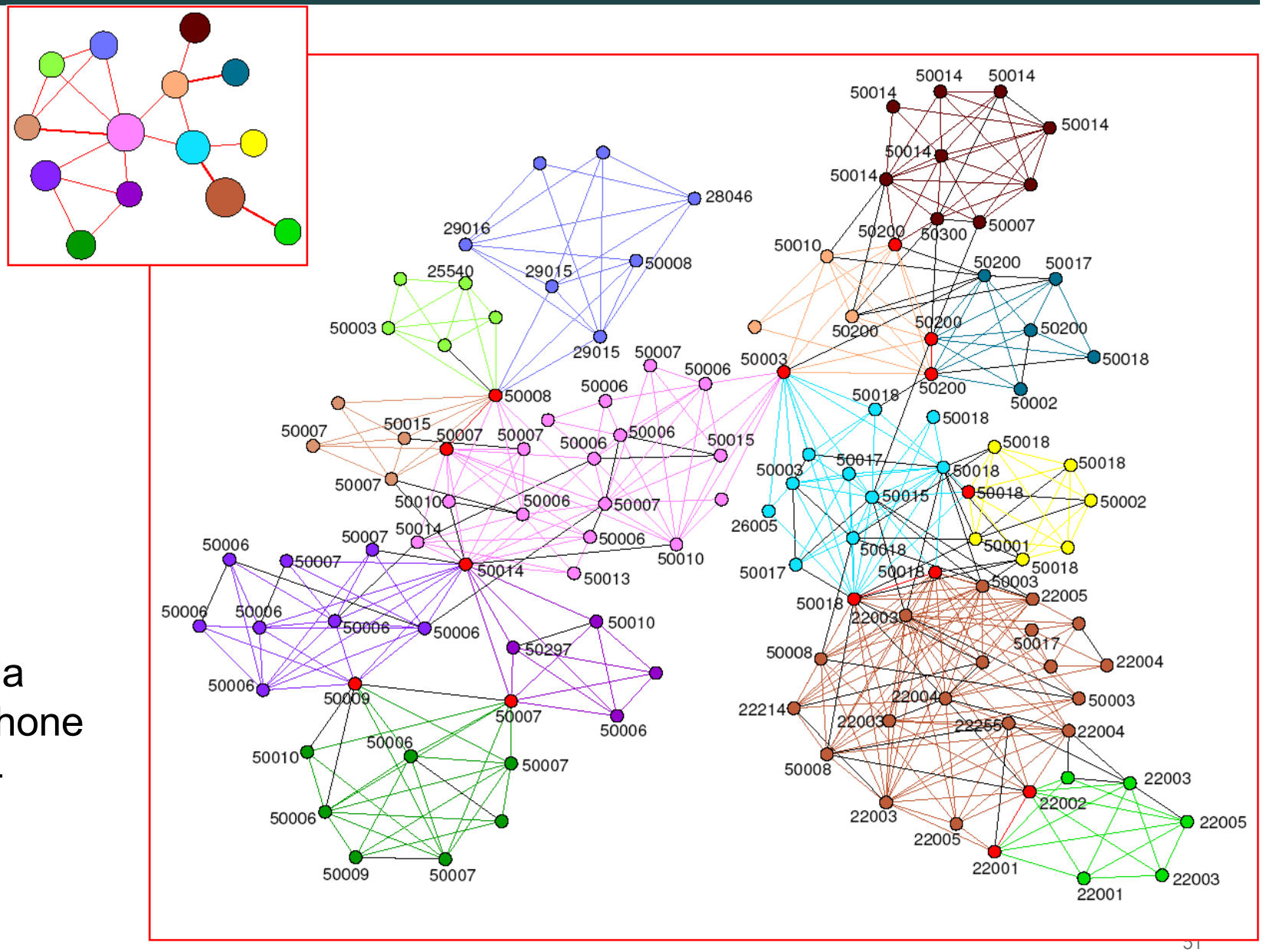
	1	1	0	0	1	0
	1	1	0	0	0	0
	0	0	0	0	0	0
	0	0	0	0	0	0
	1	0	0	0	1	0
	0	0	0	0	0	1

(3) Thresholded matrix at 3



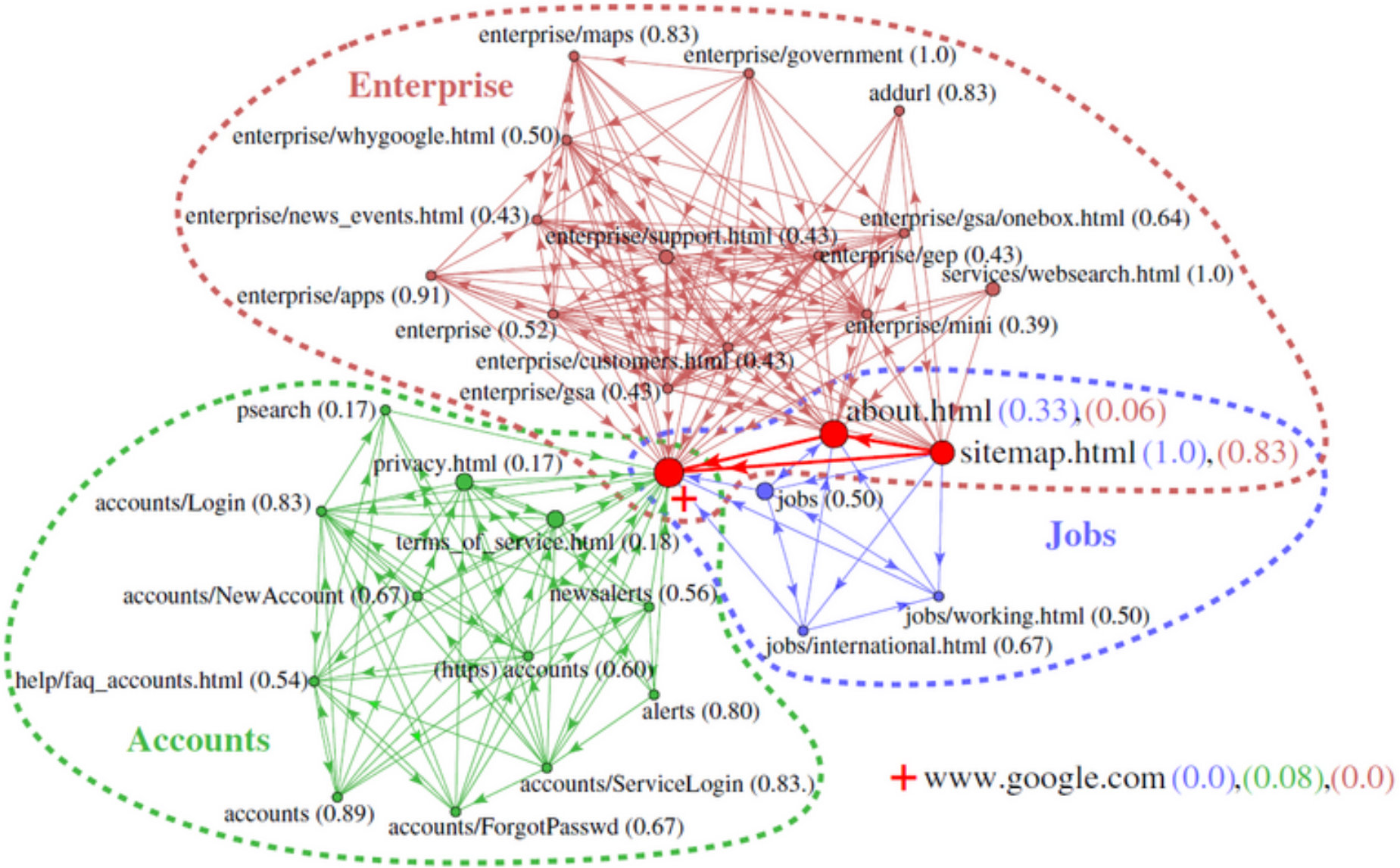
(4) Communities (connected components)

Example: Phone-Call Network



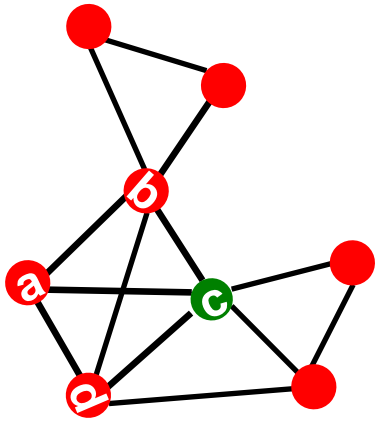
Communities in a "tiny" part of a phone call network of 4 million users
[Palla et al., '07]

Example: Website



How to Find Maximal Cliques?

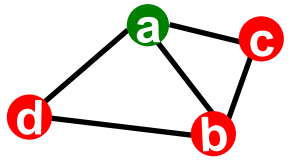
- **No nice way, hard combinatorial problem**
- **Maximal clique:** Clique that can't be extended
  - $\{a, b, c\}$ is a clique but not maximal clique
  - $\{a, b, c, d\}$ is maximal clique



- **Algorithm:** Sketch
  - Start with a seed node
  - Expand the clique around the seed
  - Once the clique cannot be further expanded we found the maximal clique
- **Note:**
  - This will generate the same clique multiple times

How to Find Maximal Cliques?

- Start with a seed vertex a
- **Goal:** Find the max clique Q that a belongs to
  - **Observation:**
    - If some x belongs to Q then it is a neighbor of a
      - **Why?** If $a, x \in Q$ but edge (a, x) does not exist, Q is not a clique!
- **Recursive algorithm:**
  - Q … current clique
  - R … candidate vertices to expand the clique to



- **Example:** Start with a and expand around it

Q=

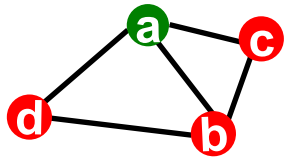
R=

Steps of the recursive algorithm

$\Gamma(u)$ … neighbor set of u

How to Find Maximal Cliques?

- Start with a seed vertex a
- **Goal:** Find the max clique Q that a belongs to
  - **Observation:**
    - If some x belongs to Q then it is a neighbor of a
      - **Why?** If $a, x \in Q$ but edge (a, x) does not exist, Q is not a clique!



- **Recursive algorithm:**
  - Q … current clique
  - R … candidate vertices to expand the clique to
- **Example:** Start with a and expand around it

Q=	{a}	{a,b}	{a,b,c}	bktrack	{a,b,d}
R=	{b,c,d}	{b,c,d}	{d}∩Γ(c)={}		{c}∩Γ(d)={}
		∩Γ(b)={c,d}			

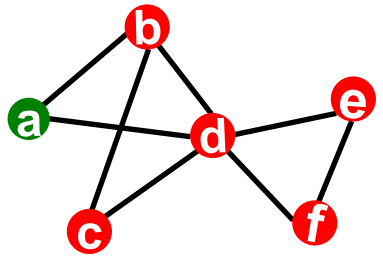
Steps of the recursive algorithm

Γ(u) … neighbor set of *u*

How to Find Maximal Cliques?

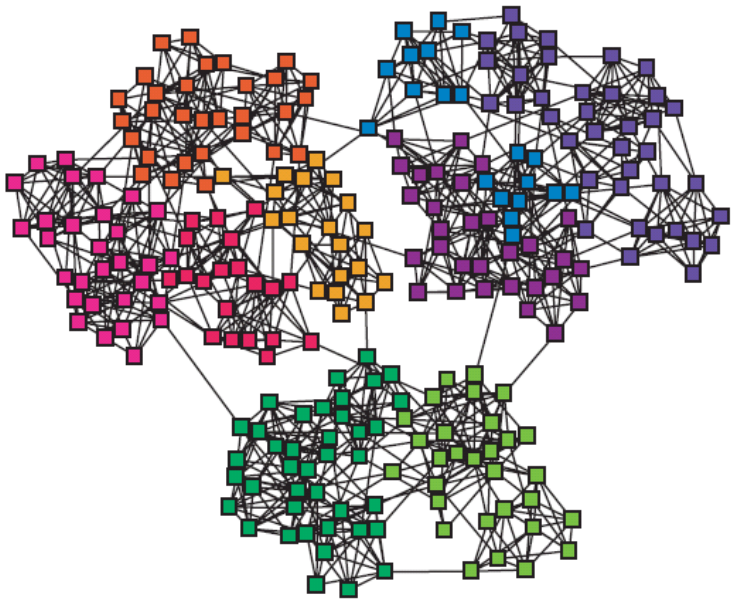
- Q … current clique
- R … candidate vertices

- **Expand(R,Q)**
  - **while** R ≠ {}
    - p = vertex in R
    - Q_p = Q ∪ {p}
    - R_p = R ∩ Γ(p)
    - **if** R_p ≠ {}: Expand(R_p,Q_p)
      else: output Q_p
    - R = R − {p}



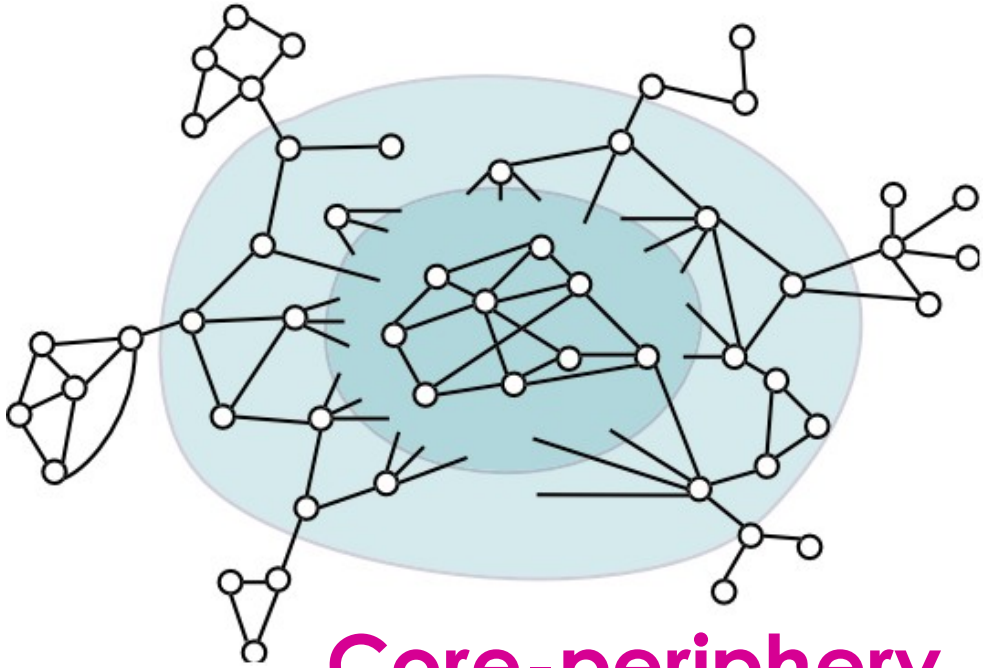
Network and Communities

- **How do we reconcile these two views?**
  (and still do community detection)



Community structure

vs.

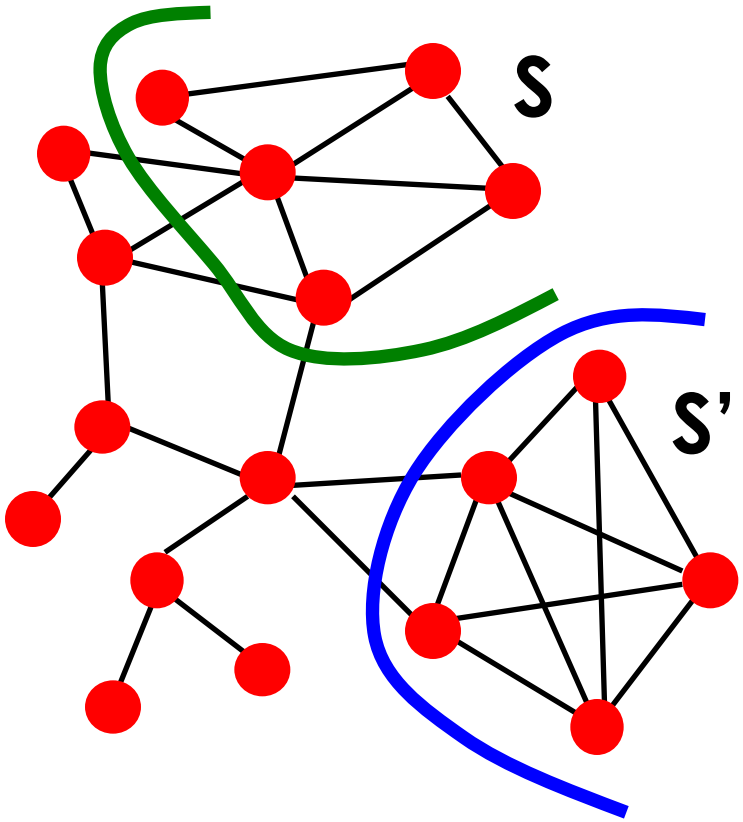


Core-periphery

Statistical Properties of Community Structure in Large Social and Information Networks by J. Leskovec, K. Lang, A. Dasgupta, M. Mahoney. *WWW'08*

Community Score

- **How community-like is a set of nodes?**
- **A good cluster *S* has**
  - Many edges internally
  - Few edges pointing outside
- What's a good metric: **Conductance**



$$\phi(S) = \frac{|\{(i,j) \in E; i \in S, j \notin S\}|}{\sum_{s \in S} d_s}$$

Small conductance corresponds to good clusters

(Note |S| < |V|/2)

Network Community Profile Plot

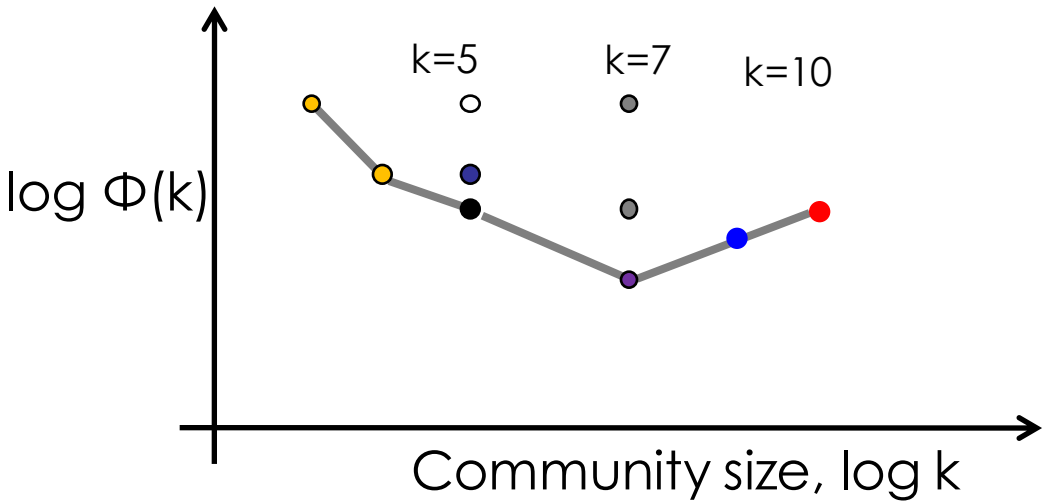
(Note |S| < |V

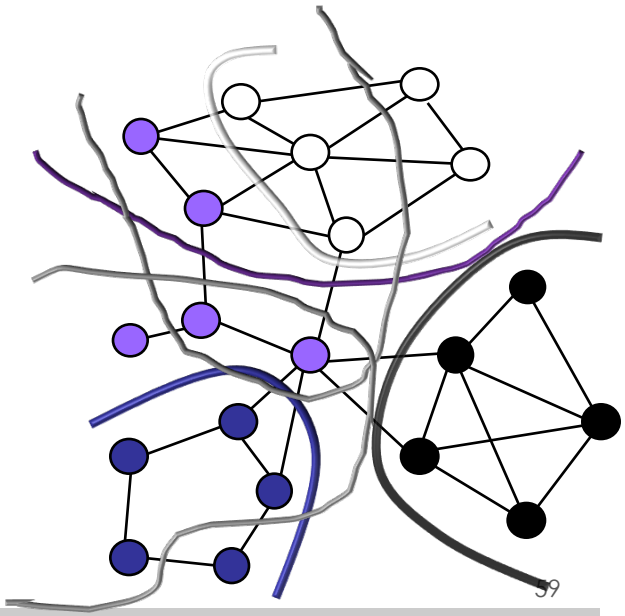
- Define:

Network community profile (**NCP**) plot

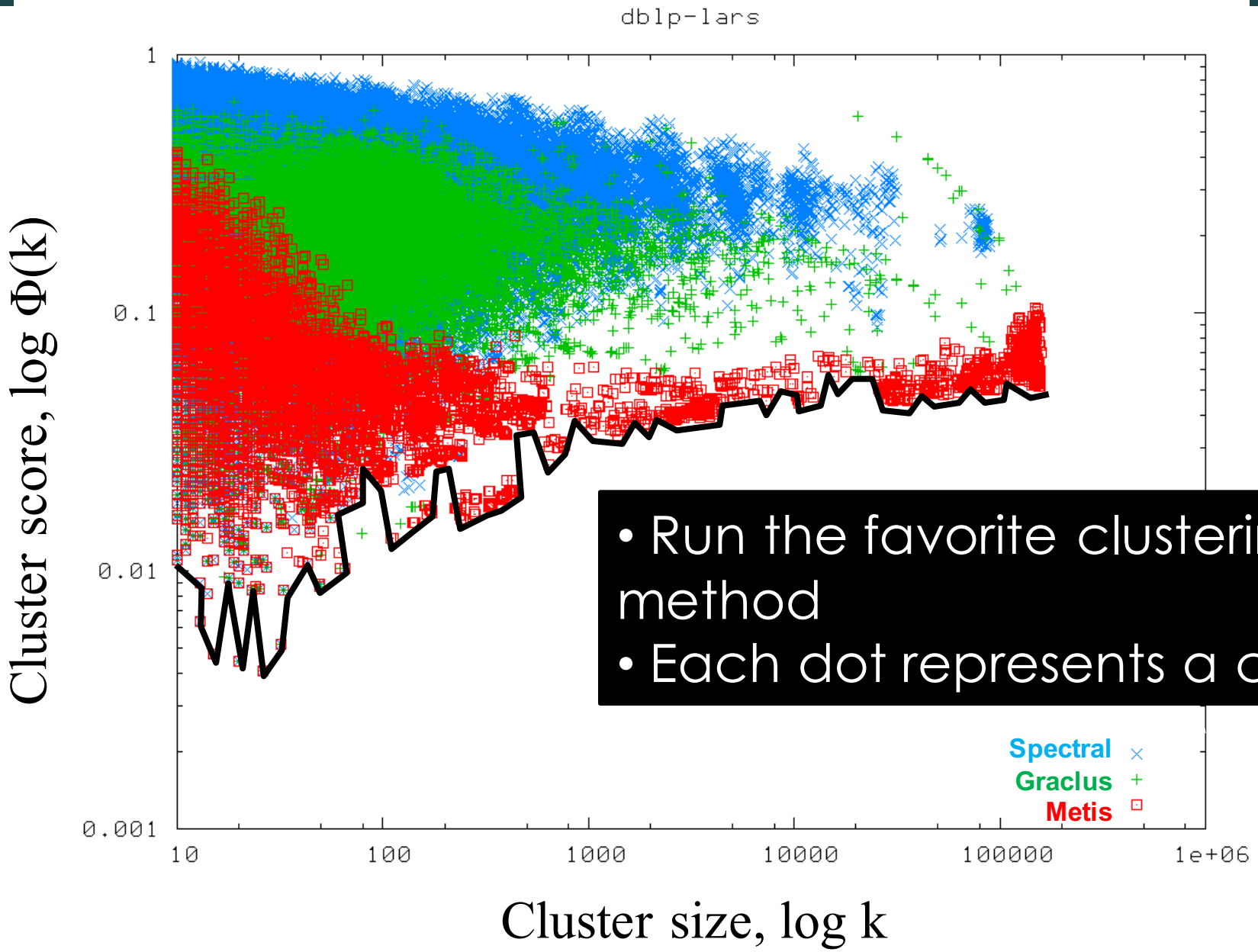
Plot the score of **best** community of size *k*

$$\Phi(k) = \min_{S \subset V, |S|=k} \phi(S)$$





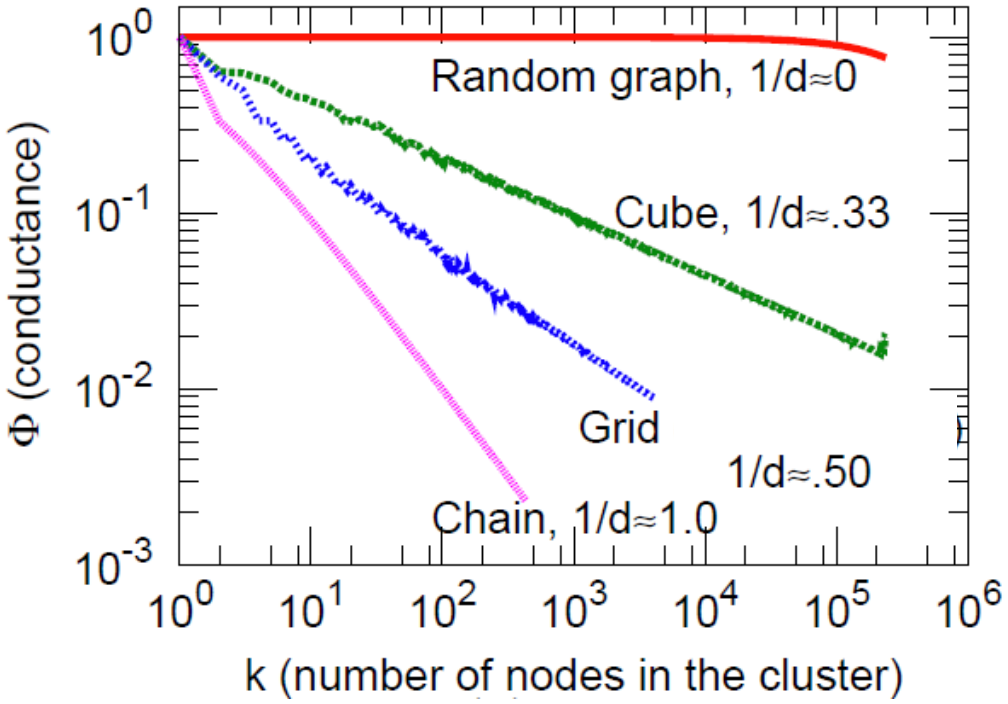
How to (Really) Compute NCP?



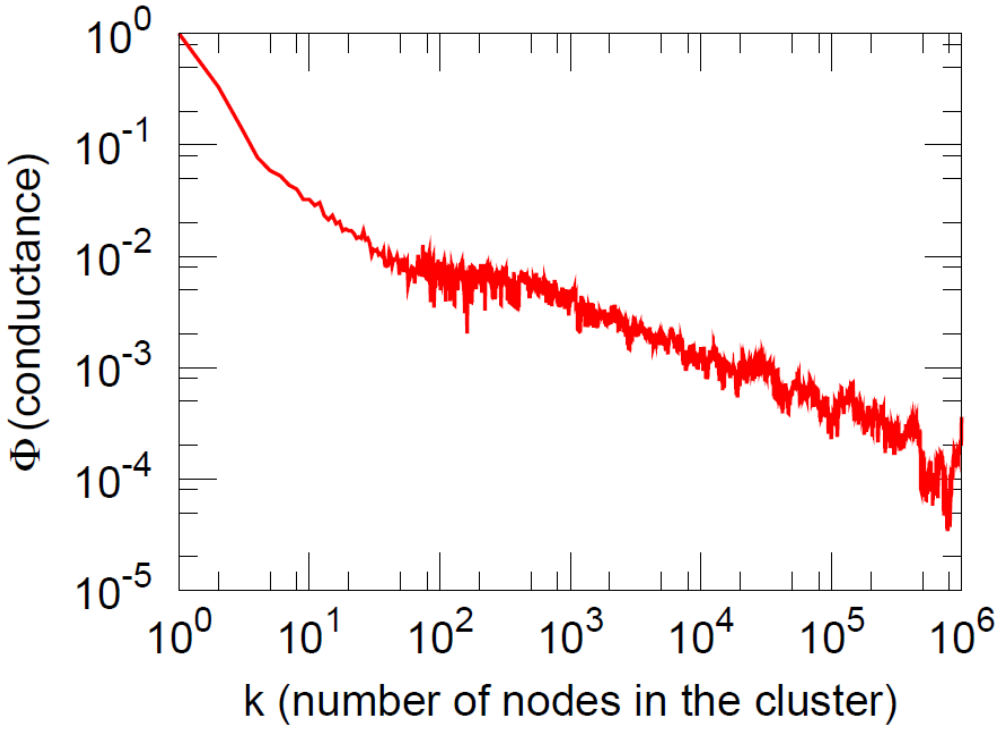
- Run the favorite clustering method
- Each dot represents a cluster

NCP Plot: Meshes

- **Meshes, grids, dense random graphs:**



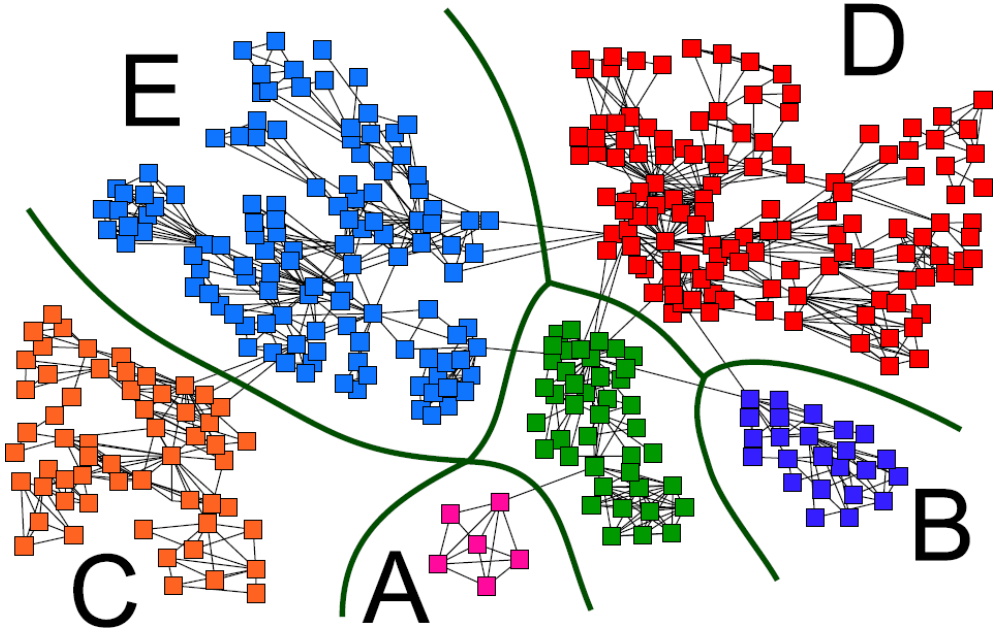
d-dimensional meshes

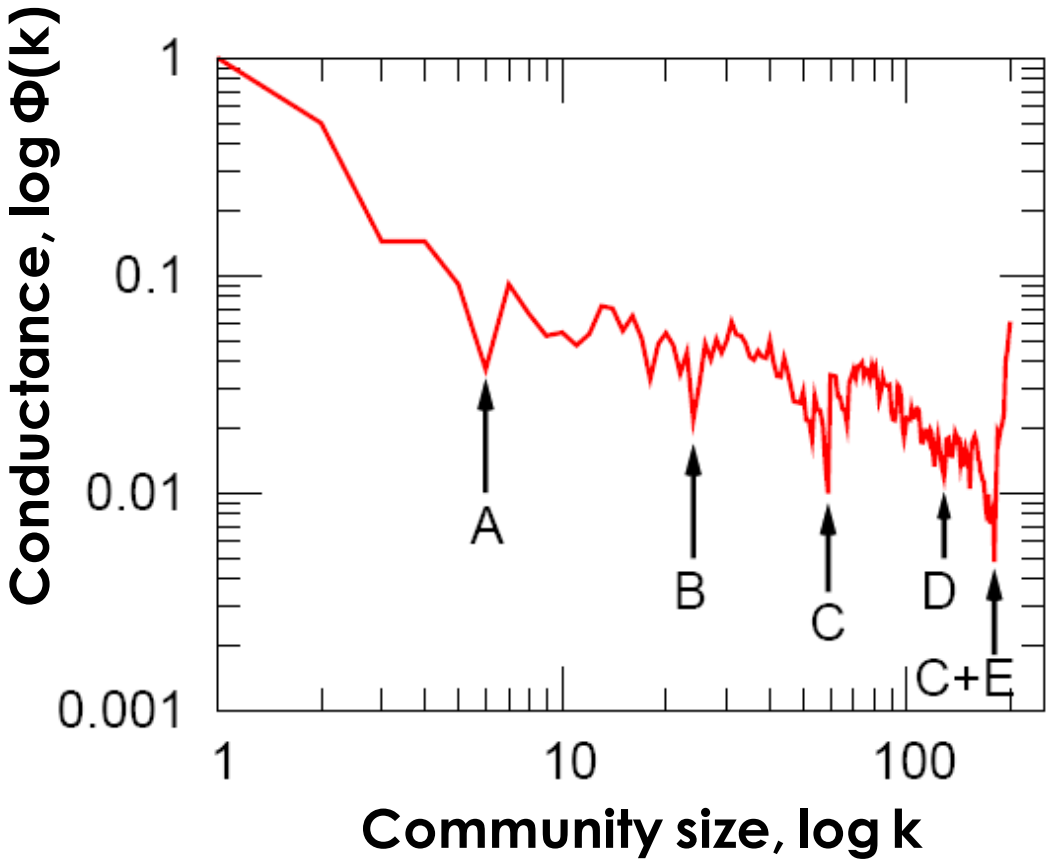


California road network

NCP plot: Network Science

- **Collaborations between scientists in networks**
  [Newman, 2005]



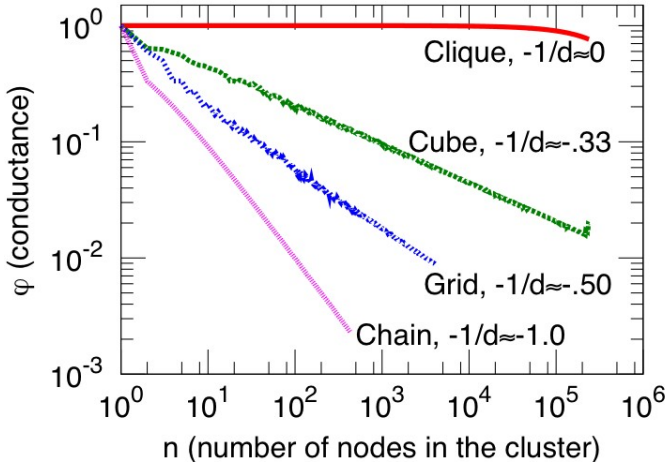


Dips in the conductance graph correspond to the "good" clusters we can visually detect

Natural Hypothesis

Natural hypothesis about NCP:

- NCP of real networks slopes downward
- Slope of the NCP corresponds to the "dimensionality" of the network

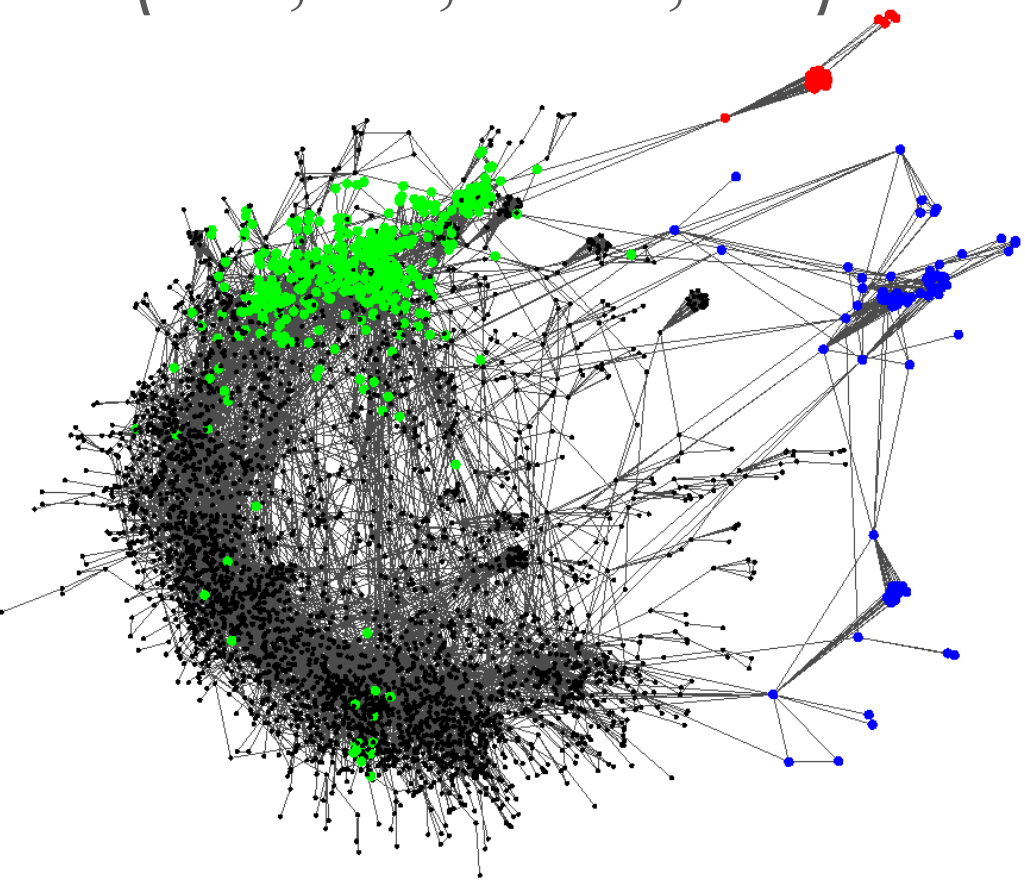


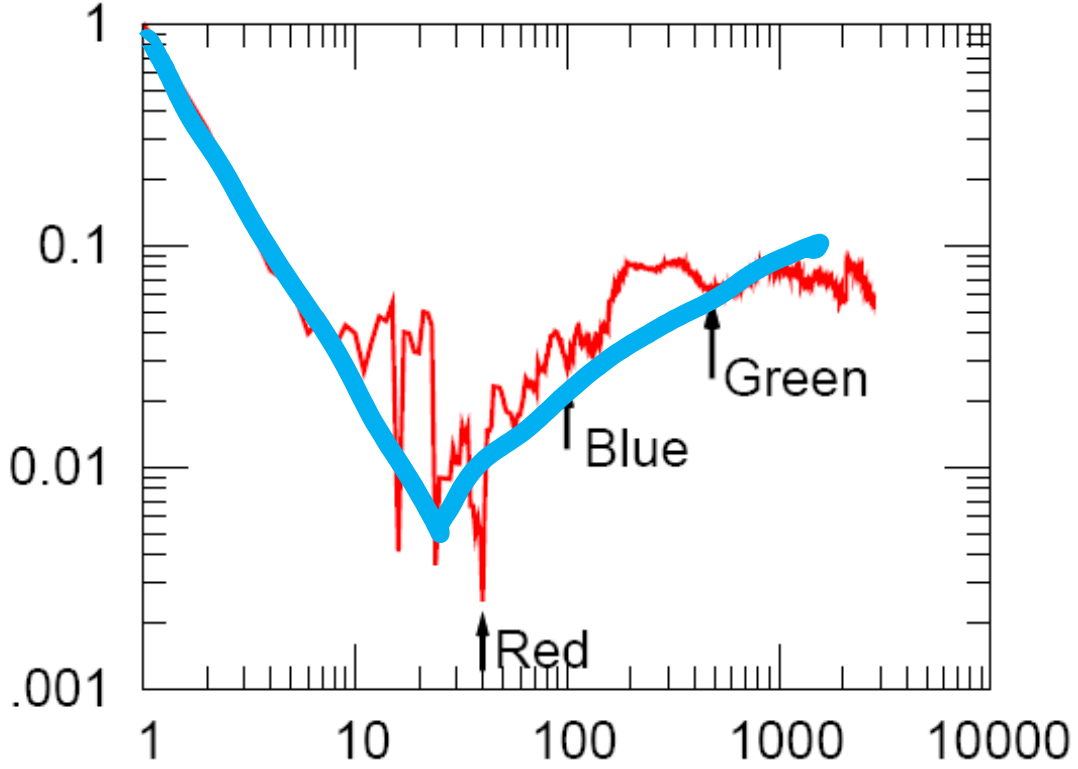
• Social nets	Nodes	Edges	Description
LIVEJOURNAL	4,843,953	42,845,684	Blog friendships [5]
EPINIONS	75,877	405,739	Trust network [28]
CA-DBLP	317,080	1,049,866	Co-authorship [5]
• Information (citation) networks			
CIT-HEP-TH	27,400	352,021	Arxiv hep-th [14]
AMAZONPROD	524,371	1,491,793	Amazon products [8]
• Web graphs			
WEB-GOOGLE	855,802	4,291,352	Google web graph
WEB-WT10G	1,458,316	6,225,033	TREC WT10G
• Bipartite affiliation (authors-to-papers) networks			
ATP-DBLP	615,678	944,456	DBLP [21]
ATM-IMDB	2,076,978	5,847,693	Actors-to-movies
• Internet networks			
ASSKITTER	1,719,037	12,814,089	Autonom. sys.
GNUTELLA	62,561	147,878	P2P network [29]

Large Networks: Very Different

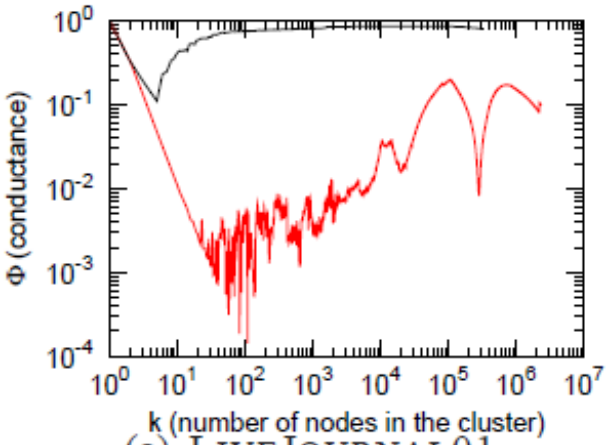
Typical example: General Relativity collaborations

(n=4,158, m=13,422)

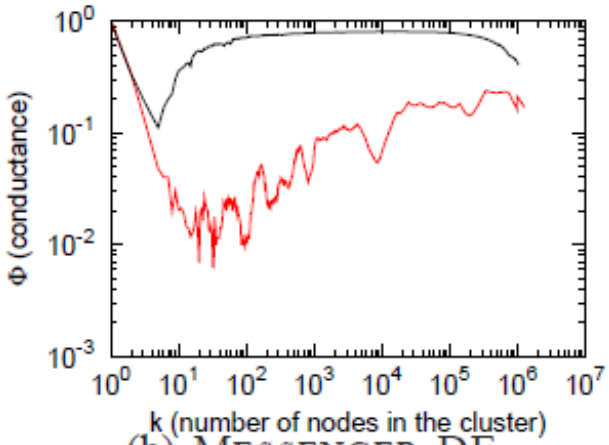




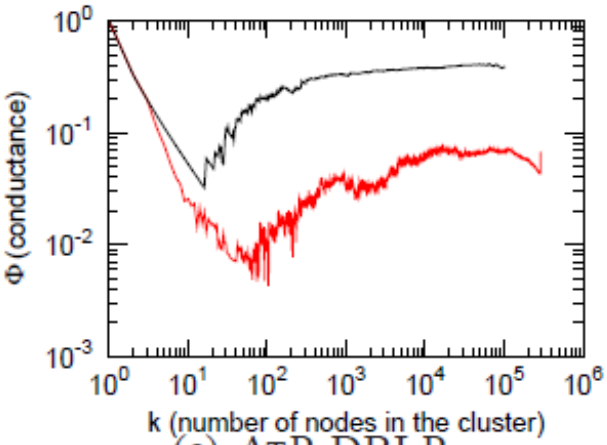
More NCP Plots of Networks



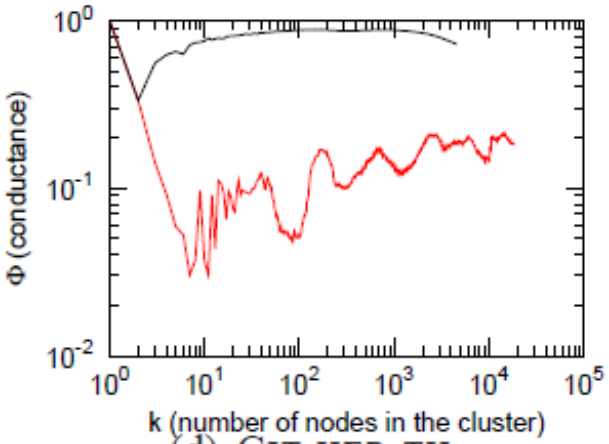
(a) LIVEJOURNAL01



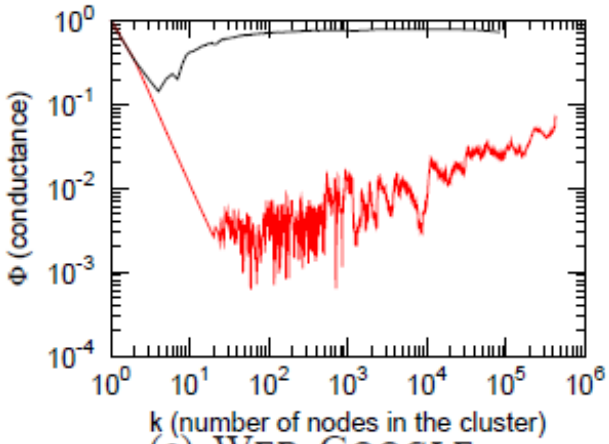
(b) MESSENGER-DE



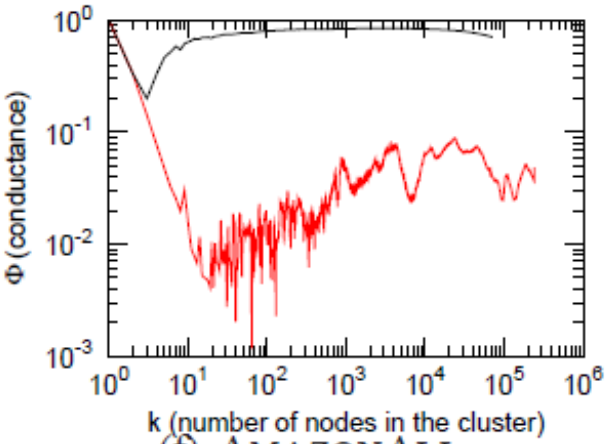
(c) ATP-DBLP



(d) CIT-HEP-TH



(e) WEB-GOOGLE

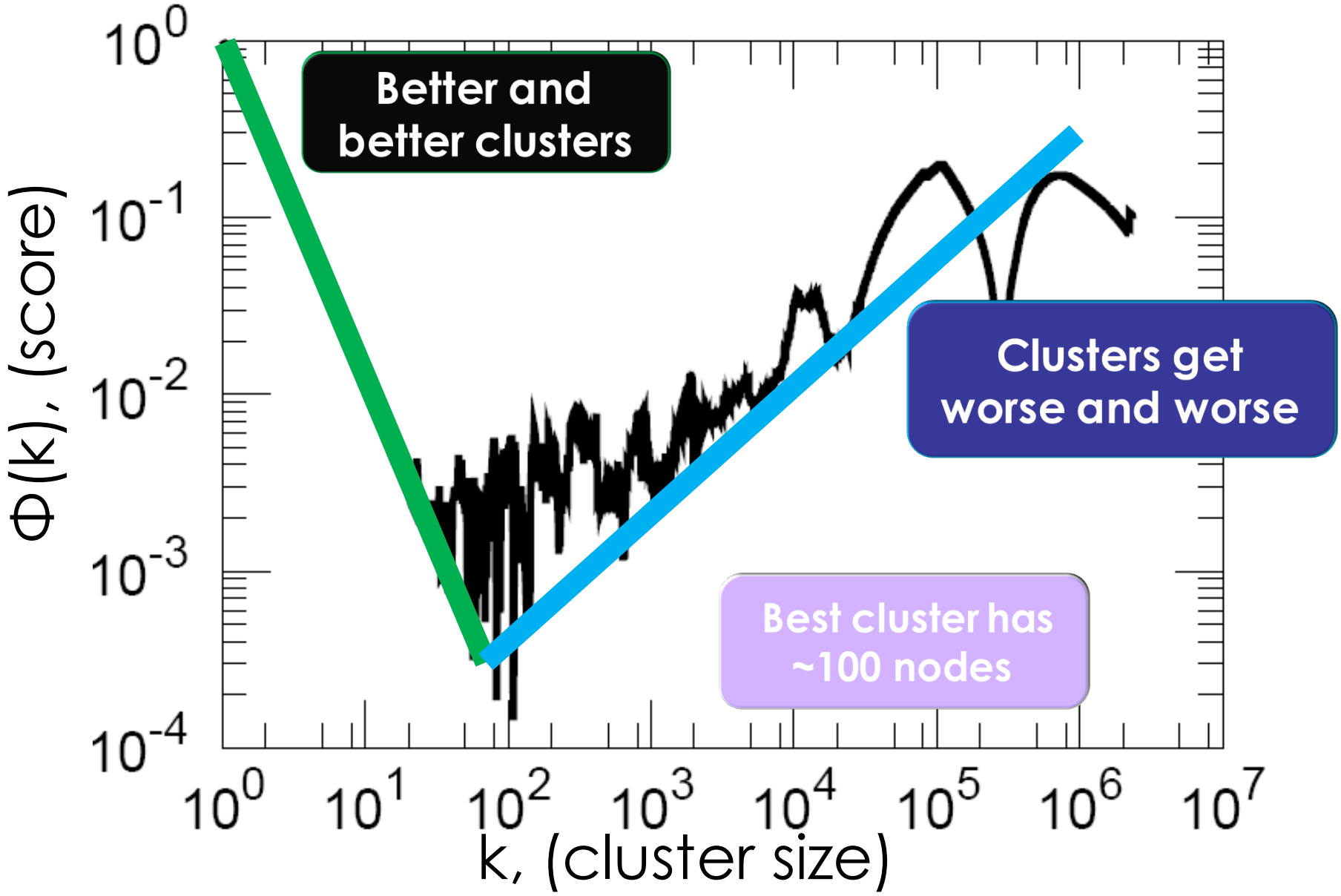


(f) AMAZONALL

-- Rewired graph

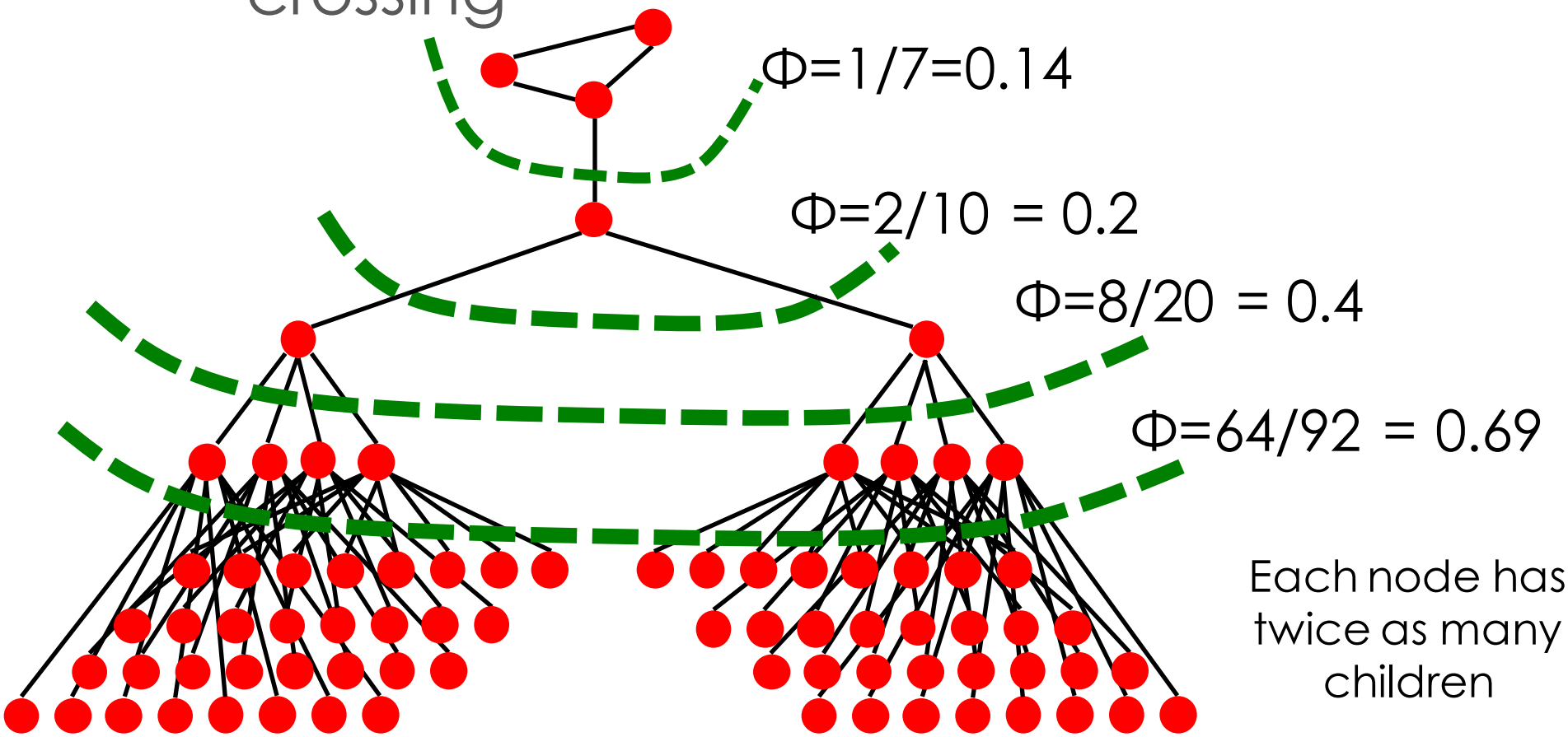
-- Real graph

NCP: LiveJournal (n=5m, m=42m)

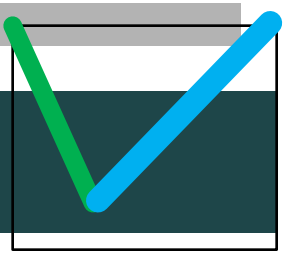


Explanation: The Upward Part

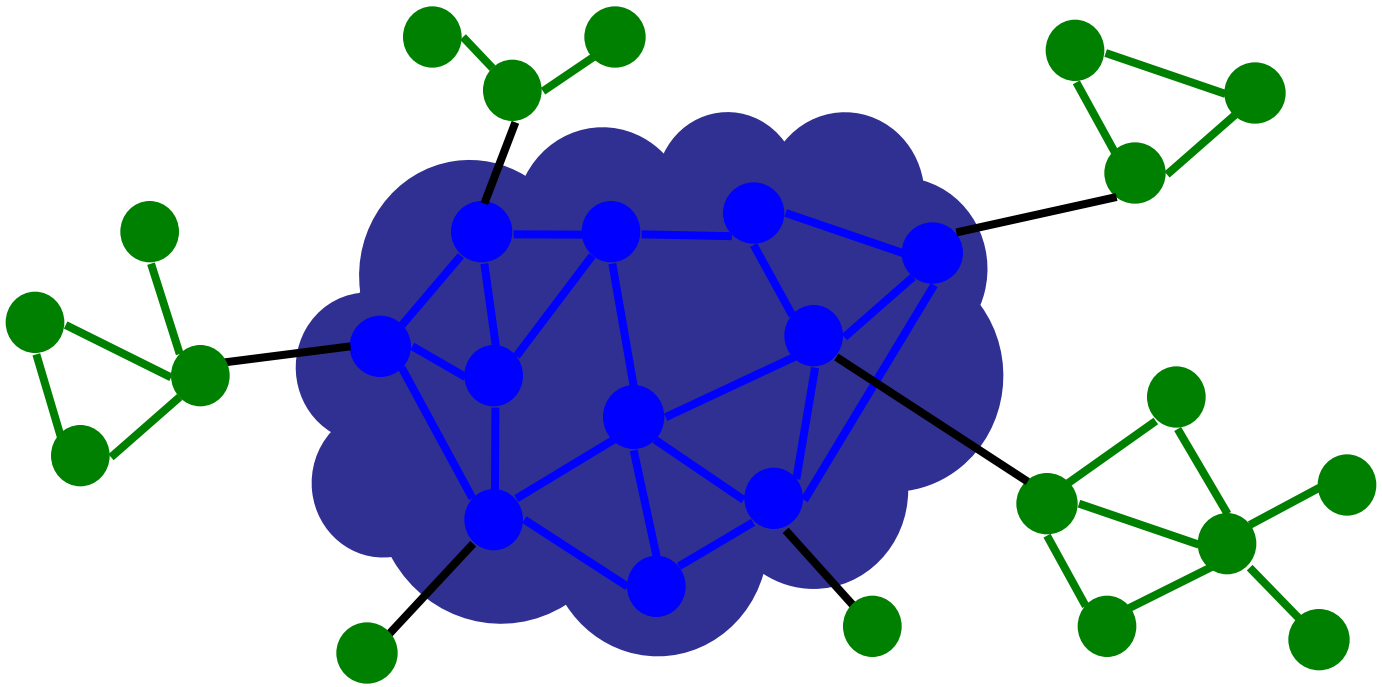
- As clusters grow the number of edges inside grows **slower** that the number crossing

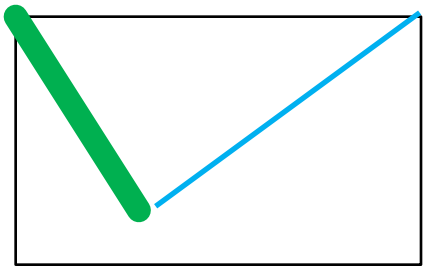


Explanation: Downward Part



- Empirically we note that **best clusters** (corresponding to **green nodes**) are **barely connected** to the network

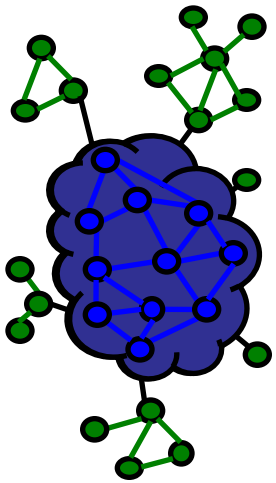


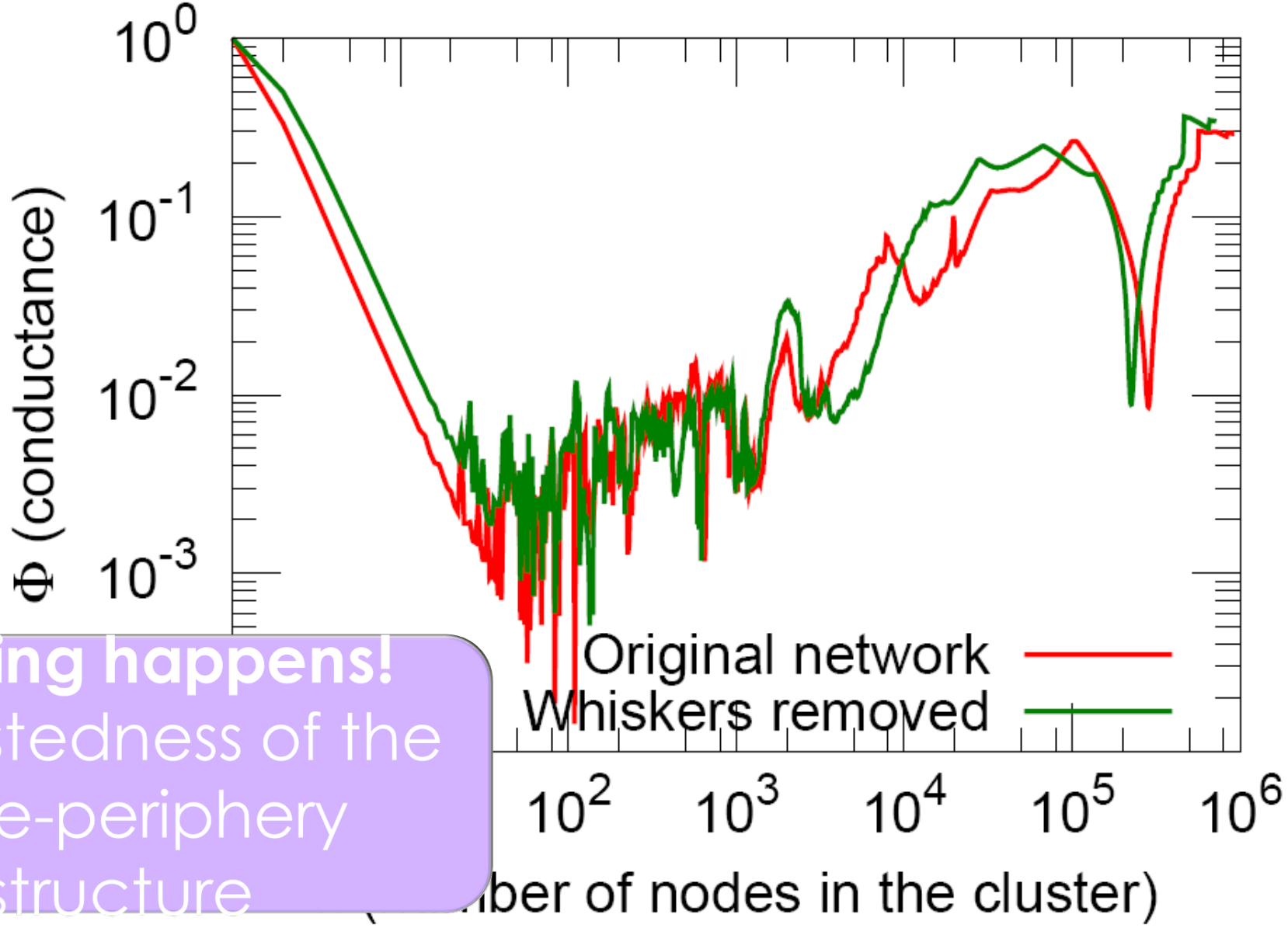


NCP plot

⇒ **Core-periphery structure**

What If We Remove Good Clusters?

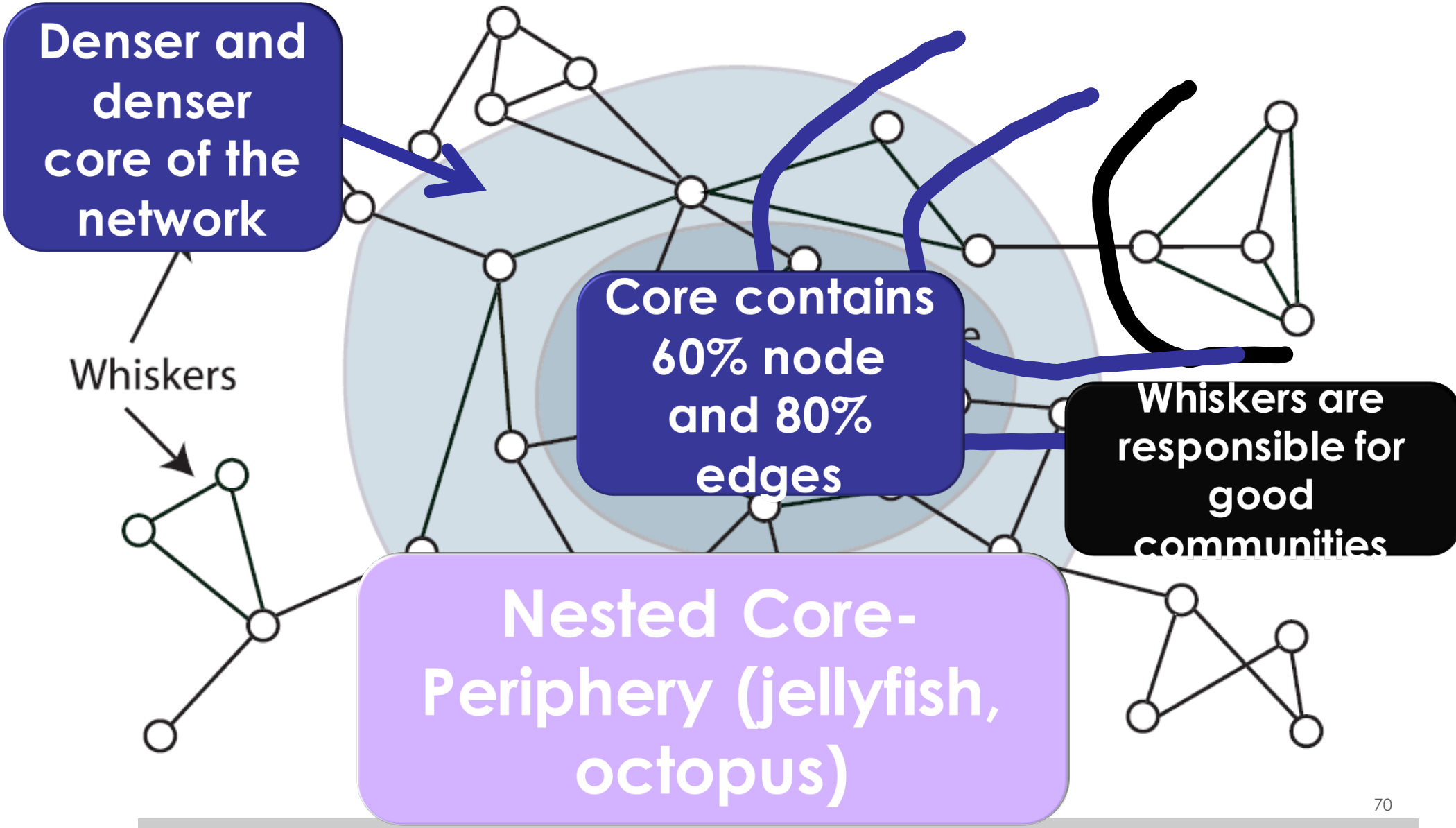




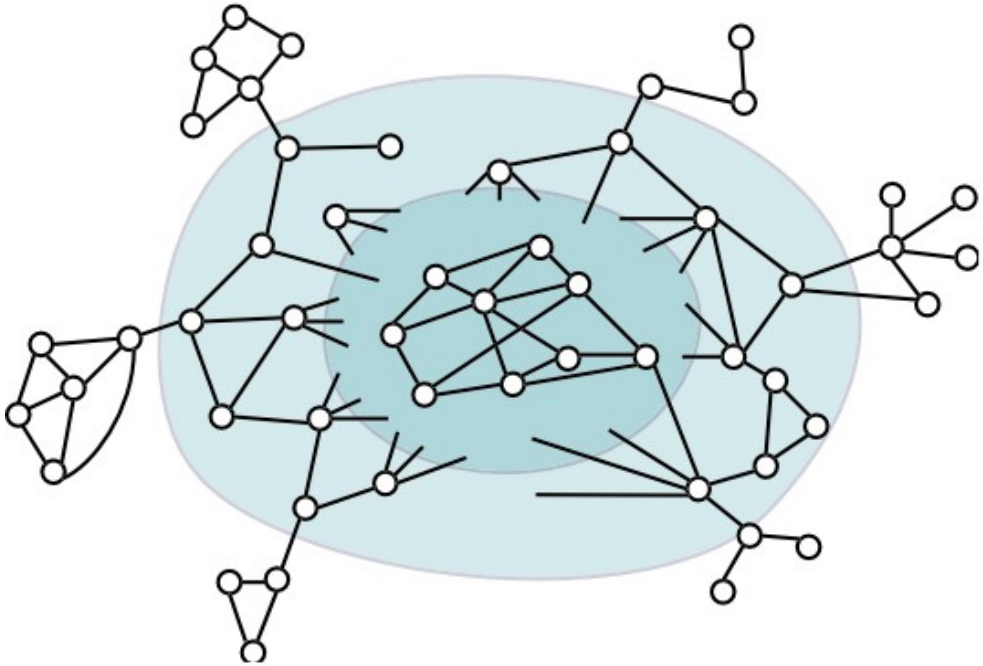
Nothing happens!

⇒ Nestedness of the core-periphery structure

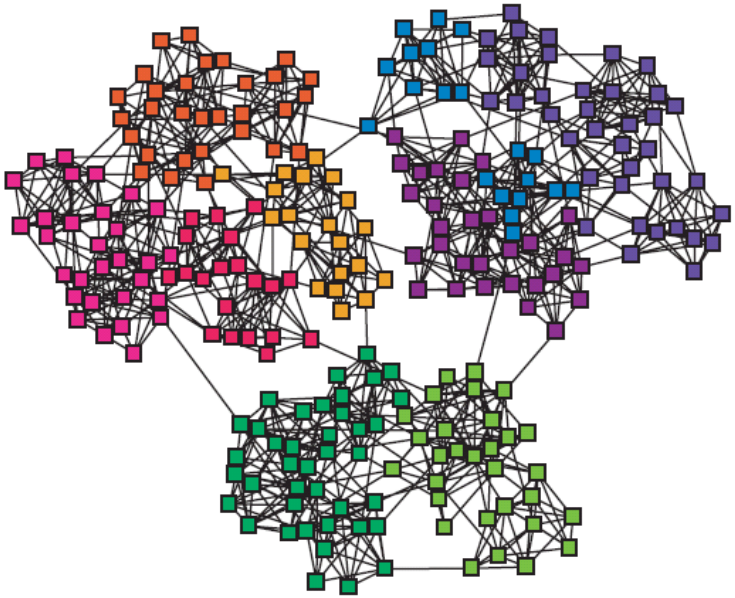
Suggested Network Structure



Reconciling the two views

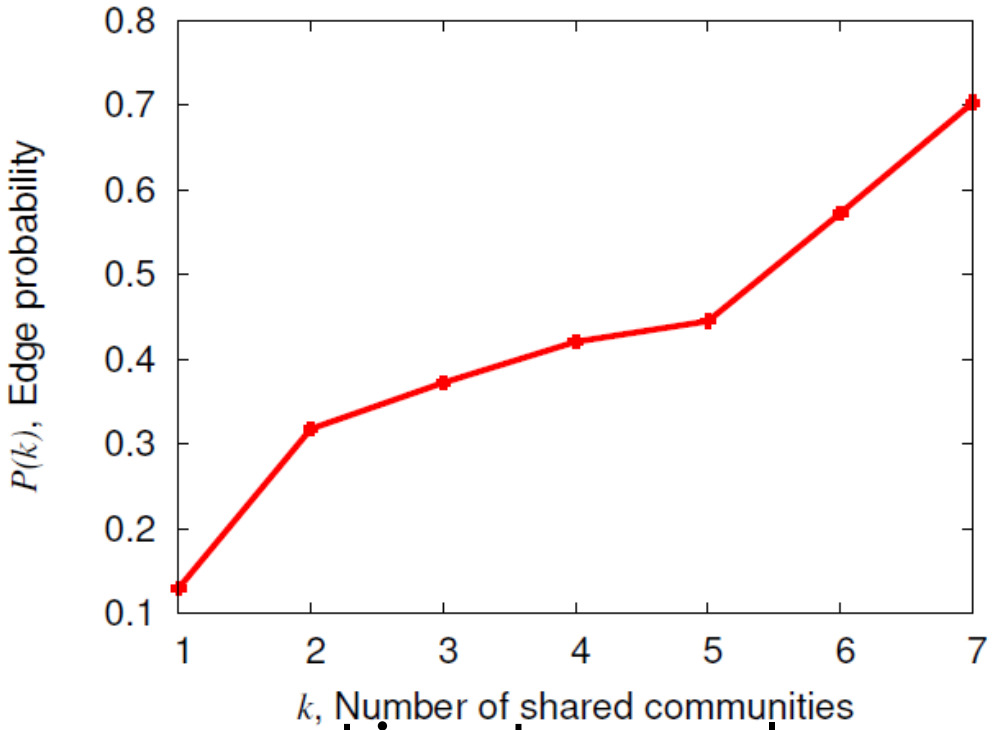


vs.

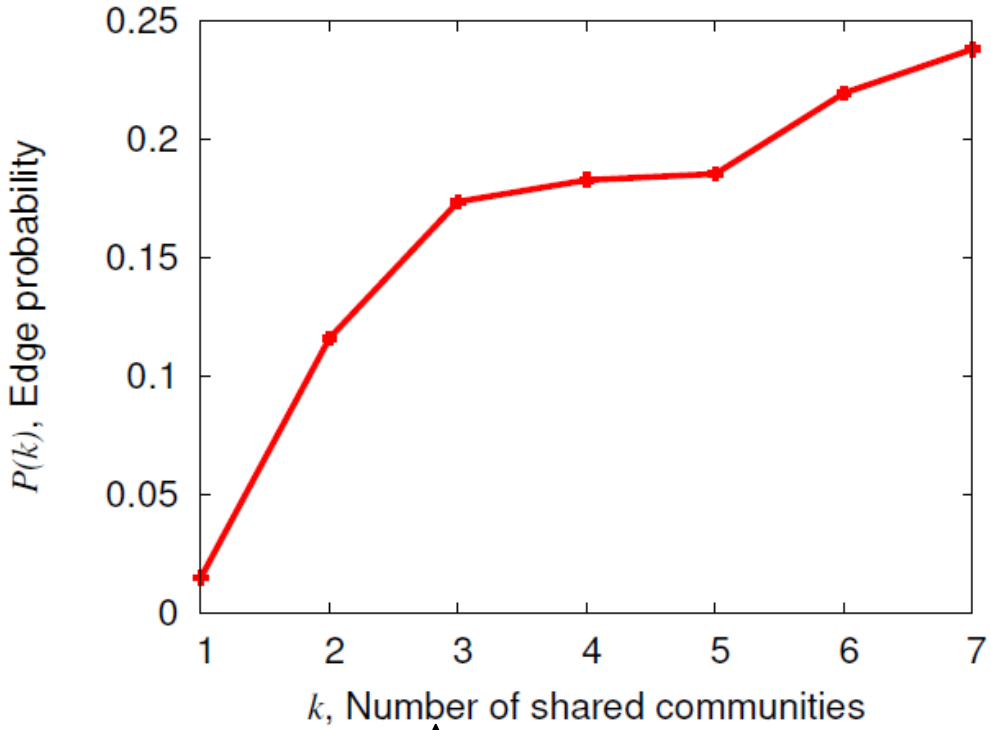


Ground-truth Communities

- Basic question: nodes u, v share k communities
- What's the edge probability?

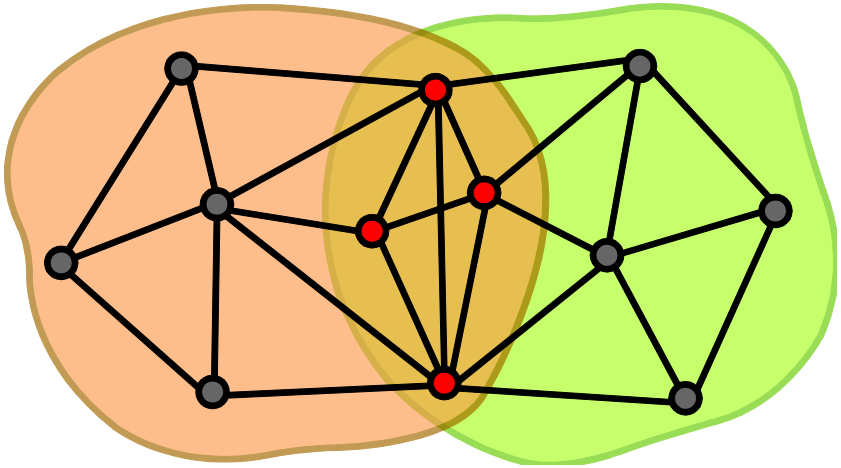


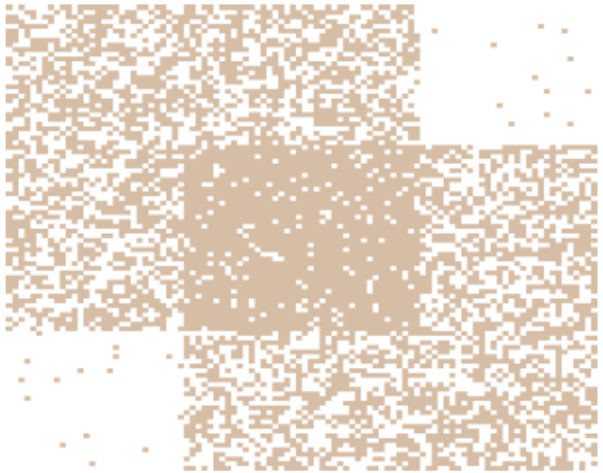
LiveJournal social network



Amazon product network

Edge density in overlaps is higher

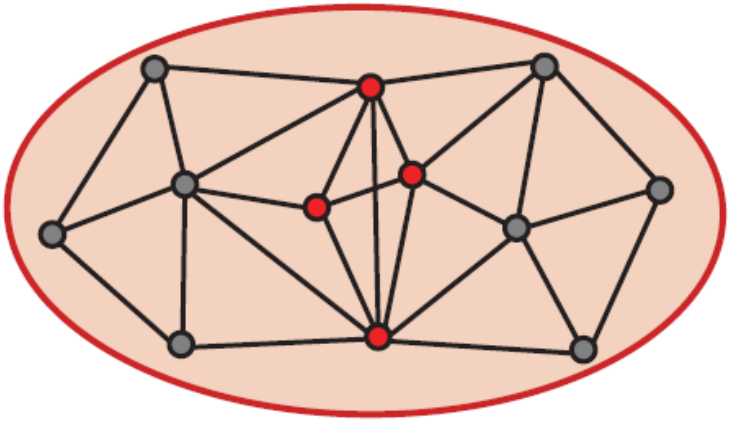


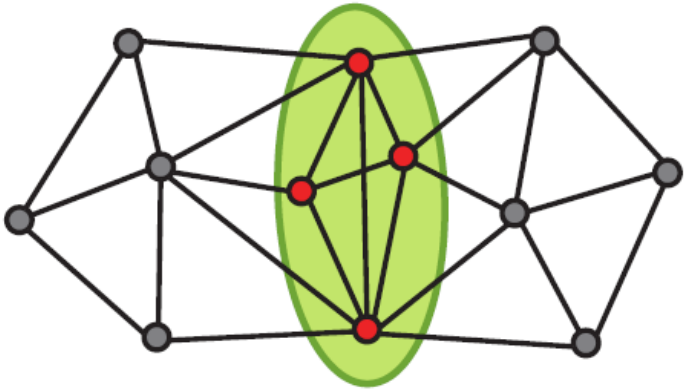


"The more different foci (communities) that two individuals share, the more likely it is that they will be tied" - S. Feld, 1981

Many Methods Fail

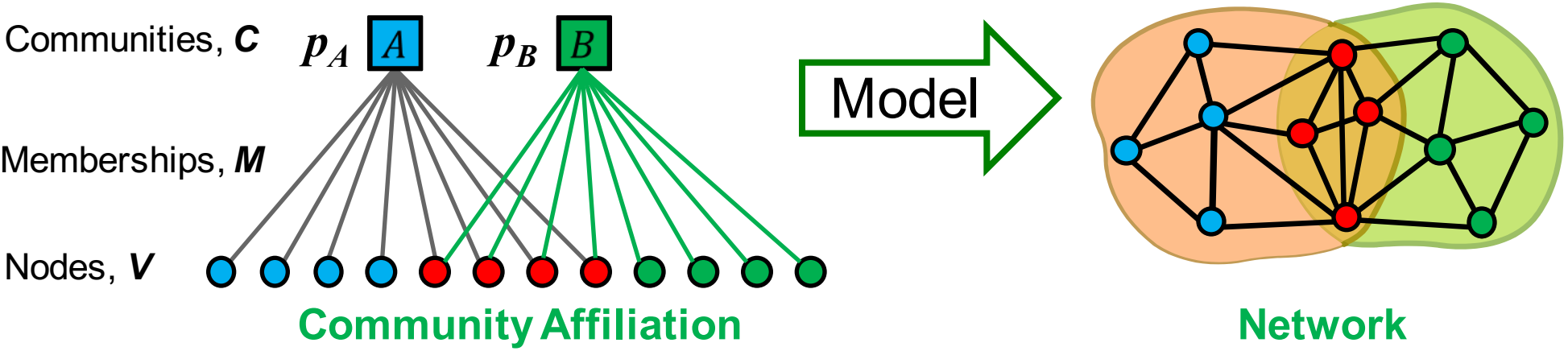
- **Many methods fail to detect dense overlaps:**
  - Clique percolation, …





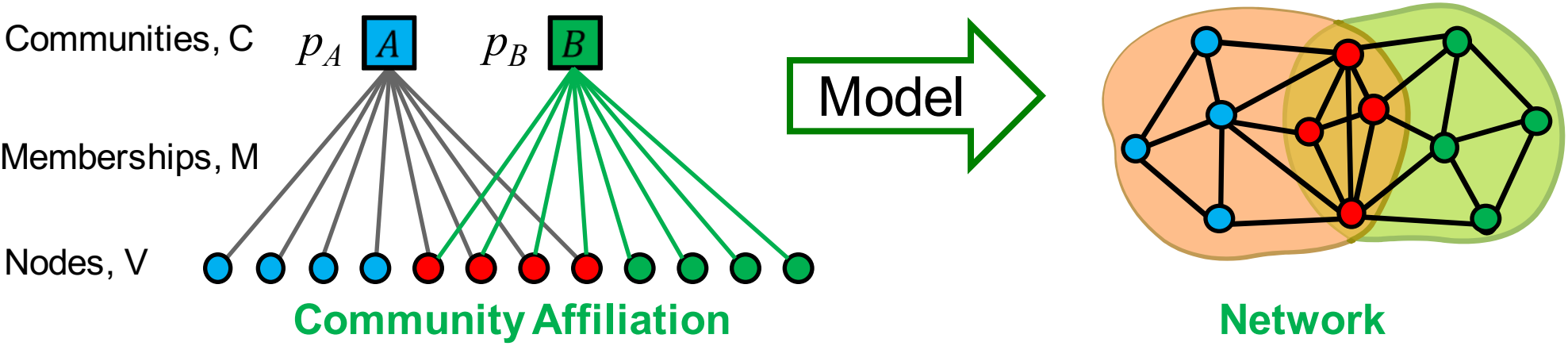
Clique percolation

Solution 2 for overlapping communities: Community-Affiliation Graph Model (AGM)



- **Generative model: How is a network generated from community affiliations?**
- **Model parameters:**
  - Nodes V, Communities C, Memberships M
  - Each community c has a single probability p_c

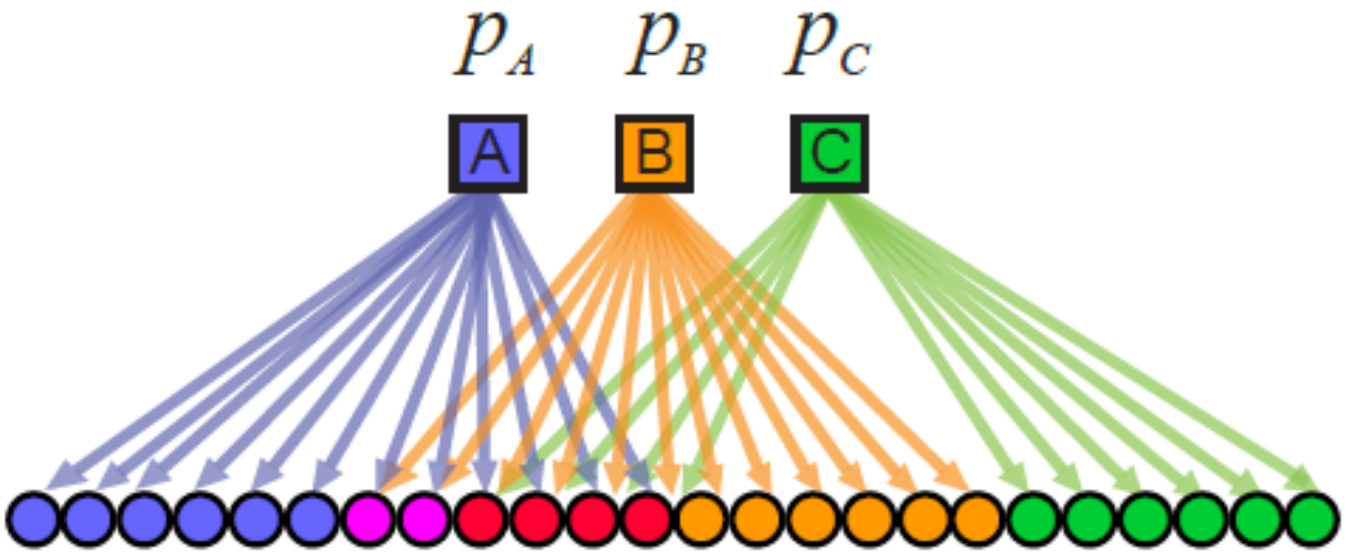
AGM: Generative Process



- **Given parameters (V, C, M, $\{p_c\}$)**
  - Nodes in community c connect to each other by flipping a coin with probability p_c
  - **Nodes that belong to multiple communities have multiple coin flips:** **D**ense community overlaps
    - If they "miss" the first time, they get another chance through the next community"

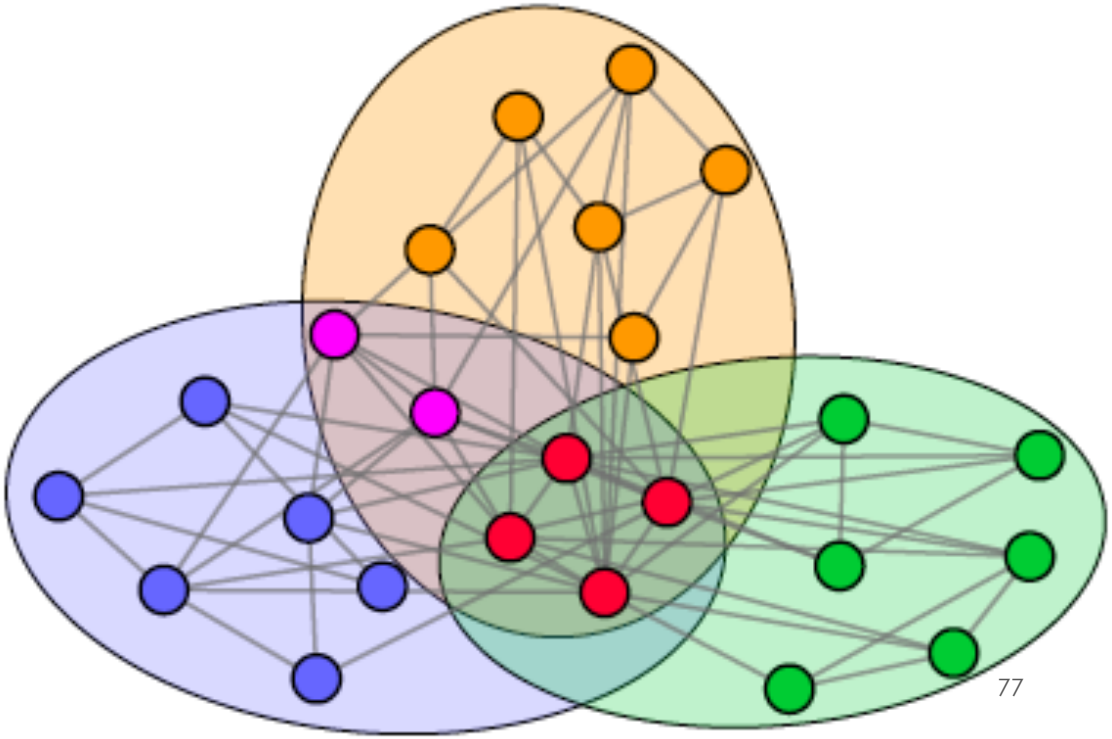
$$p(u,v) = 1 - \prod_{c \in M_u \cap M_v} (1 - p_c)$$

AGM: Dense Overlaps



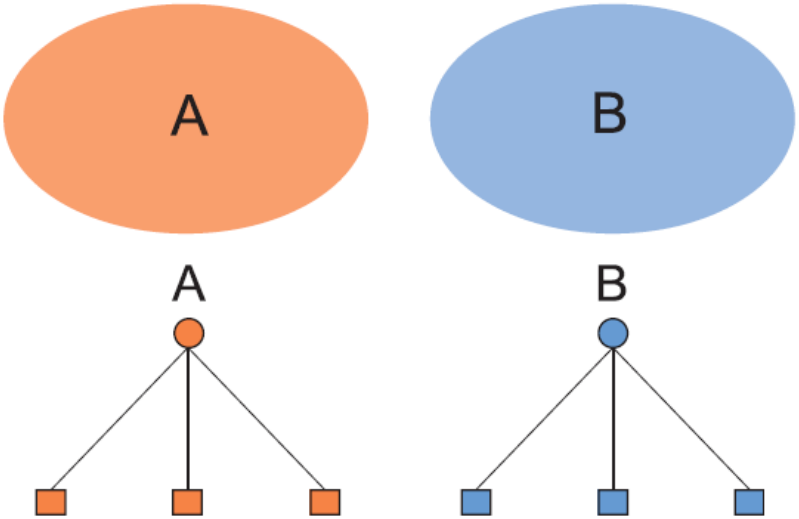
Model

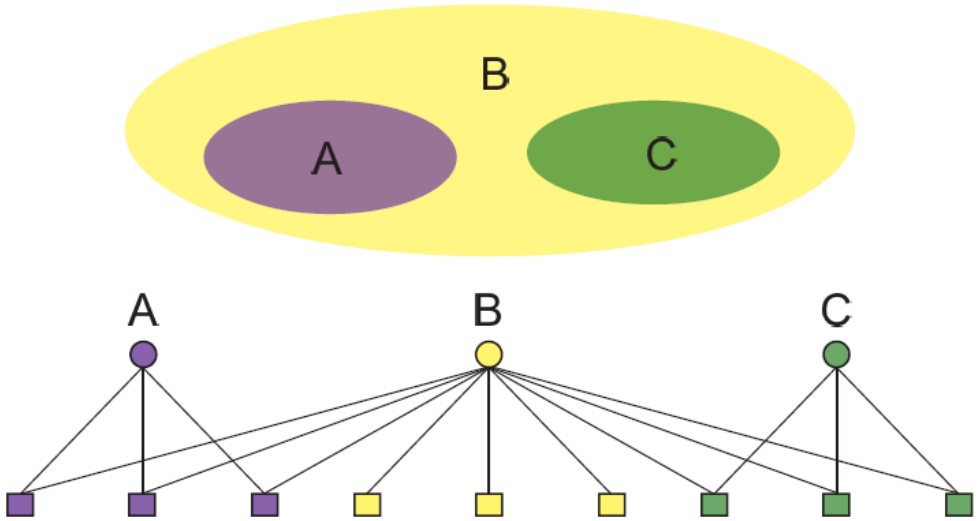
Network

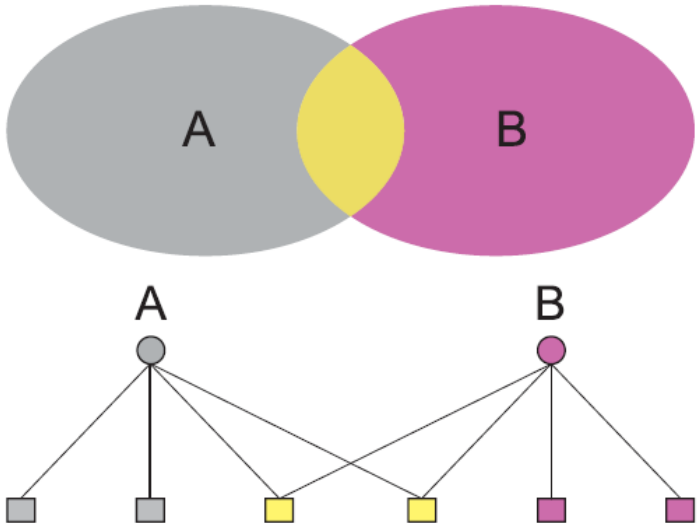


Community-Affiliation Graph Model

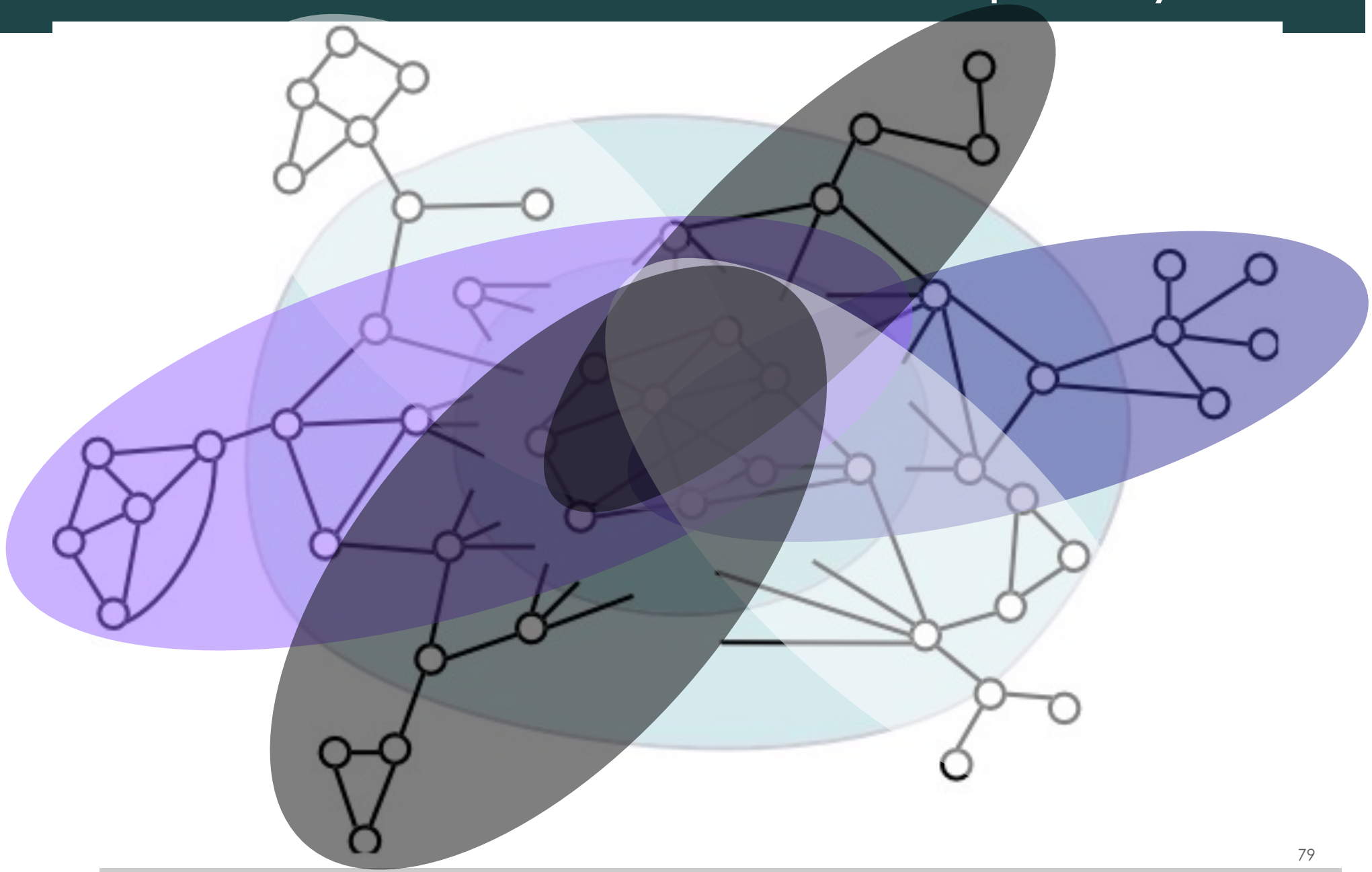
- **AGM is flexible and can express variety of network structures:**
  Non-overlapping, Nested, Overlapping



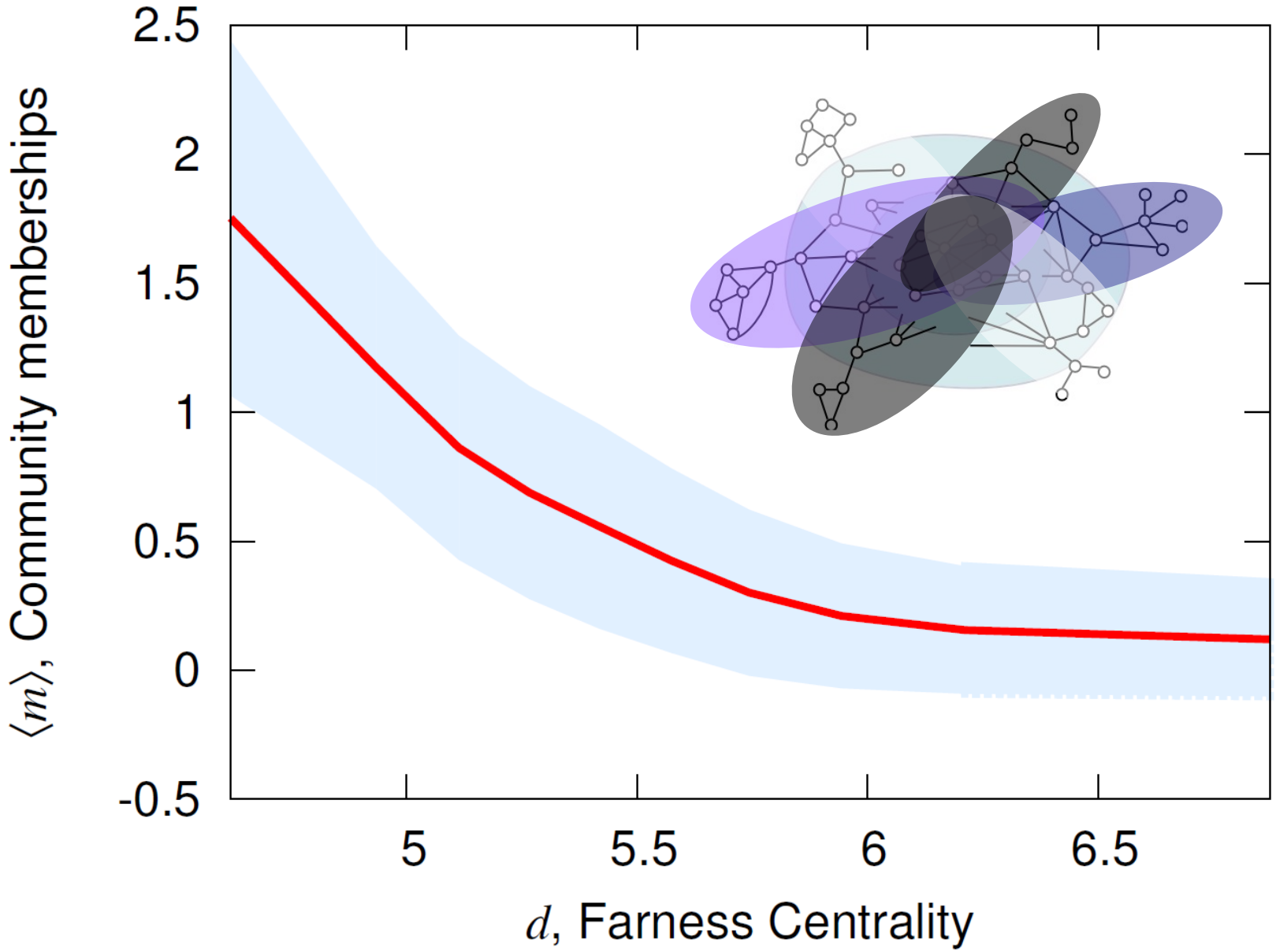




Connections: Core-Periphery

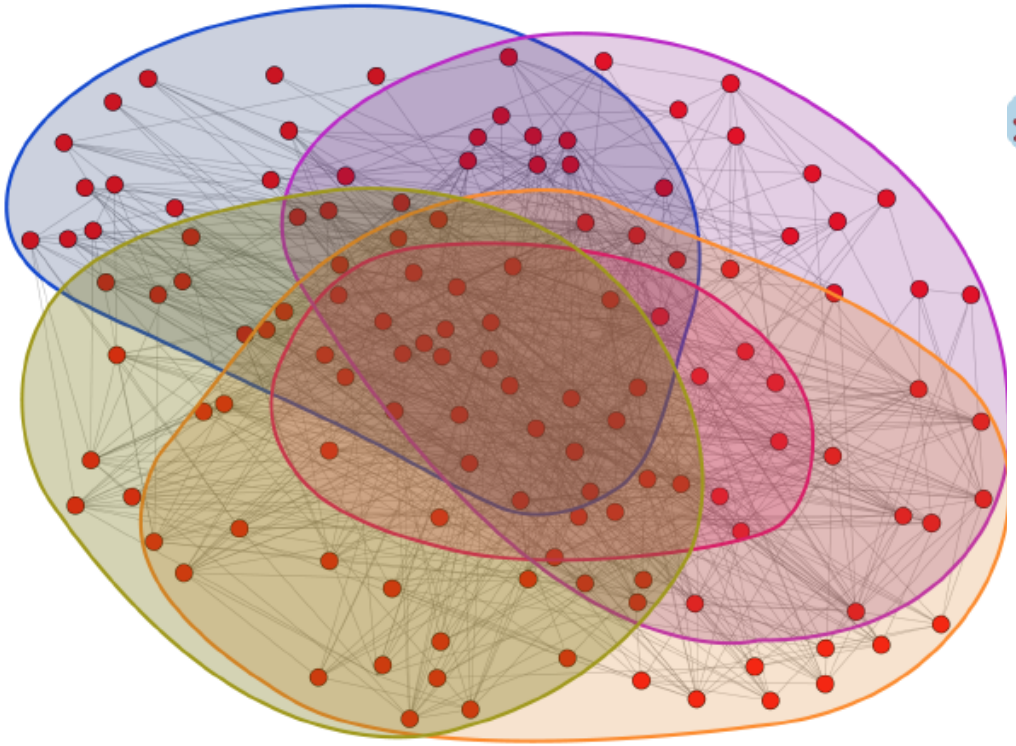


Test of the Conjecture

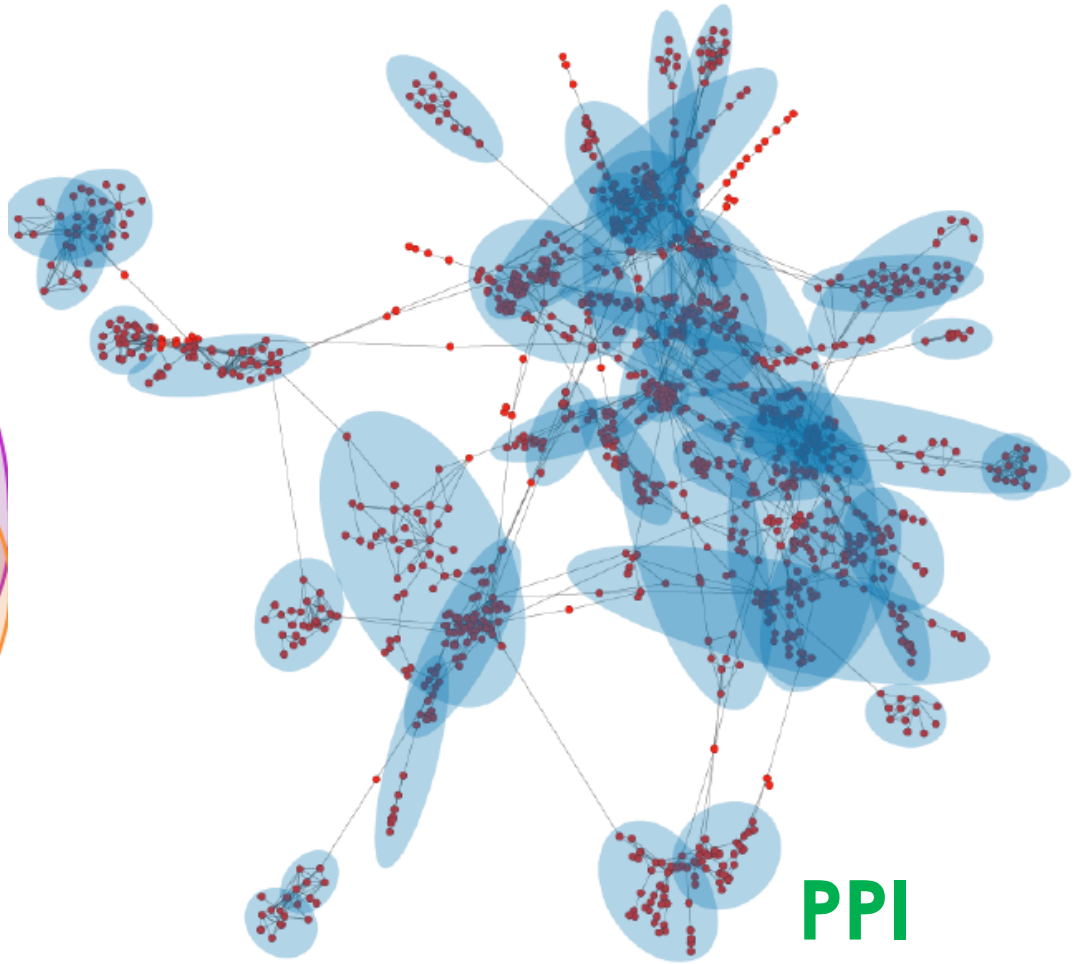


LiveJournal social network

Primary & Secondary Cores



Foodweb



PPI

- **Primary core:** Foodwebs, Web-graph, Social
- **Secondary cores:** PPI, Products

wrap up

- community structure is a way of ‘x-raying’ the network, finding out what it’s made of
- if you need to slice it up, you can
- you can look for specific structures
  - k-cliques, k-cores, etc.
- but most popular is to discover the “natural” community boundaries
- these boundaries may well overlap!