Social and Information Networks

CSCC46H, Fall 2019
Lecture 3

Prof. Ashton Anderson
ashton@cs.toronto.edu

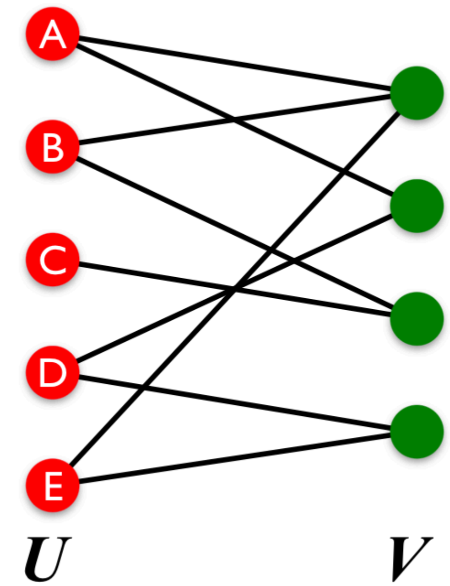
Logistics

A1 out, due Monday, Oct 7 on MarkUs

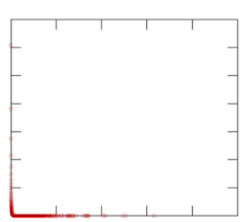
First letter of last name A-H? First blog post due Oct 4 (blog details up on website this week)

Structure of Networks

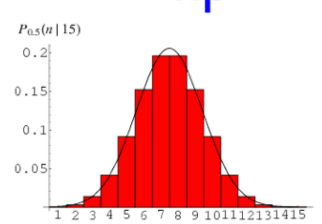
- **Last time:**
  - 1) basic network properties
  - 2) network measurements
  - 3) G_{np}
  - 4) (start of) strength of weak ties



MSN



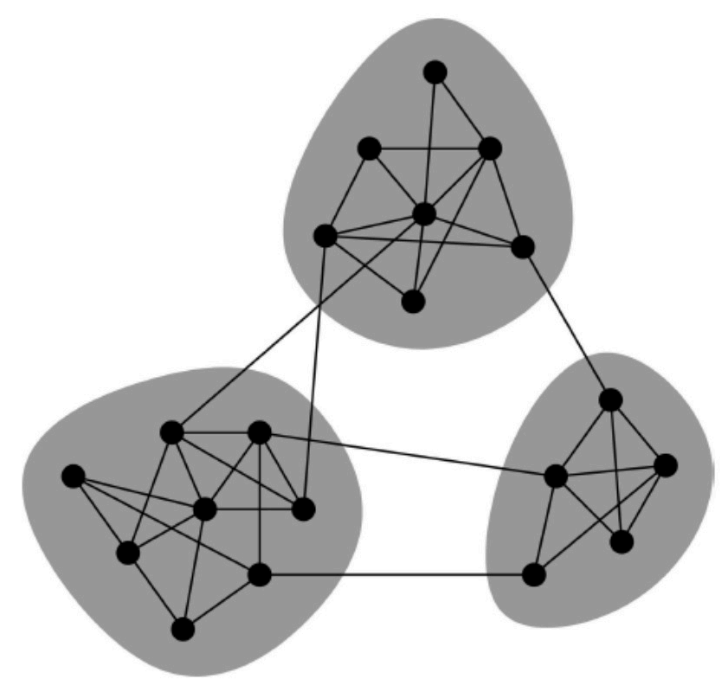
G_{np}



0.11

k / n

$\approx 8 \cdot 10^{-8}$

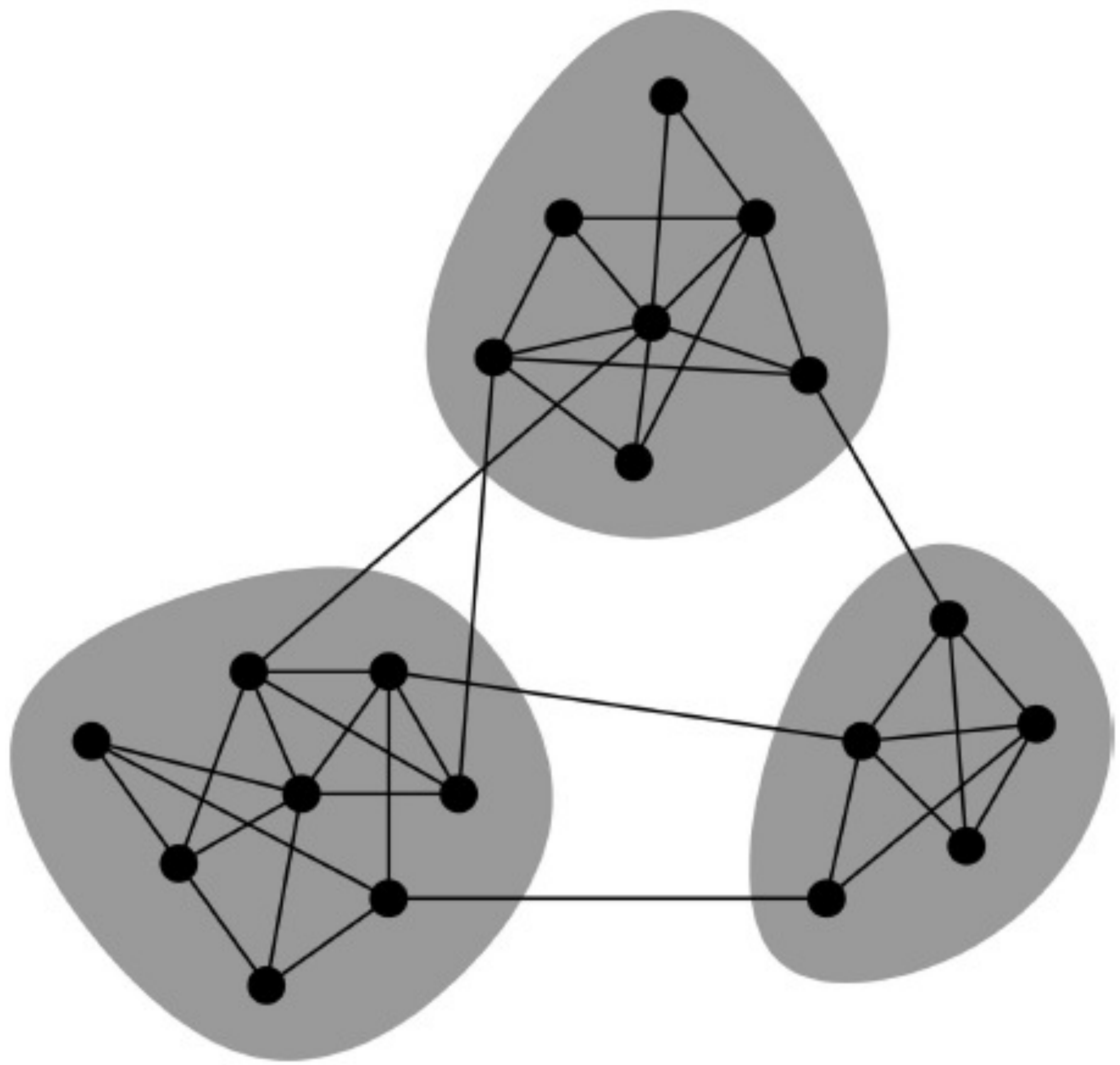


Today

1) Strength of weak ties
2) Community detection
3) Empirical network phenomena

Networks & Communities

We often think of networks "looking" like this:



What can lead to such a conceptual picture?

Networks: Flow of Information

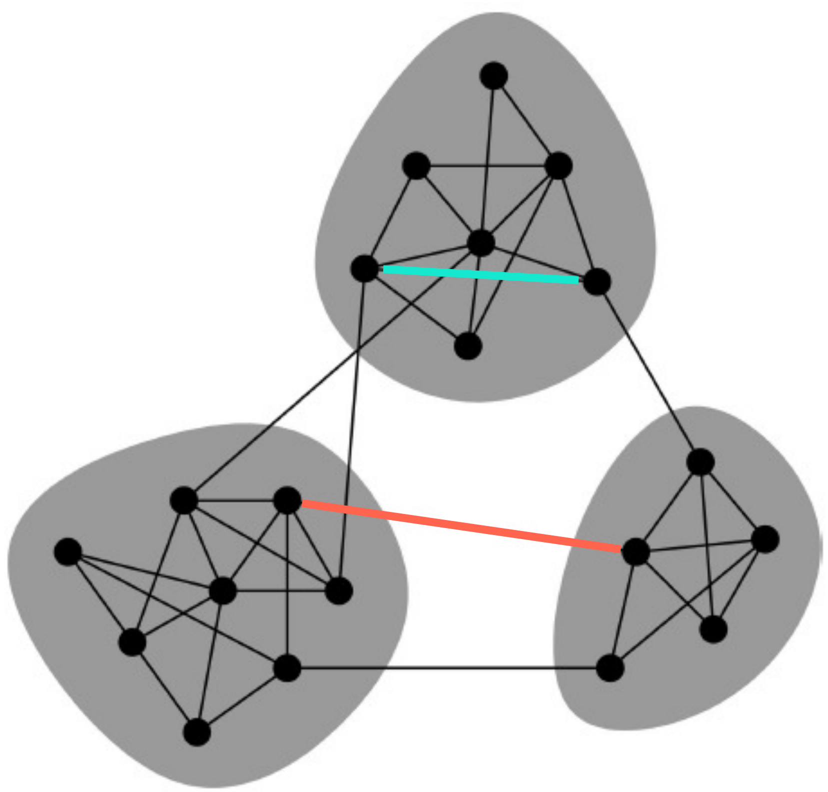
- **How does information flow through networks?**
  - What structurally distinct roles do nodes play?
  - What roles do different **links** (short vs. long) play?
- **How people find out about new jobs?**
  - Mark Granovetter, part of his PhD in 1960s
  - People find the information through personal contacts
- **But:** Contacts were often **acquaintances** rather than close friends
  - **This is surprising:** One would expect your friends to help you out more than casual acquaintances

Why is it that acquaintances are most helpful?

Granovetter's Answer

Two perspectives on friendships:

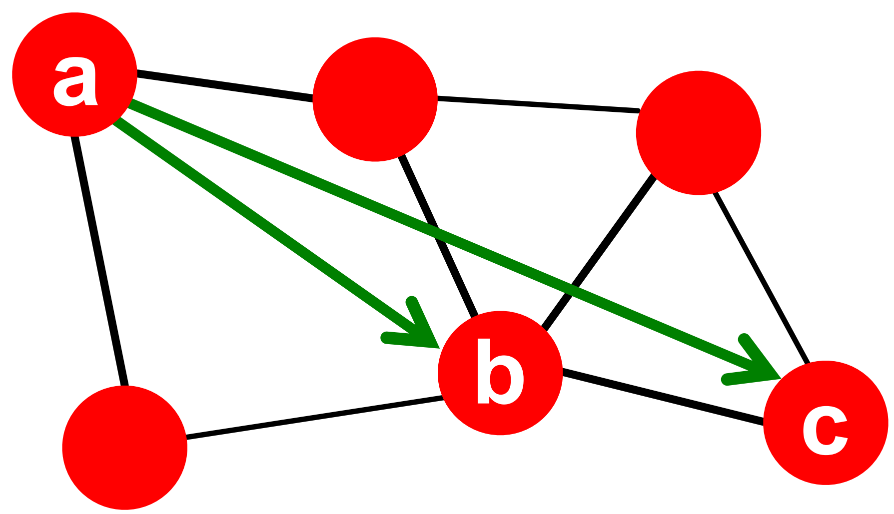
Structural: Friendships span different parts of the network



The two highlighted edges are structurally different: one spans two different "communities" and the other is inside a community

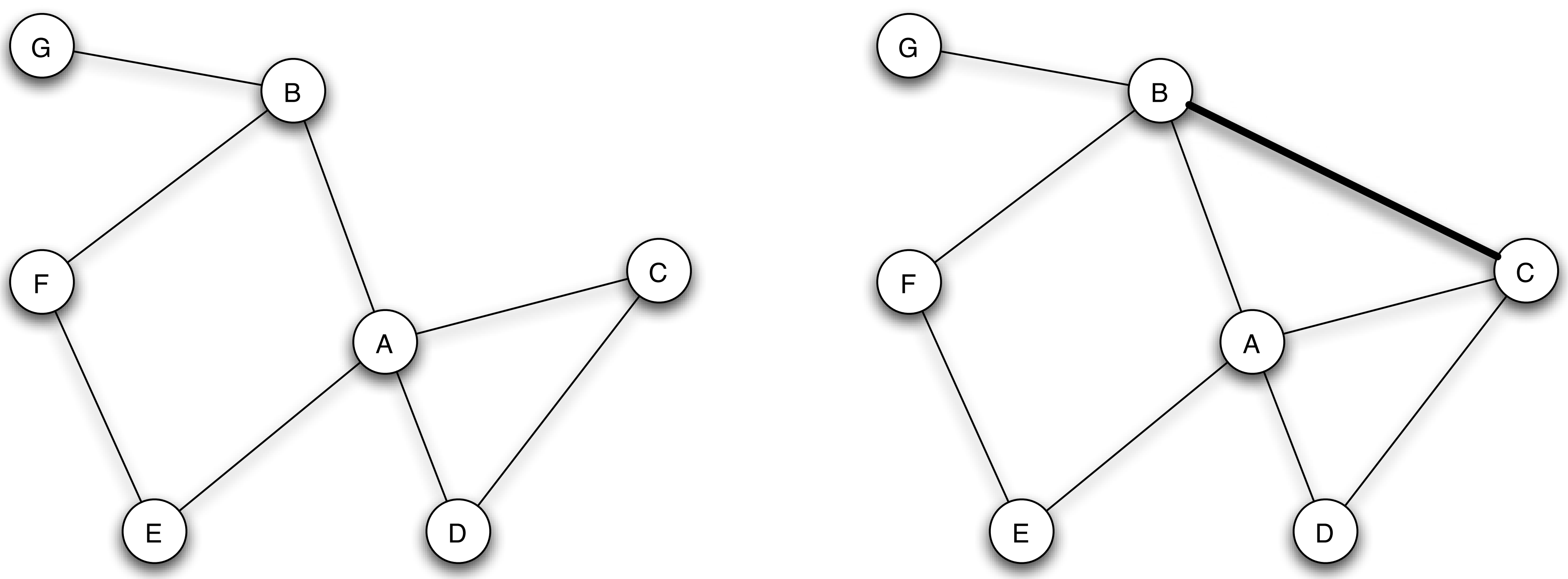
Interpersonal: Friendship between two people vary in strength, you can be close or not so close to someone

Structural force: Triadic closure



Which edge is more likely:
a–b or a–c?

Triadic closure



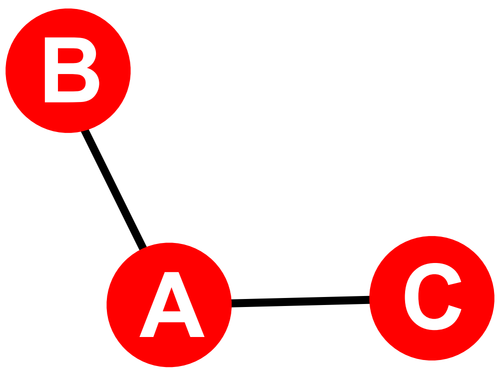
Informally: If two people in a social network have a friend in common, then there is an increased likelihood that they will become friends themselves at some point in the future.

Triadic Closure

- **Triadic closure == High clustering coefficient**

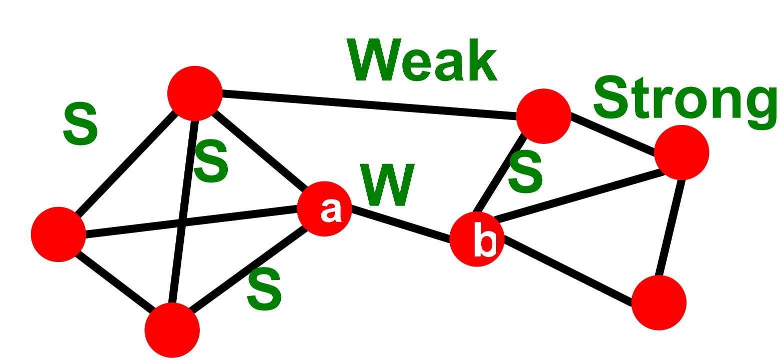
Reasons for triadic closure:

- If B and C have a friend A in common, then:
  - B is **more likely to meet** C
    - (since they both spend time with A)
  - B and C **trust** each other
    - (since they have a friend in common)
  - A has **incentive** to bring B and C together
    - (as it is hard for A to maintain two disjoint relationships)

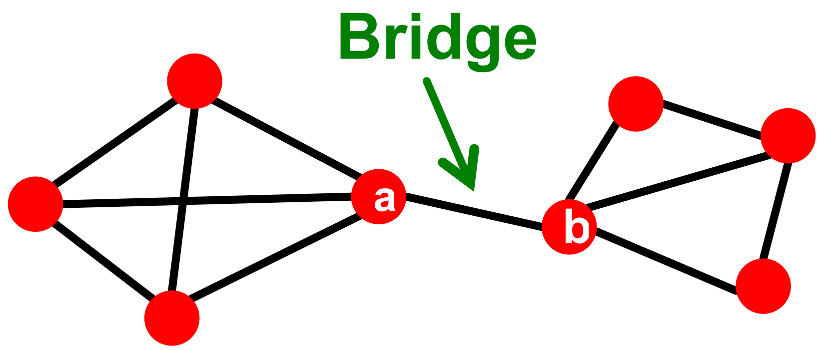


Granovetter's Explanation

- Granovetter makes a connection between social and structural role of an edge
- **First point: Structure**
  - Structurally embedded edges are also socially strong
  - Long-range edges spanning different parts of the network are socially weak
- **Second point: Information**
  - Long-range edges allow you to gather information from different parts of the network and get a job
  - Structurally embedded edges are heavily redundant in terms of information access



Network Vocabulary: Span and Bridges



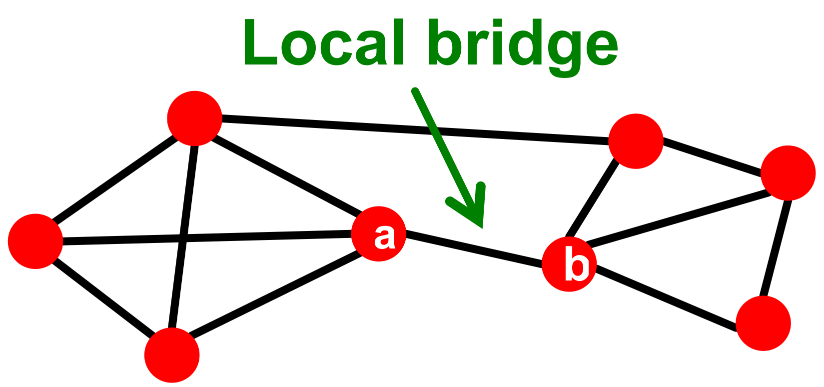
Define: **Span**

The **Span** of an edge is the distance of the edge endpoints if the edge is deleted.

Define: **Bridge edge**

If removed, it disconnects the graph

Span of a bridge edge = ∞



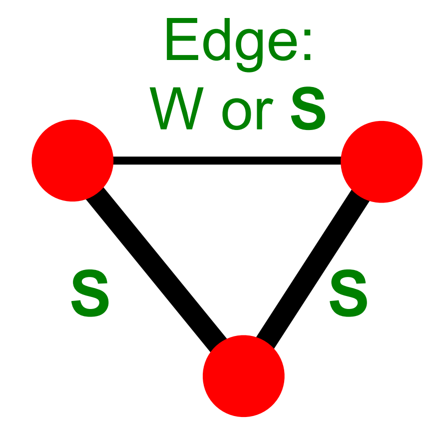
Define: **Local bridge**

Edge of **Span > 2**

(any edge that doesn't close a triangle)

Idea: Local bridges with long span are like real bridges

Granovetter's Explanation



Model: Two types of edges:

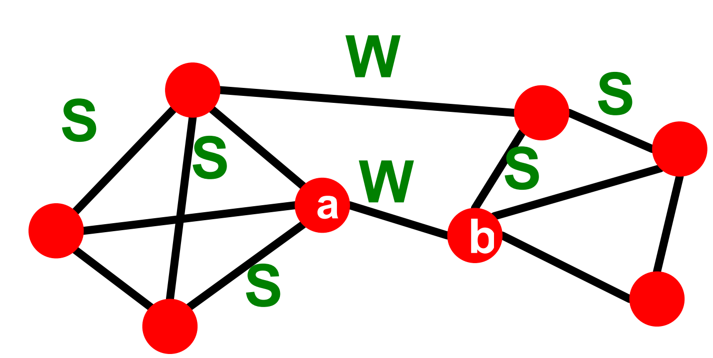
Strong (friend), **Weak** (acquaintance)

Model: Strong Triadic Closure property:

Two strong ties imply a third edge

If node A has strong ties to both nodes B and C, then there must be an edge (strong or weak) between B and C

Fact: If strong triadic closure is satisfied then **local bridges are weak ties!**

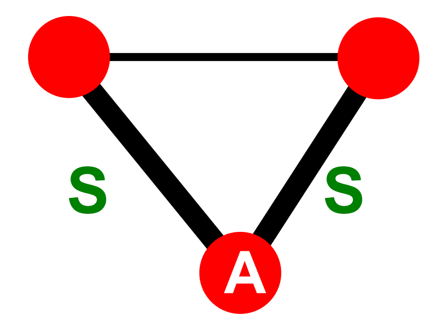


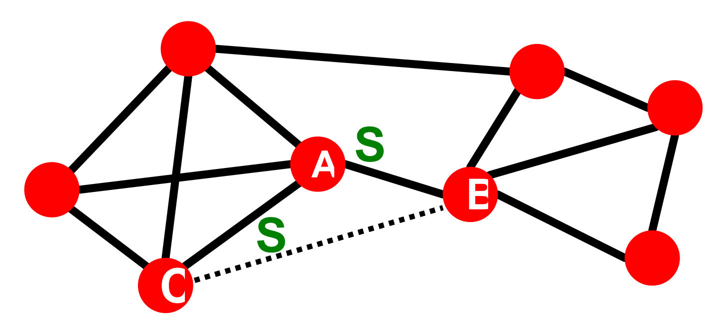
Local Bridges and Weak ties

Claim: if node A satisfies **Strong Triadic Closure** and has two strong ties, then **any local bridge adjacent to A must be a weak tie**

Proof: By contradiction:

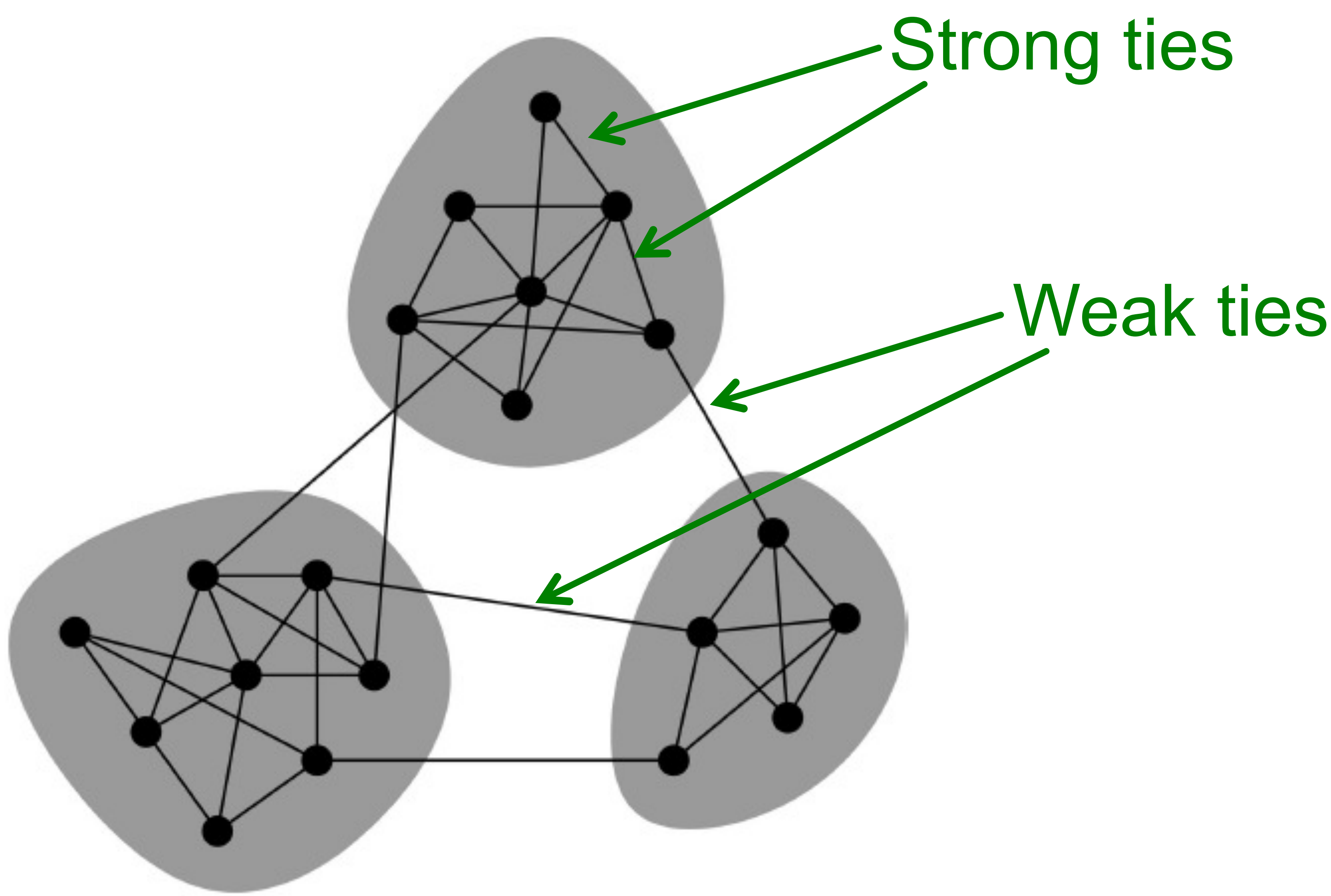
- Assume A satisfies **Strong Triadic Closure** and has **two strong ties**
- Let A–B be a **local bridge**, and assume it is a **weak tie** (to try to derive a contradiction)
- Then B–C must exist because of **Strong Triadic Closure**
- But then **A–B is not a local bridge**, because its span is 2 (without A–B, A–C–B is the shortest path)





Conceptual Picture of Networks

Granovetter's theory leads to the following conceptual picture of networks



Granovetter's Explanation

Weak ties have access to **different parts of the network**! Access to other sources and other kinds of information

Strong ties have **redundant information**

Tie strength in real data

For many years Granovetter's theory was not tested

But, today we have large who-talks-to-whom graphs:
Email, Messenger, Cell phones, Facebook

Onnela et al. 2007:
Cell-phone network of 20% of country's population
Edge strength: # phone calls

Neighborhood Overlap

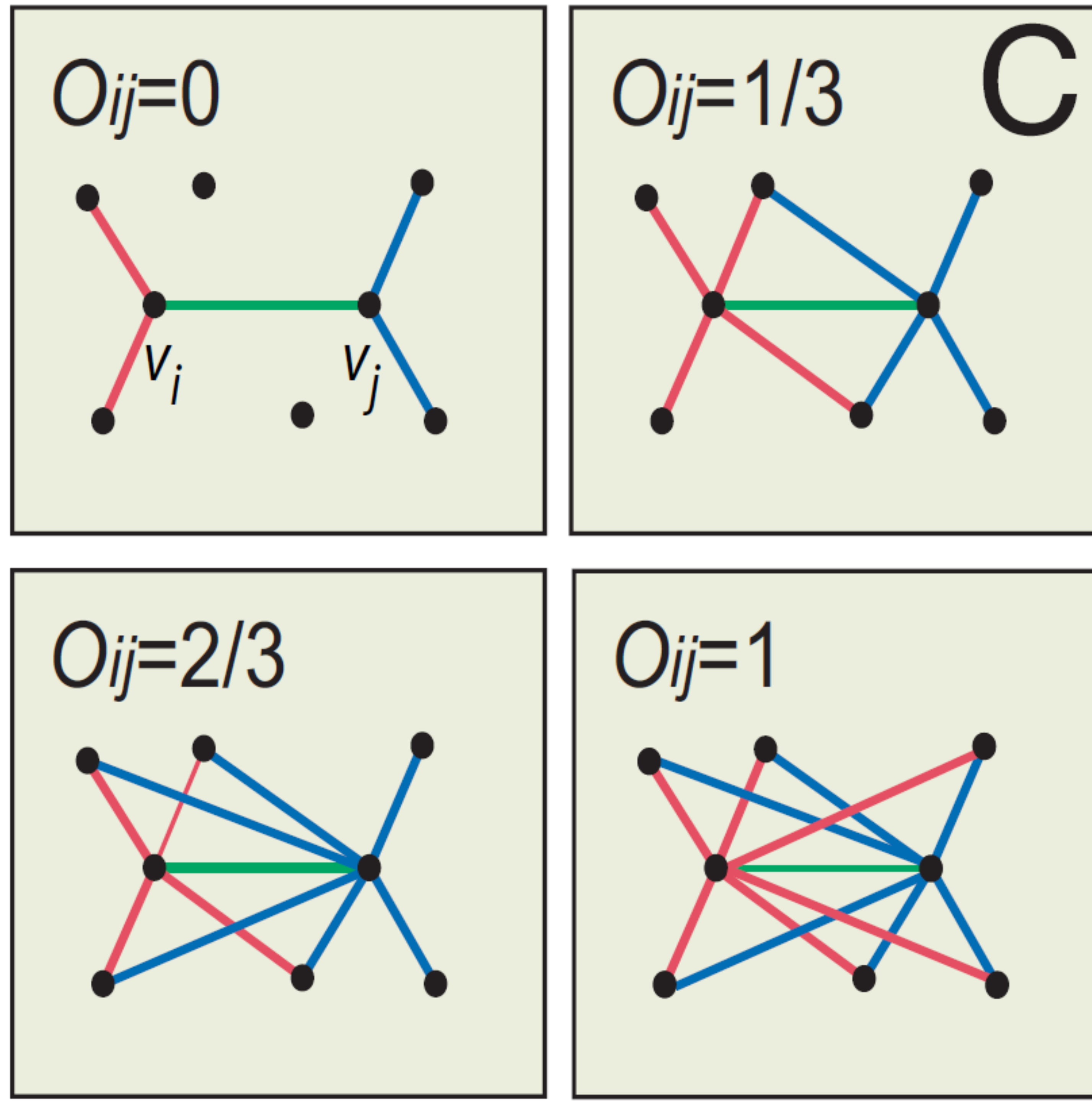
Define: **Edge overlap** as the number of shared neighbours divided by the union of neighbours:

$$O_{ij} = \frac{N(i) \cap N(j)}{N(i) \cup N(j)}$$

(N(i) = set of neighbours of node i)

O_{ij} = 0 when i–j is a local bridge

O_{ij} = 1 when i and j have all neighbours in common



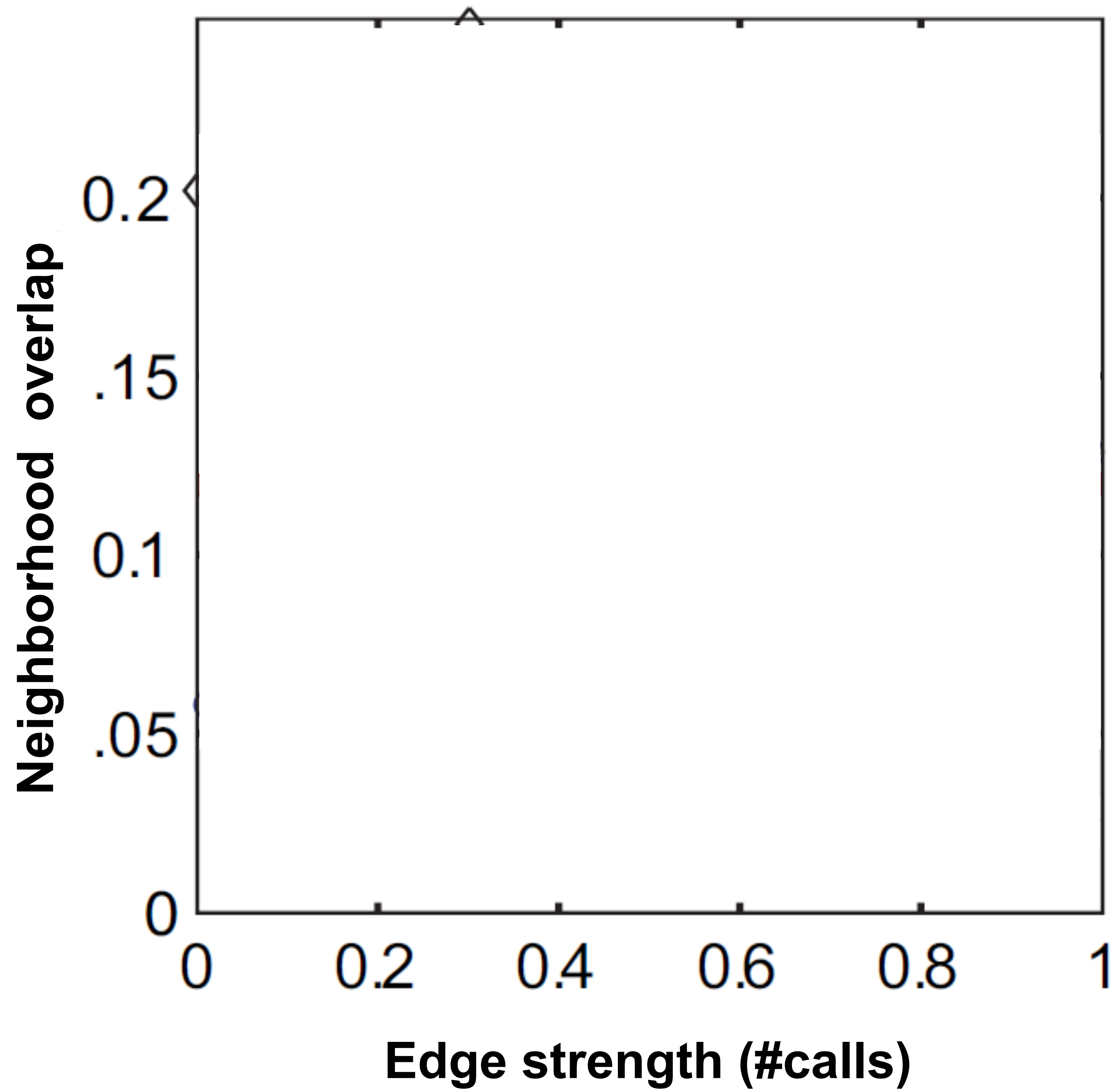
Phones: Edge Overlap vs. Strength

Let's measure the empirical relationship between edge strength and overlap in a real network!

Data: cell phone network

Legend:
x-axis: edge strength (# calls between nodes)
y-axis: overlap (how much edge bridges different parts of the network)

What do you think it will look like?



Phones: Edge Overlap vs. Strength

Legend:

True: The data

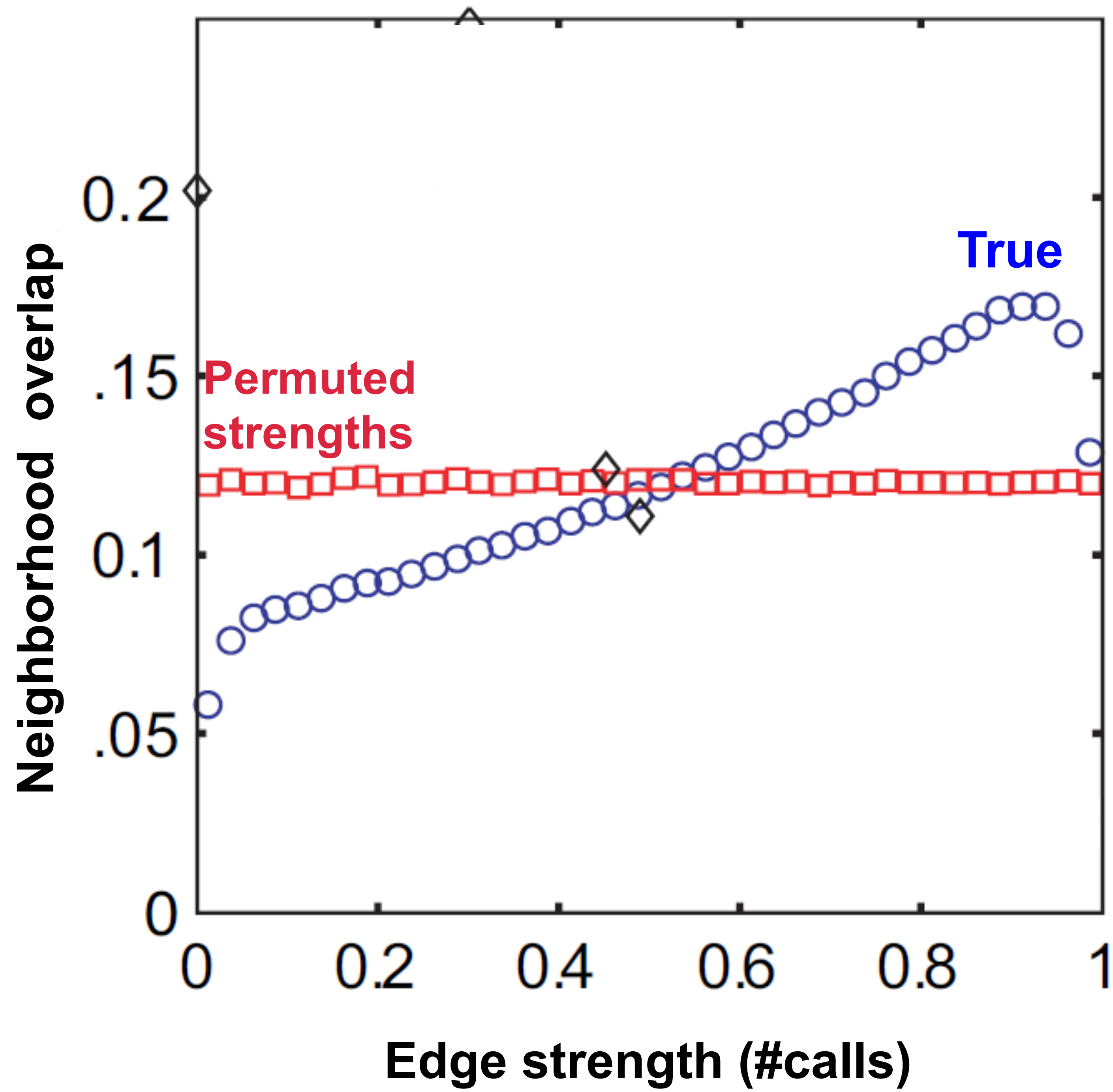
Permuted strengths: Keep the network structure but randomly reassign edge strengths

Observation:

Highly used links have high overlap!

Weak links have small overlap (bridges!)

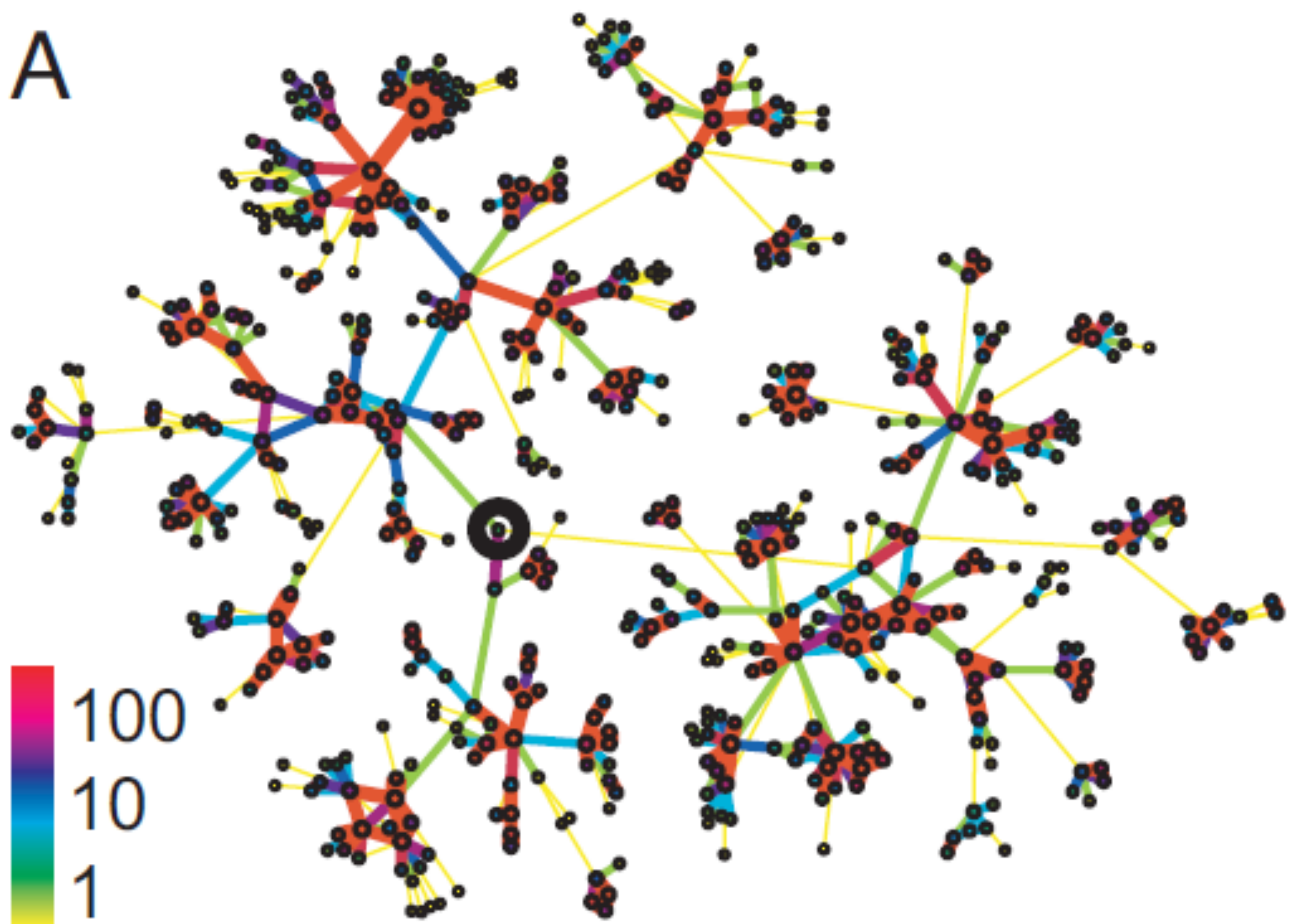
Granovetter was right



Real Network, Real Tie Strengths

Real edge strengths in mobile call graph

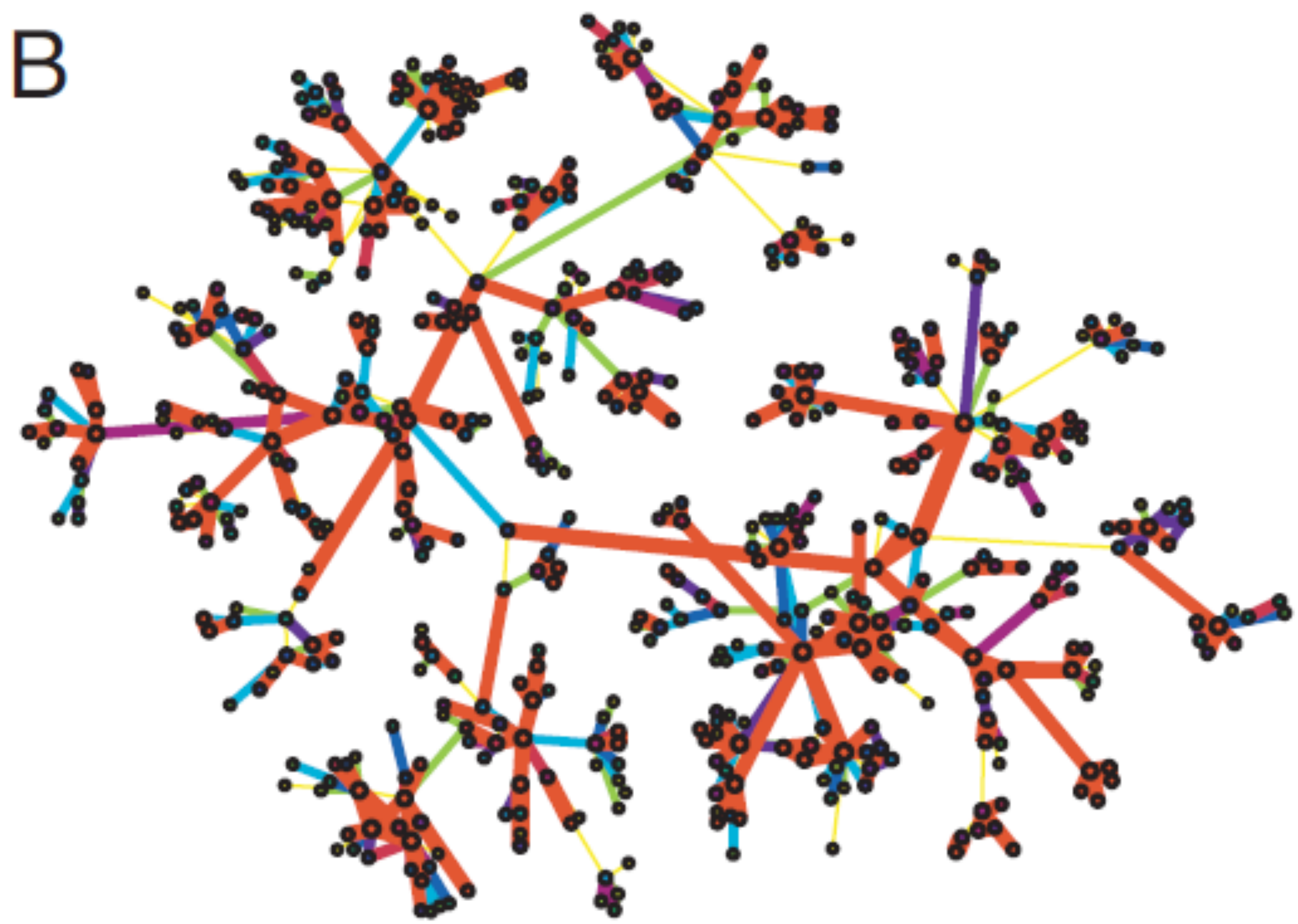
Strong ties are more embedded (have higher overlap), and occur mostly in clustered communities



Real Net, Permuted Tie Strengths

Same network, same set of edge strengths but now strengths are randomly shuffled

- Now high overlap edges are much more likely to span different parts of the network (not what we see in real life)



Link Removal by Strength

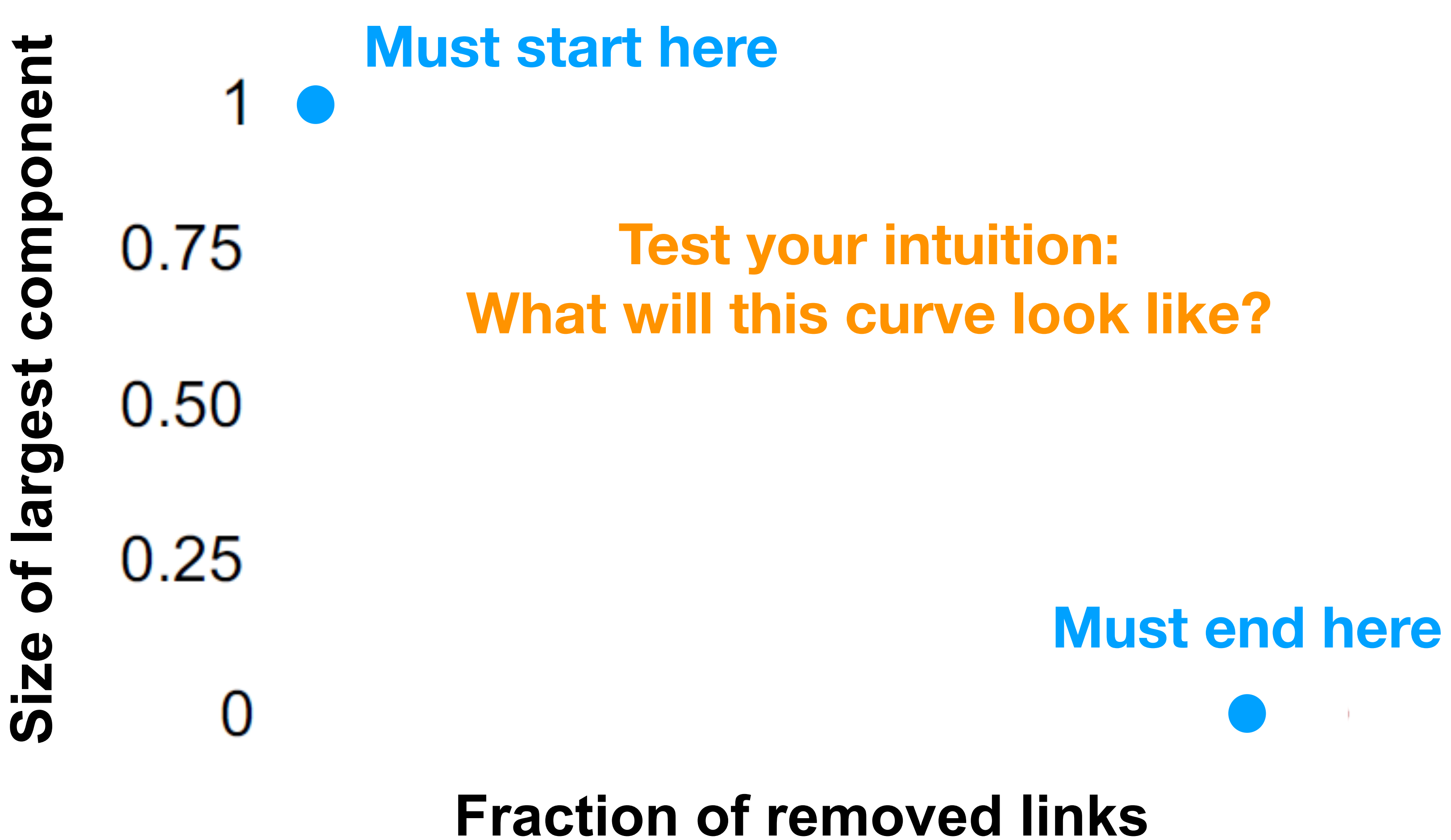
An important, recurring concept in network analysis is **network robustness**: how quickly does the graph become disconnected as you remove links?

The faster the network falls apart, the more prone to failure it is

Test importance of edges by changing the order in which you remove them

Link Removal by Strength

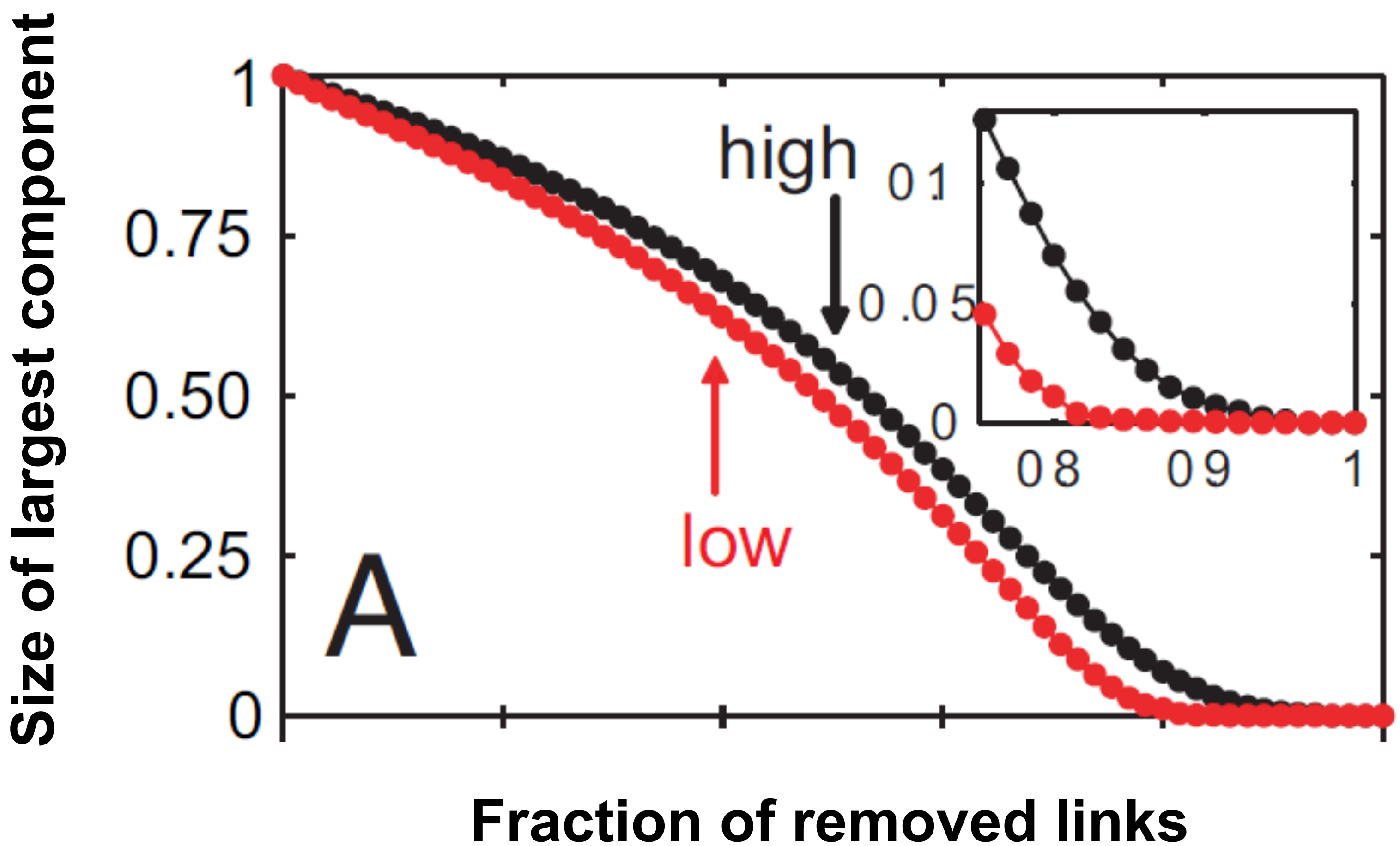
In the mobile call graph, we will test the importance of strong/weak edges, as well as high/low overlap edges, by employing this strategy



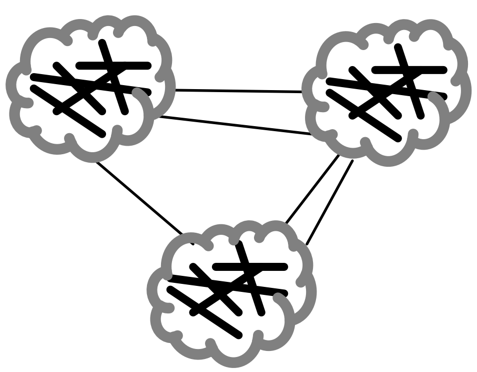
Link Removal by Strength

Removing links by **strength (#calls)**

- Low to high
- High to low



Low disconnects the network sooner

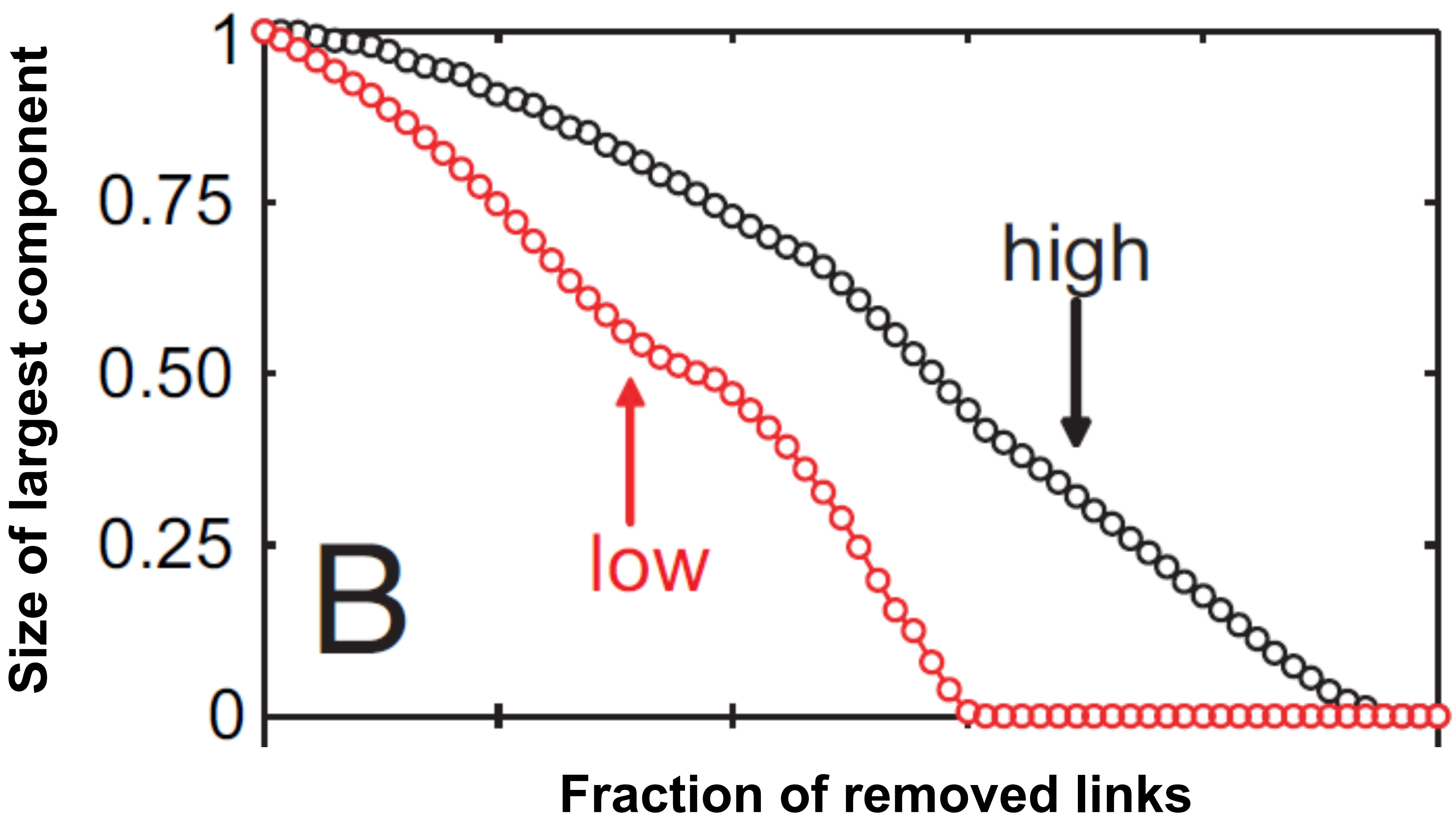


Conceptual picture of network structure

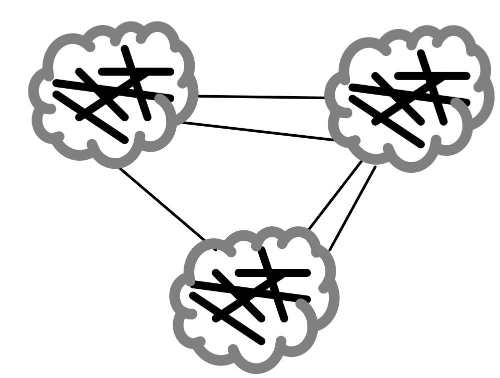
Link Removal by Overlap

Removing links based on **overlap**

- Low to high
- High to low



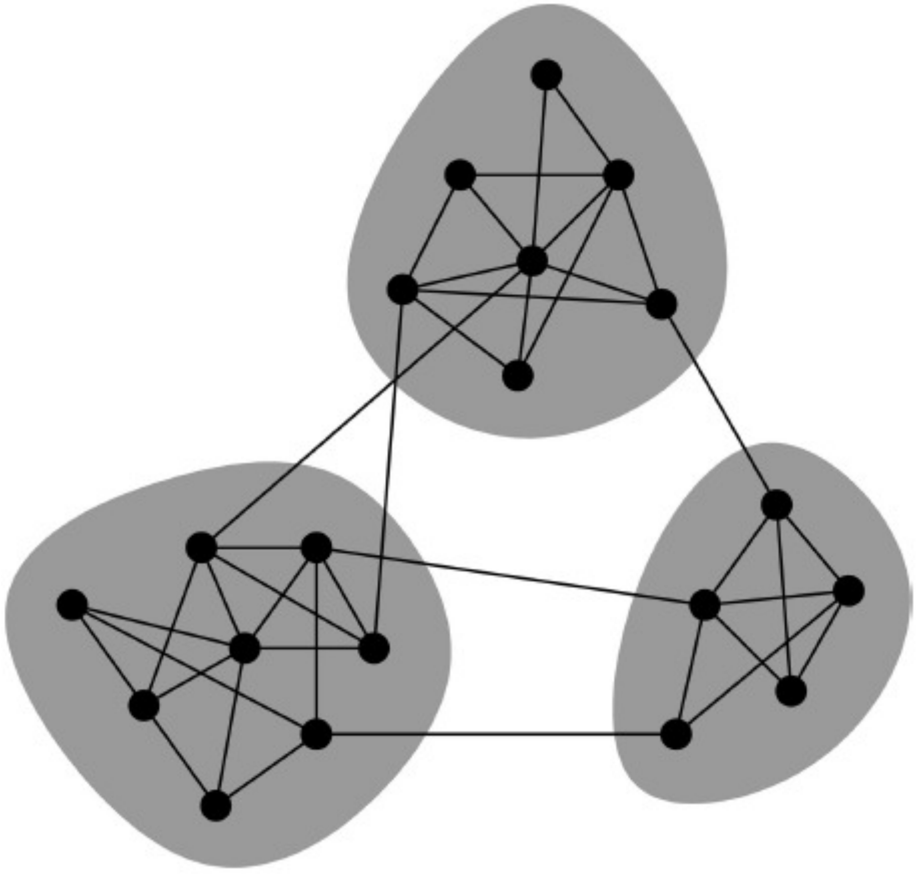
Low disconnects the network **much** sooner



Conceptual picture of network structure

Network Communities

- Granovetter's **strength of weak ties theory** suggests that networks are composed of **tightly connected sets of nodes**



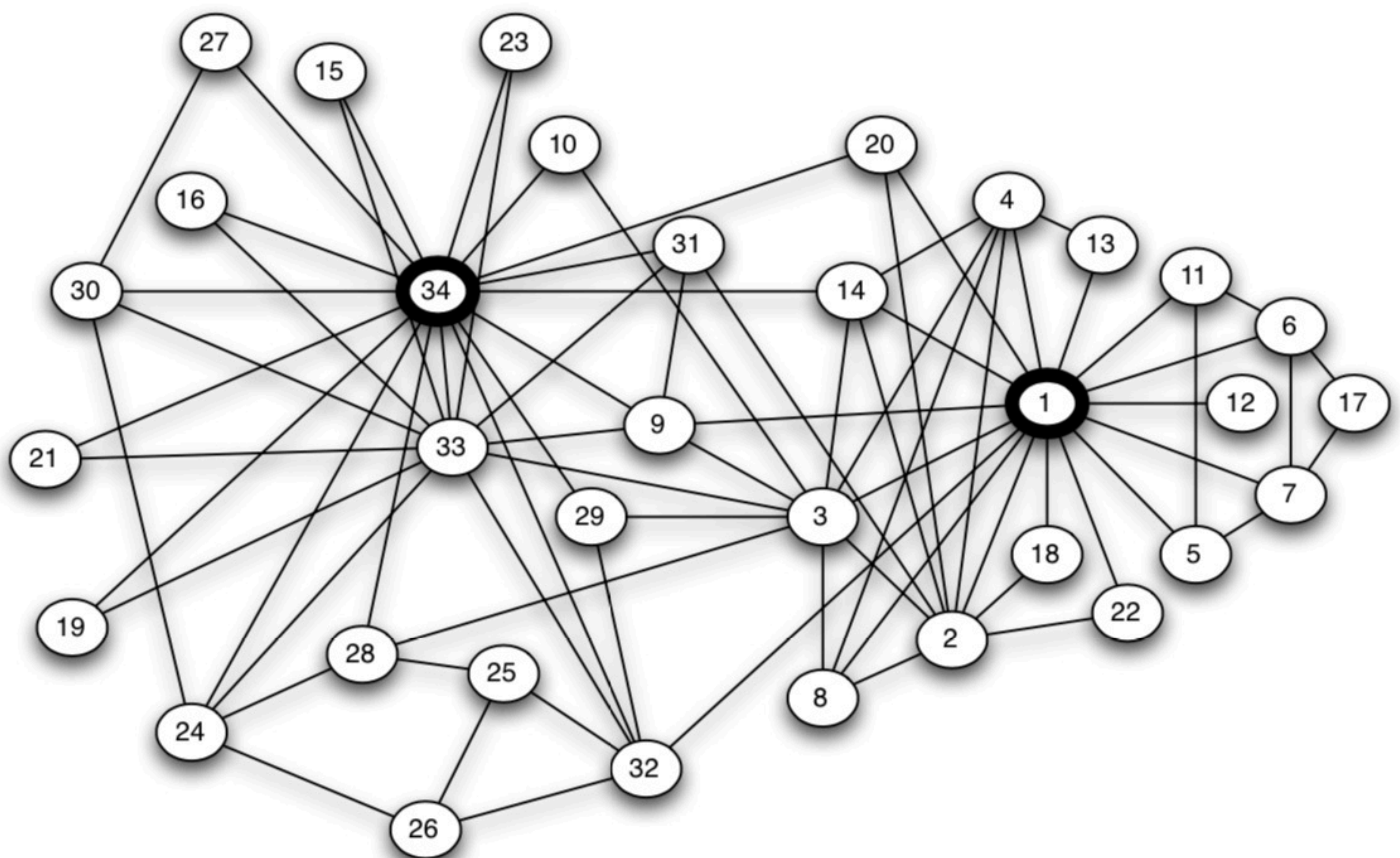
- **Network communities:**
  - Sets of nodes with **lots** of connections **inside** and **few** to **outside** (the rest of the network)

Social Network Data

Zachary's Karate club network:

Observe social ties and rivalries in a university karate club

34: president



1: instructor

Social Network Data

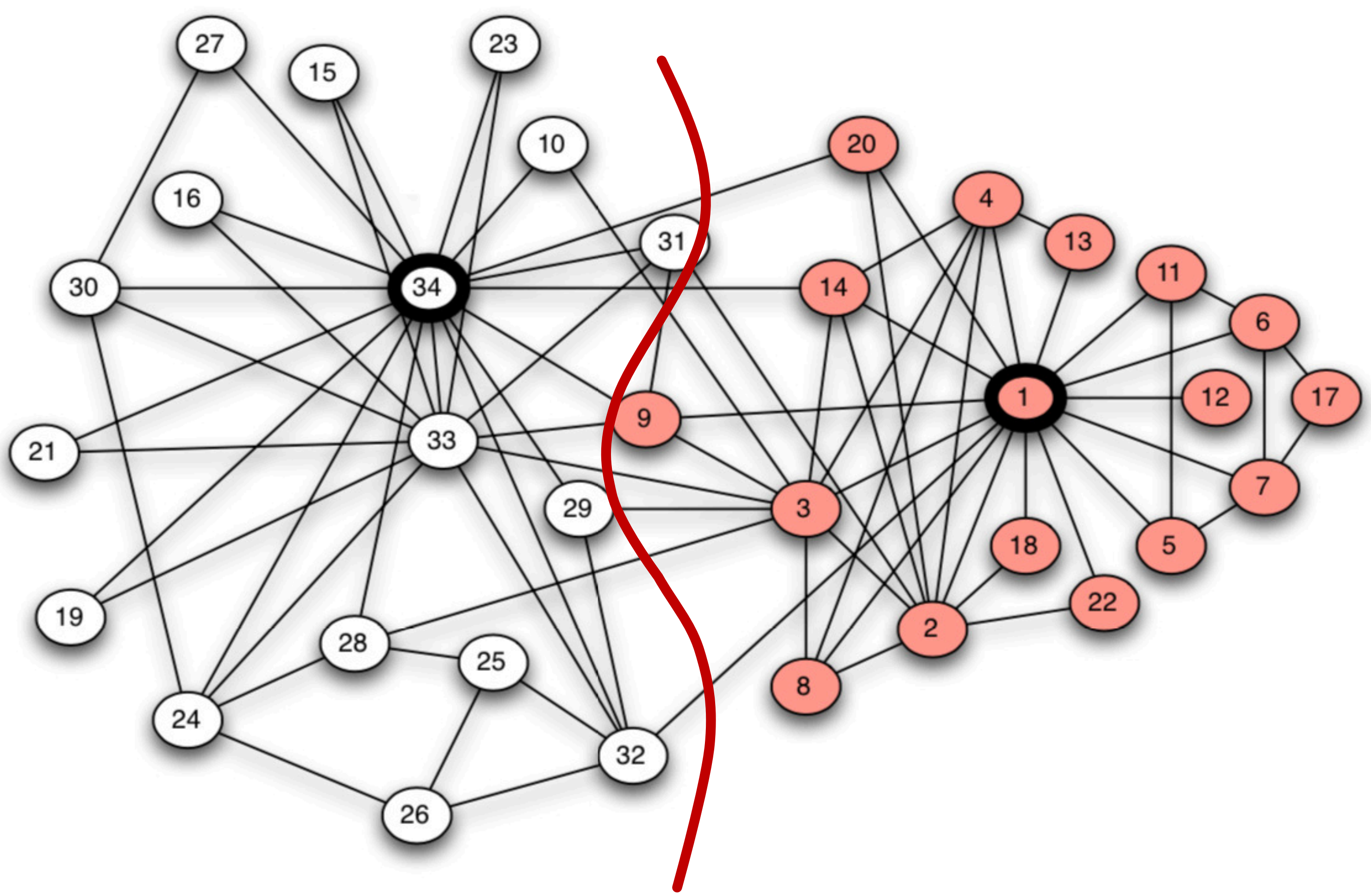
Zachary's Karate club network:

Observe social ties and rivalries in a university karate club

During his observation, conflicts led the group to split

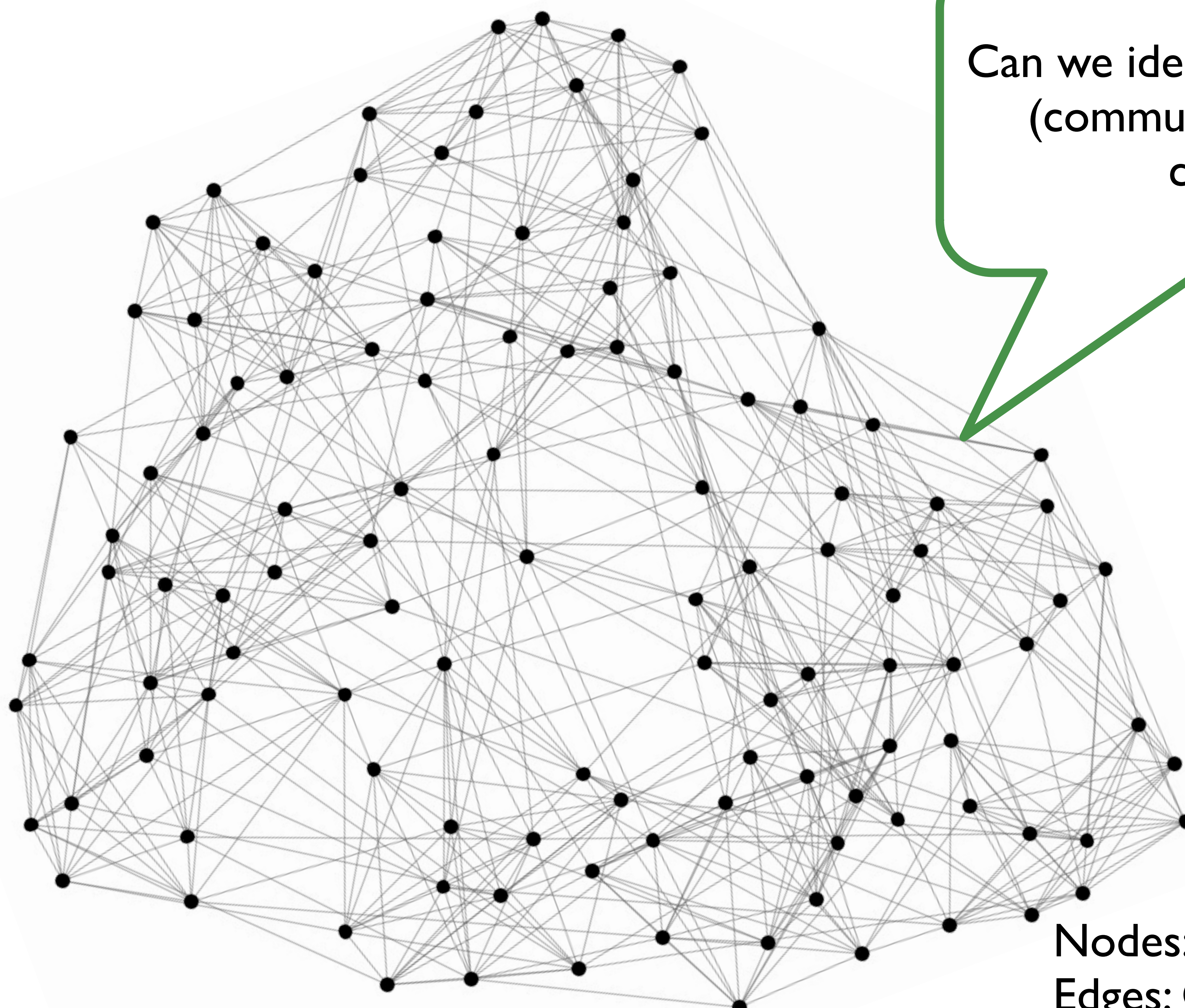
Split could be explained by a minimum cut in the network

34: president



1: instructor

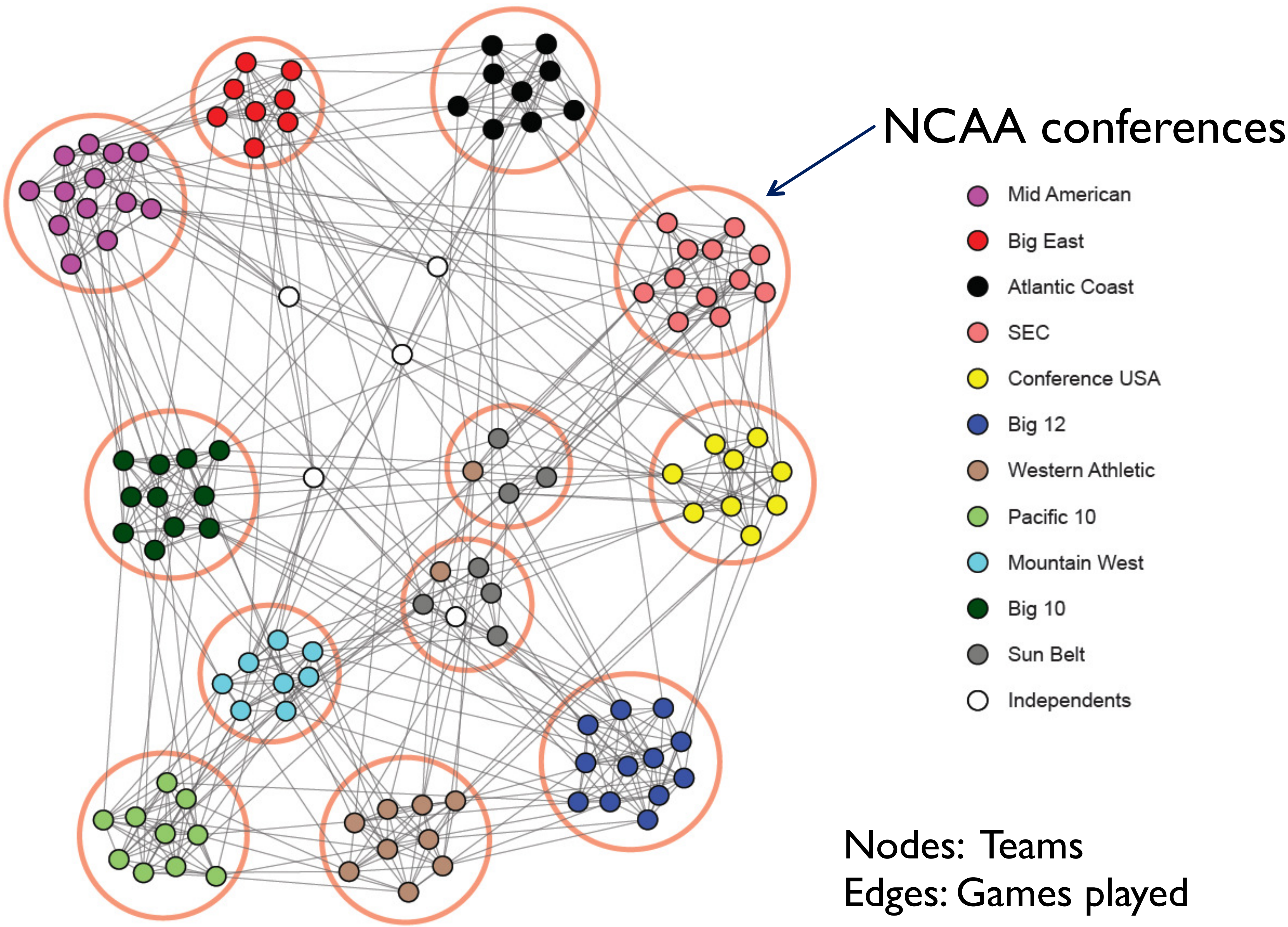
NCAA Football Network



Can we identify node groups? (communities, modules, clusters)

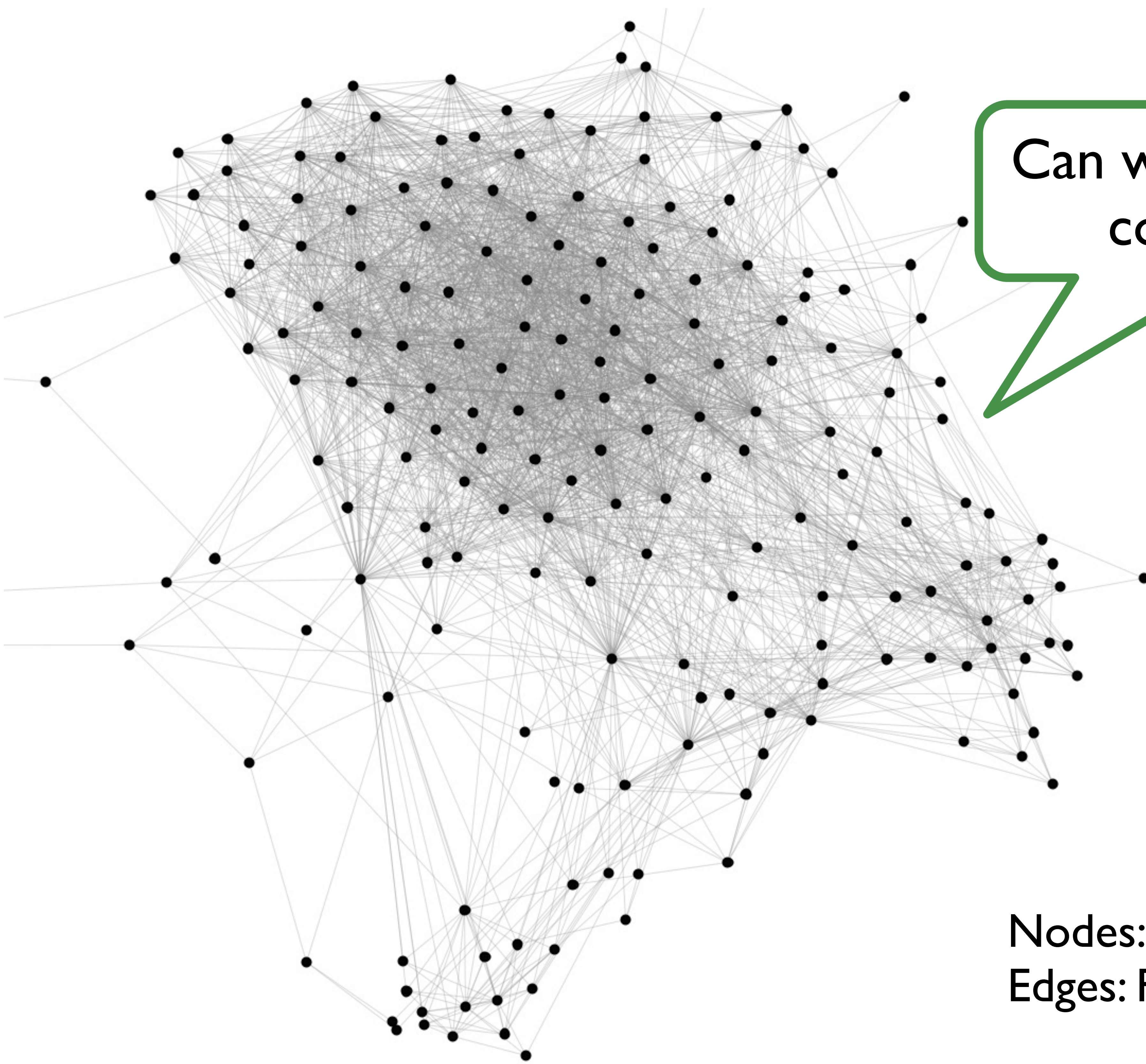
Nodes: Teams
Edges: Games played

NCAA Football Network



Nodes: Teams
Edges: Games played

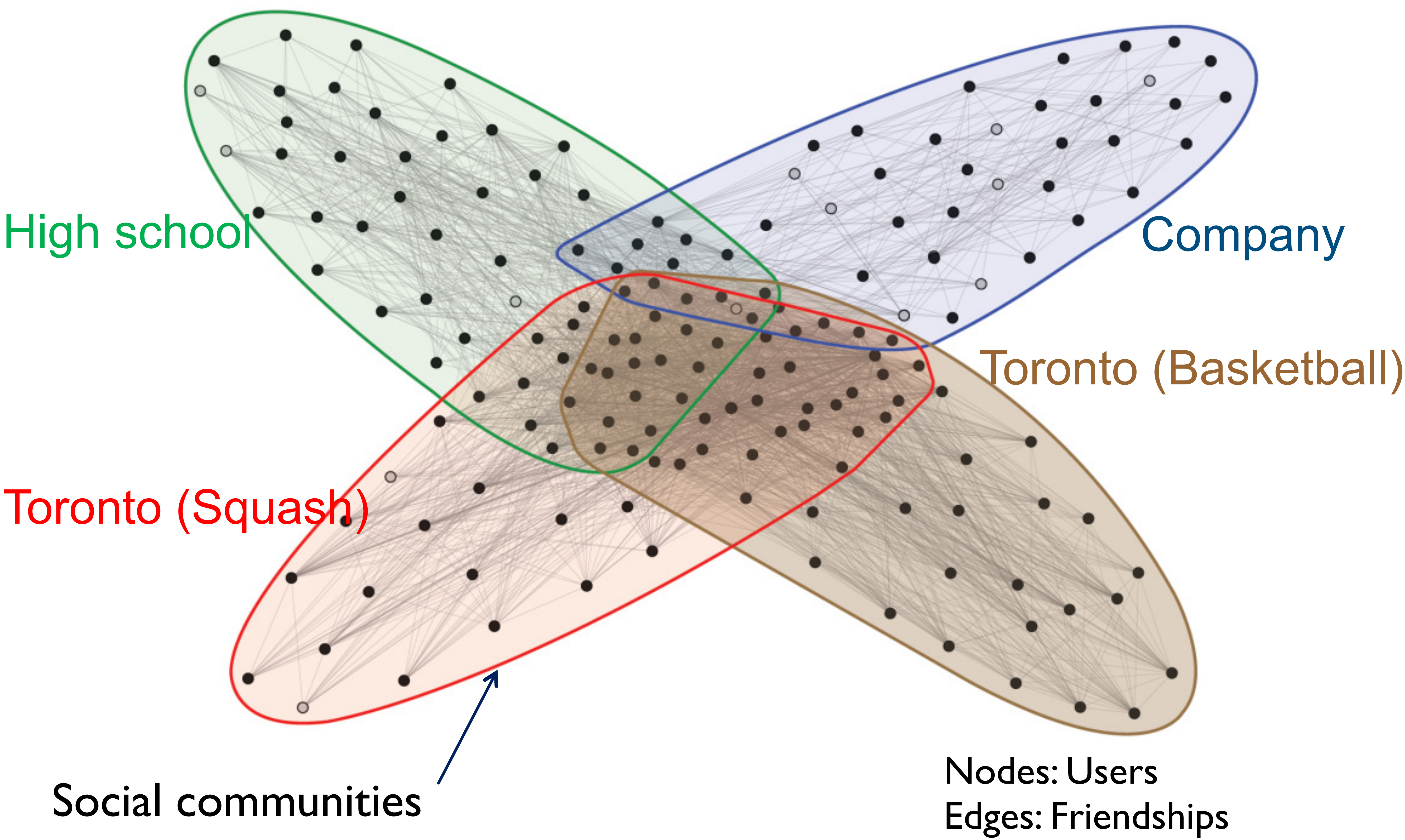
Facebook Ego-network



Can we identify social communities?

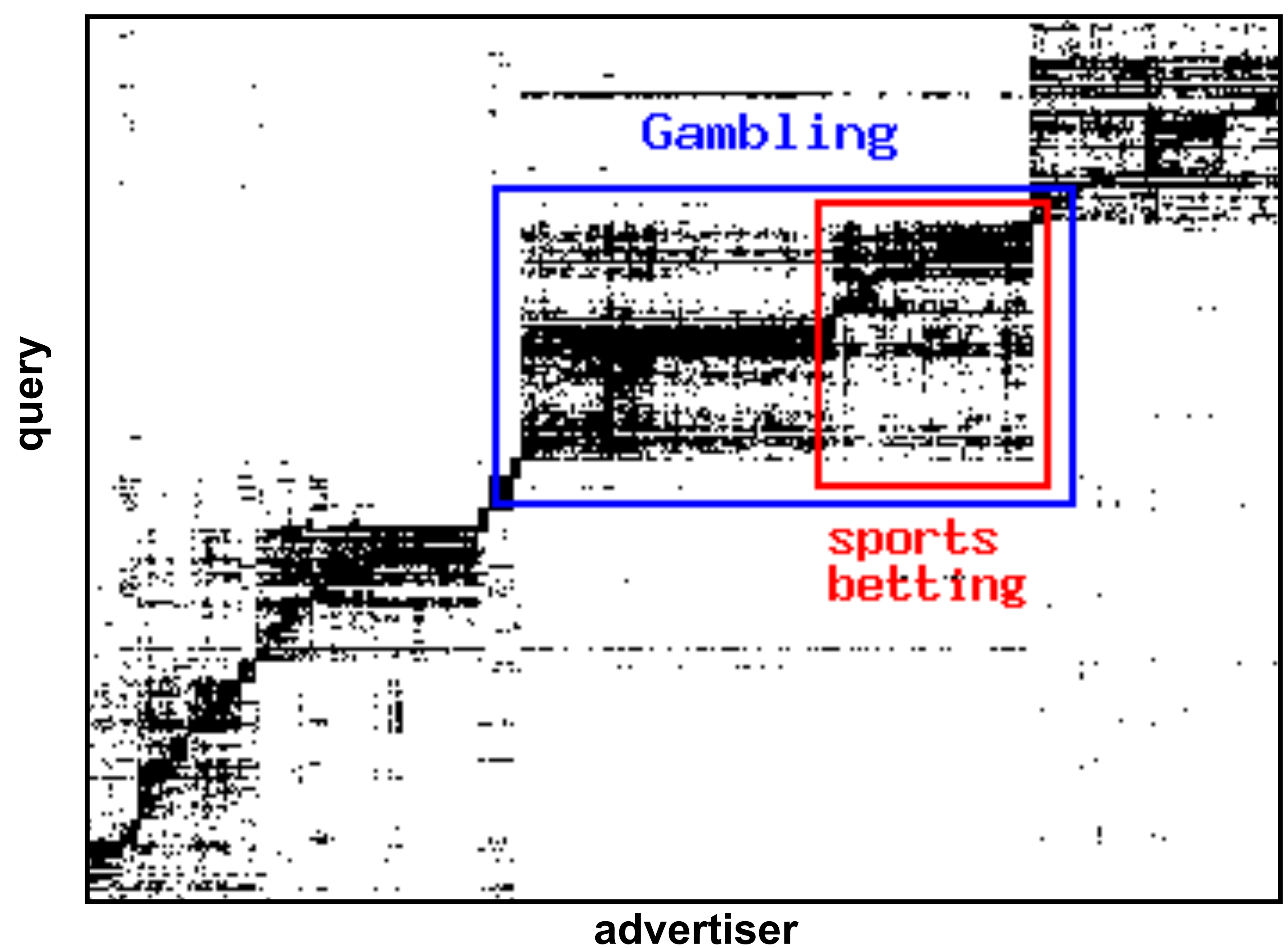
Nodes: Users
Edges: Friendships

Facebook Ego-network

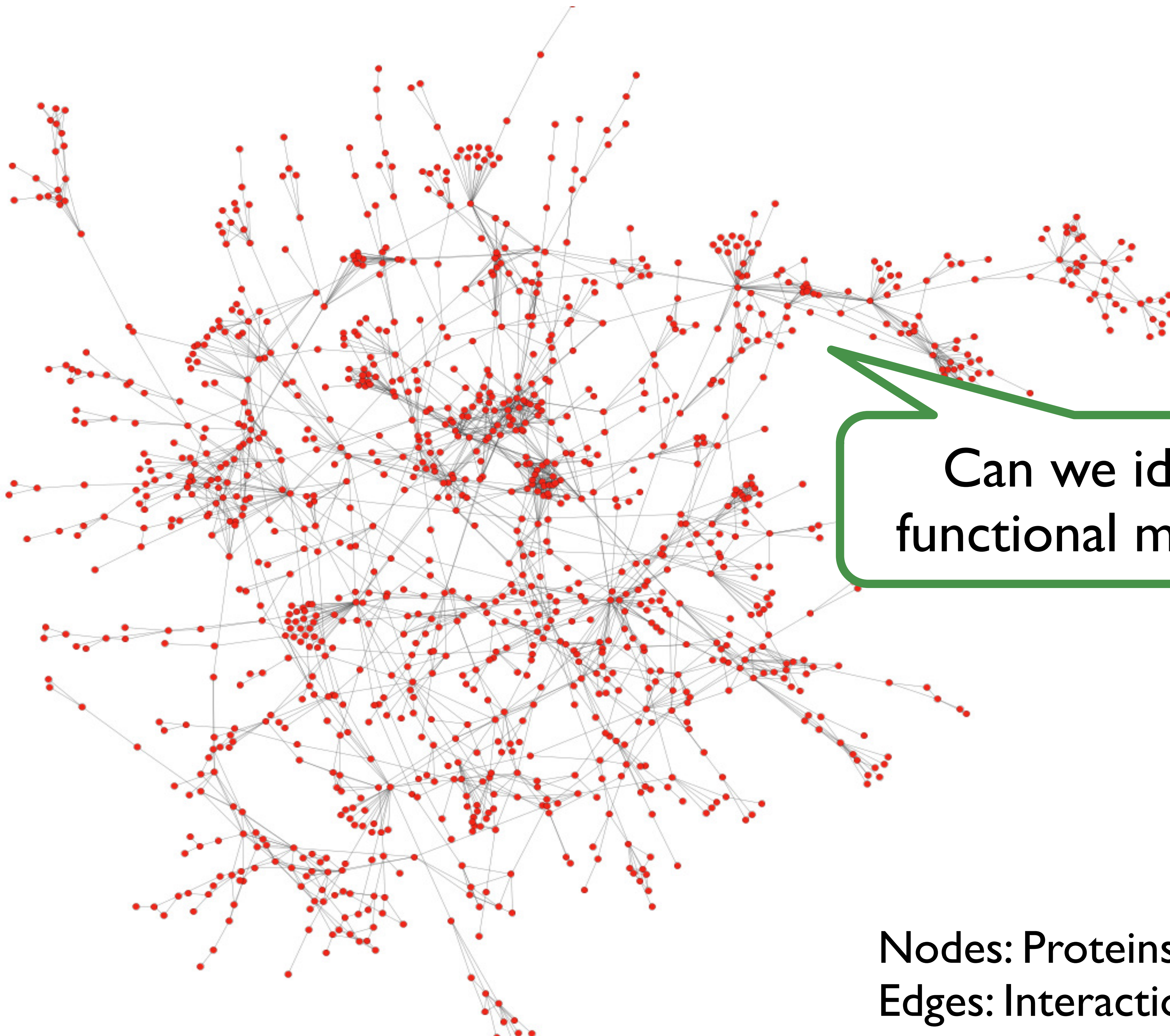


Micro-Markets in Sponsored Search

Find micro-markets by partitioning the "query x advertiser" graph:



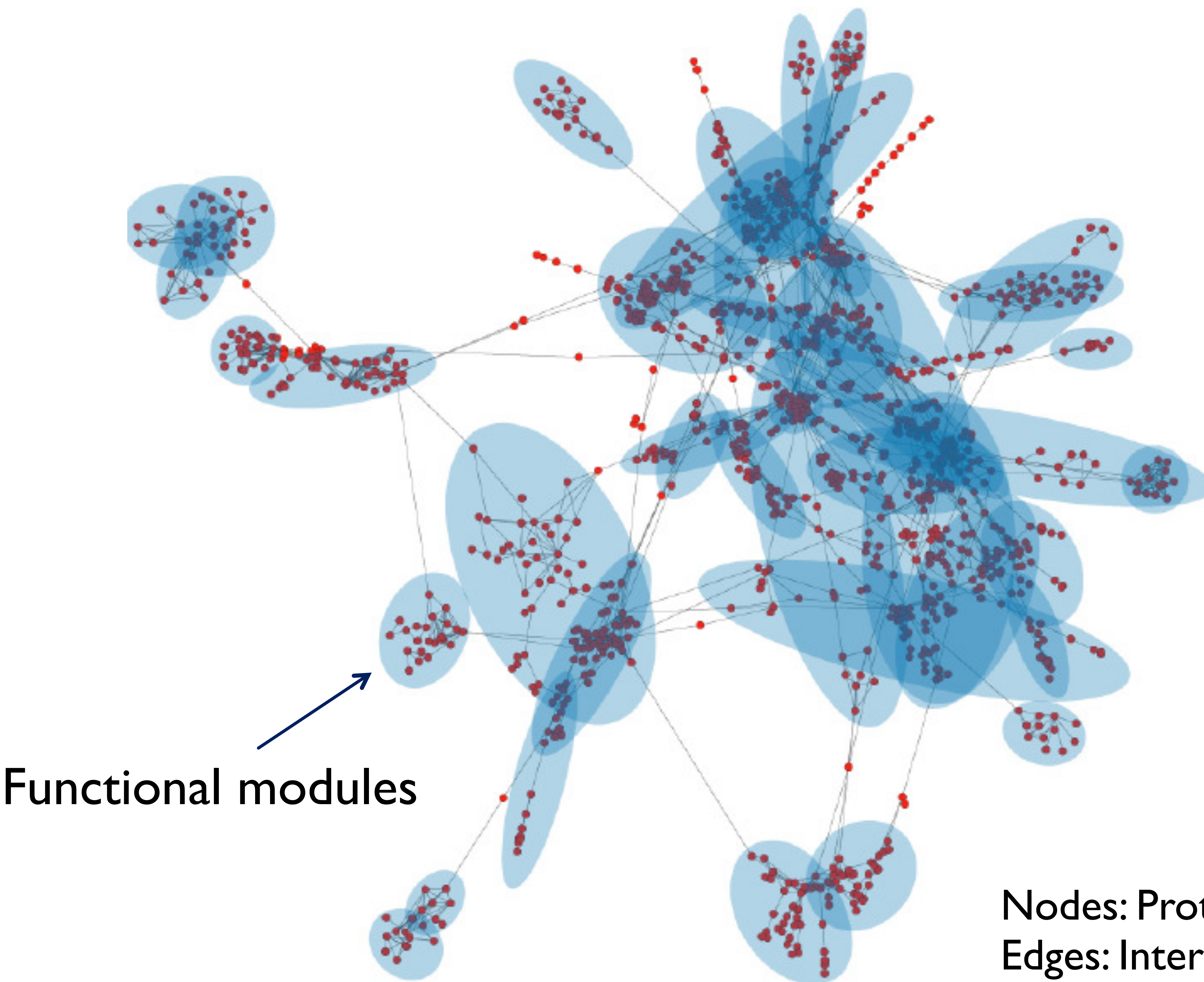
Protein-Protein Interactions



Can we identify functional modules?

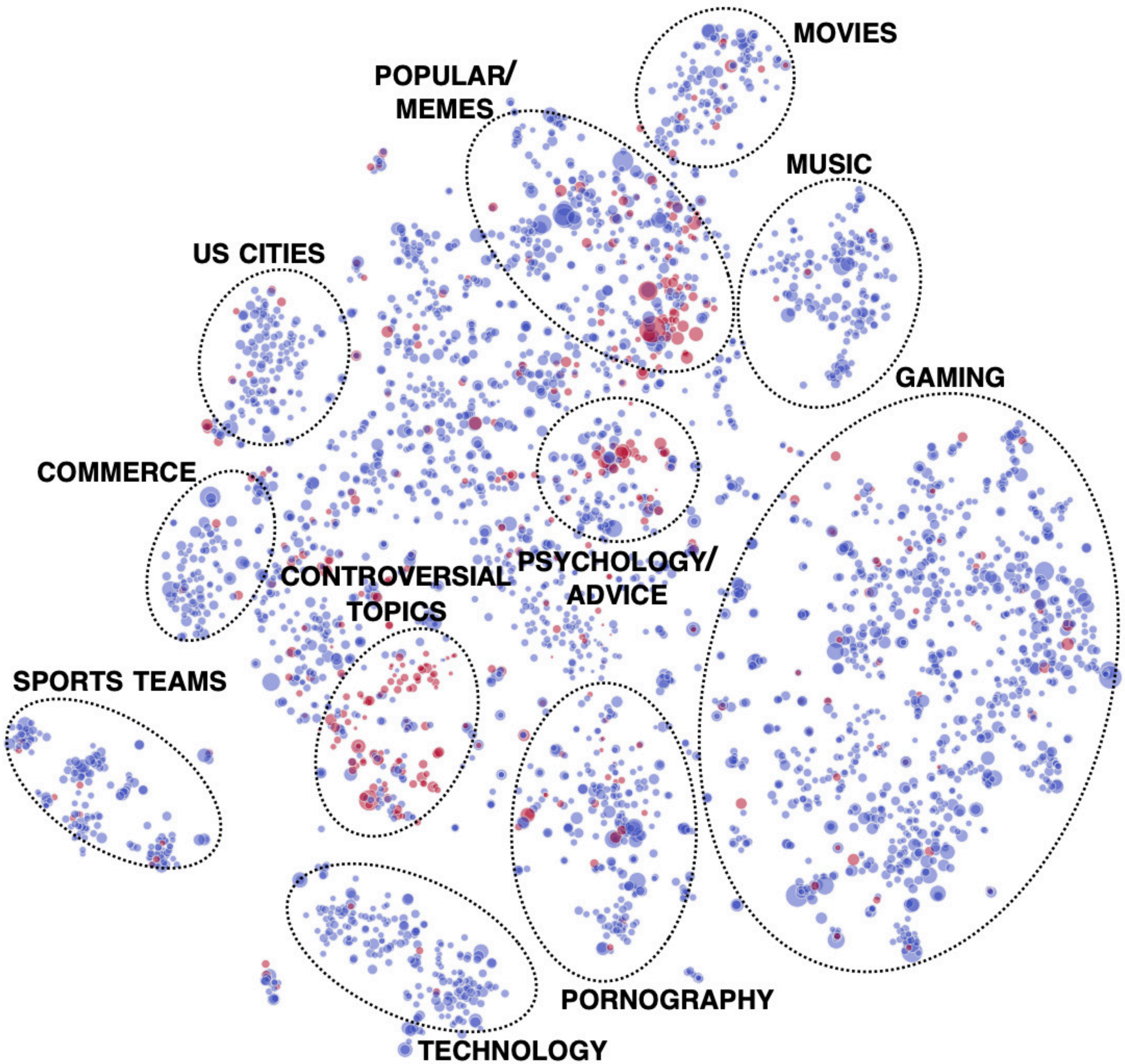
Nodes: Proteins
Edges: Interactions

Protein-Protein Interactions



Nodes: Proteins
Edges: Interactions

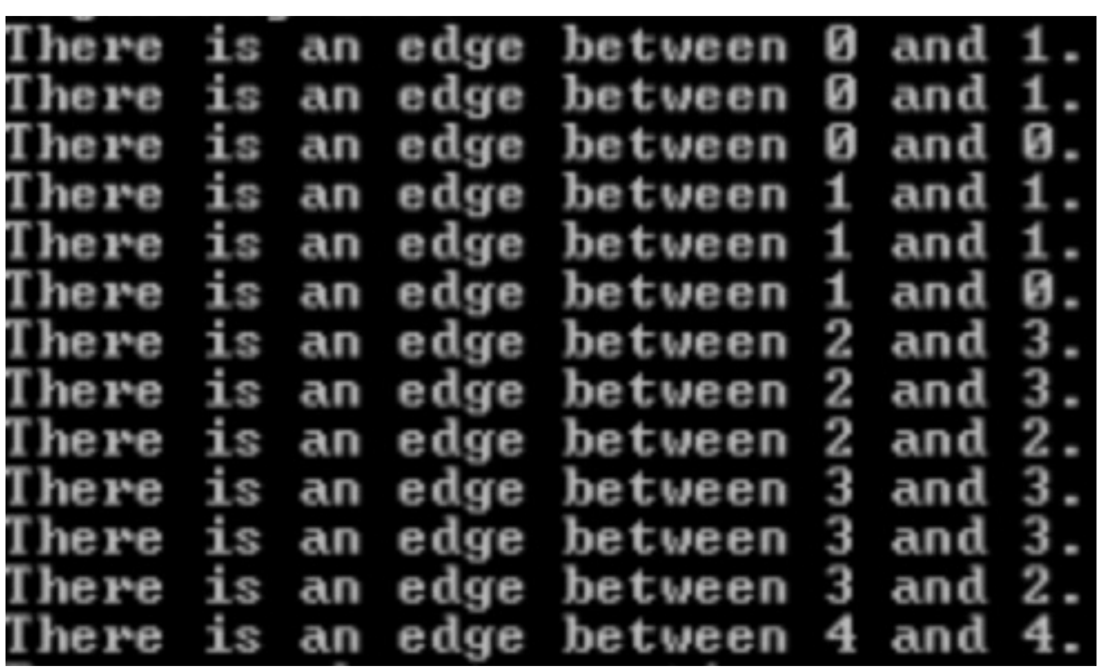
Community Structure on Reddit



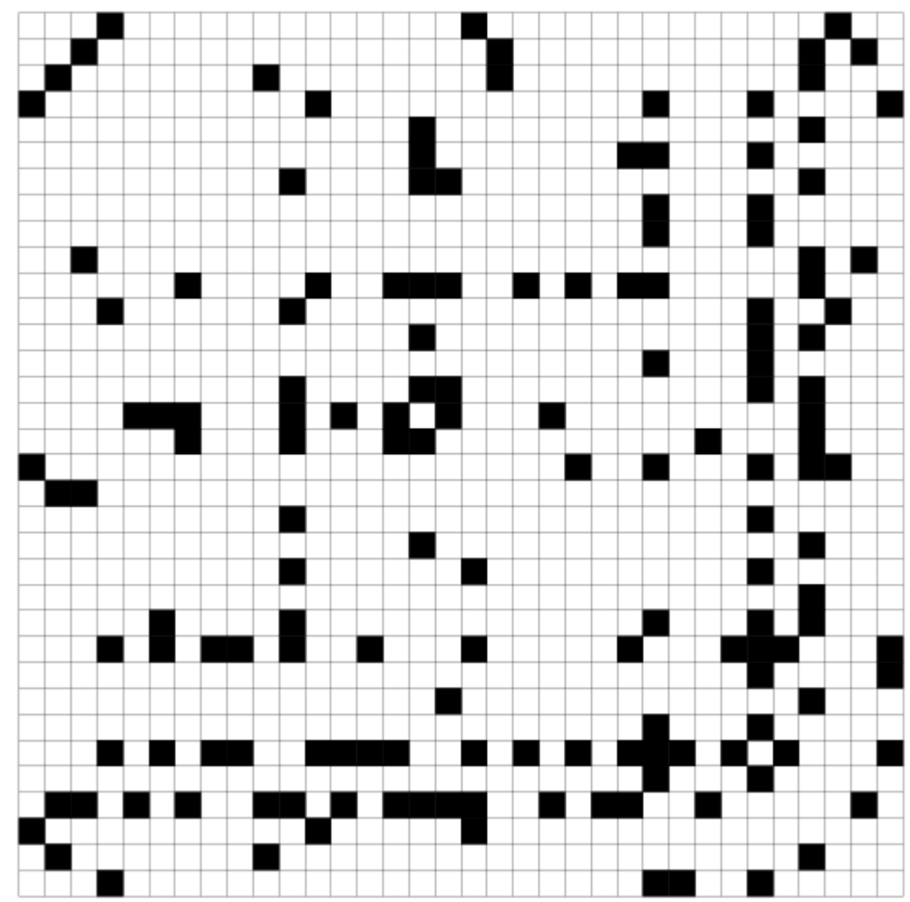
Community Structure

Many real-world networks exhibit community structure that is "obvious" to the naked eye

But what about finding communities from data?



What are the communities now?



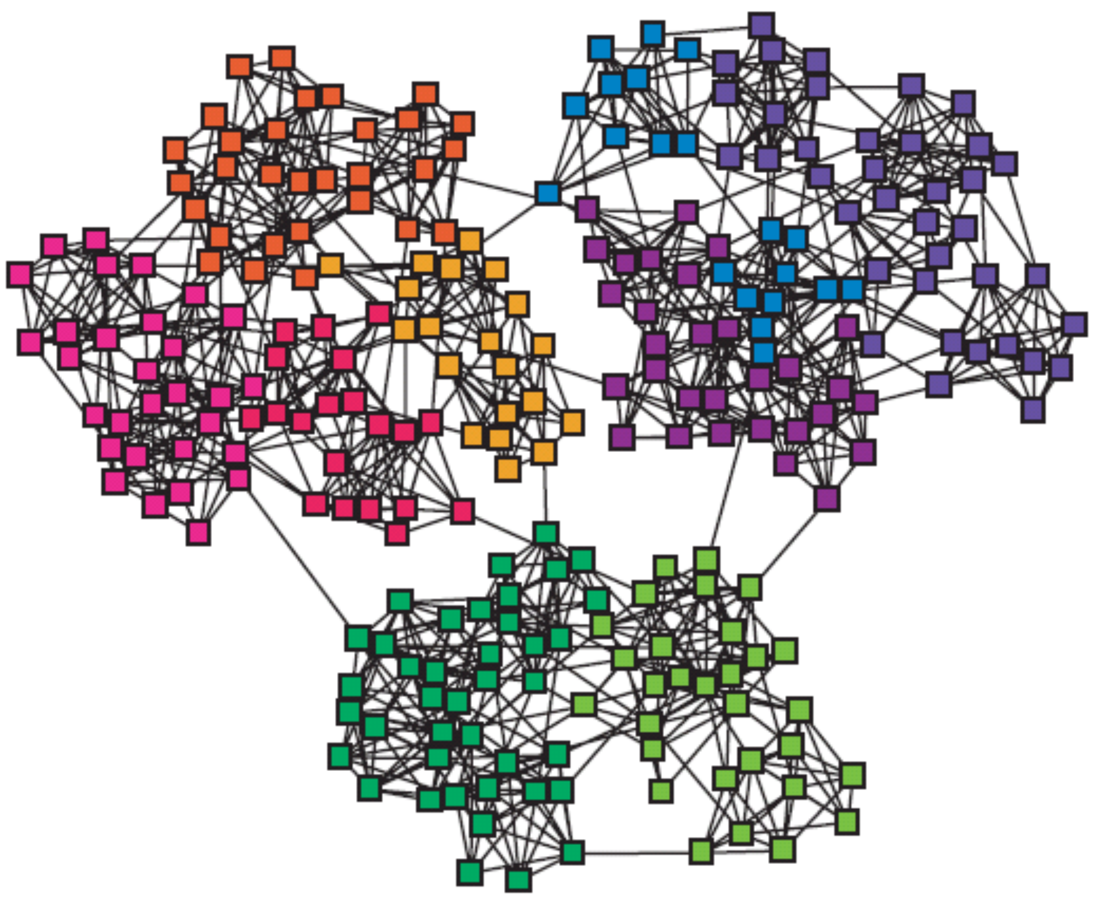
Do this with 1B edges…

Finding Network Communities

How to automatically find such densely connected groups of nodes?

Ideally such automatically detected clusters would then correspond to real groups

For example:



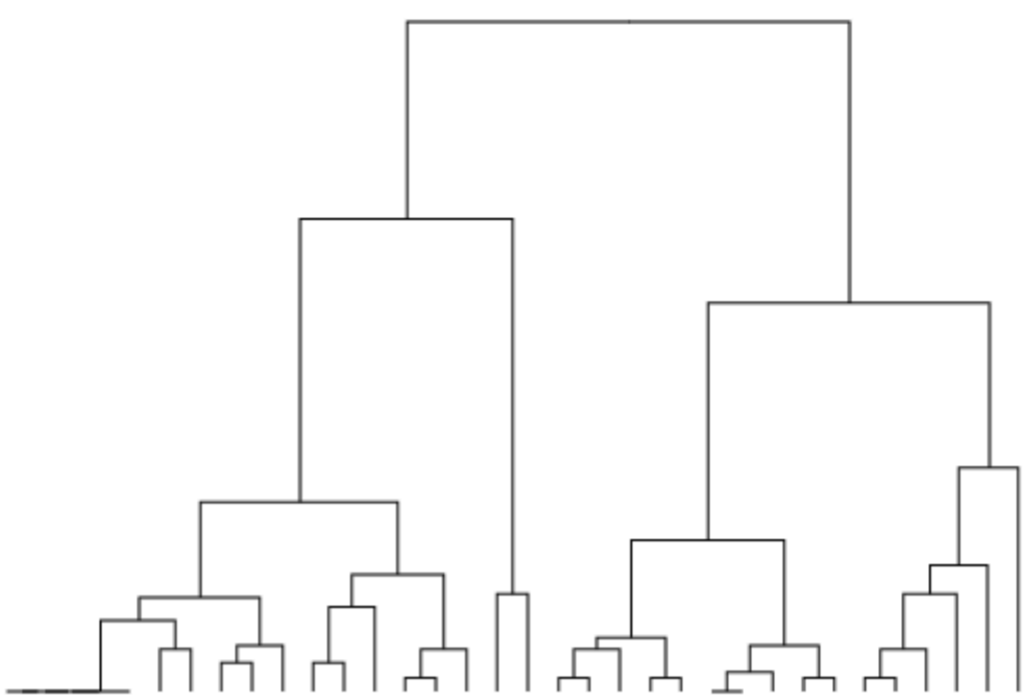
Note: We will work with undirected (unweighted) graphs

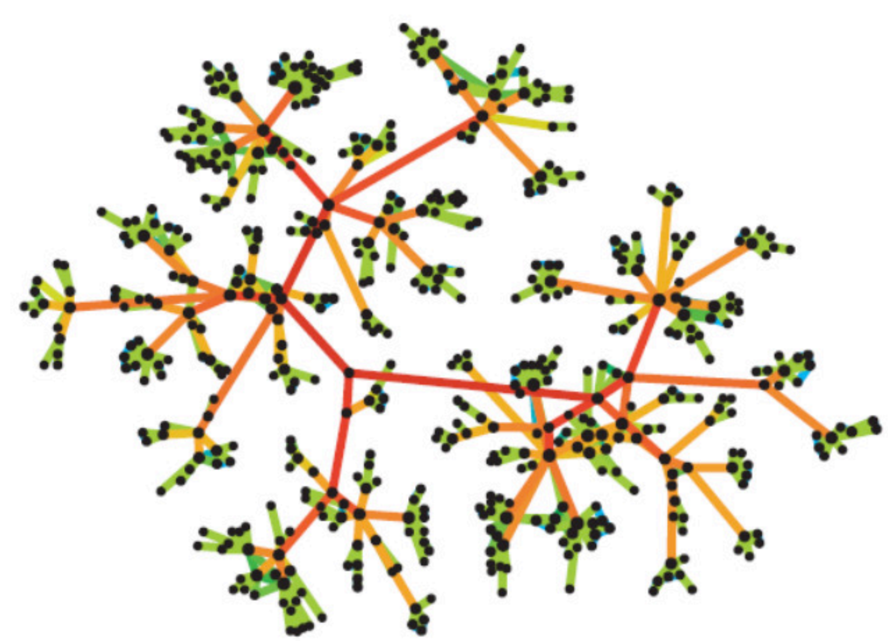
Graph Partitioning

Two general approaches:

1. Start with every node in the same cluster and break apart at "weak links" ("**divisive clustering**")

2. Start with every node in its own "community" and join communities that are close together ("**agglomerative clustering**")

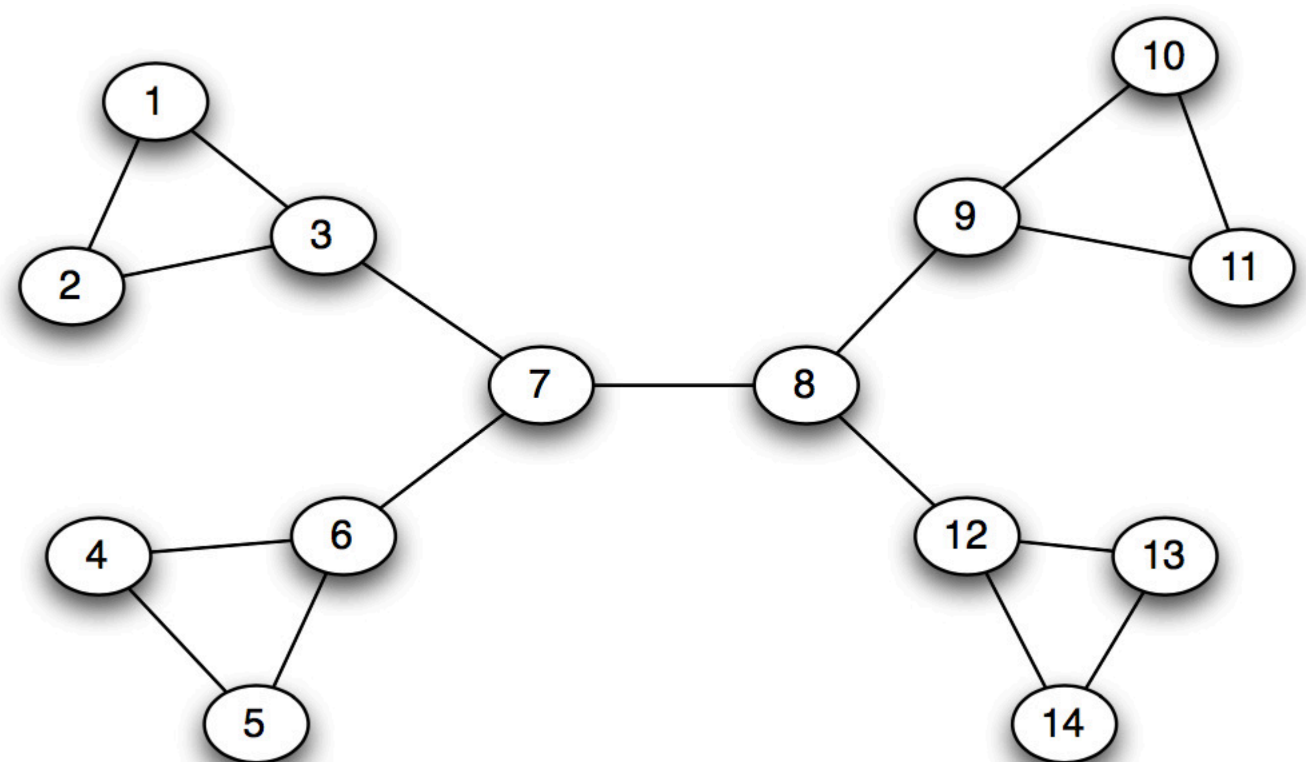




Graph Partitioning

We'll do the first: start with the whole graph as a community and recursively split it up into smaller communities

Consider the following graph:

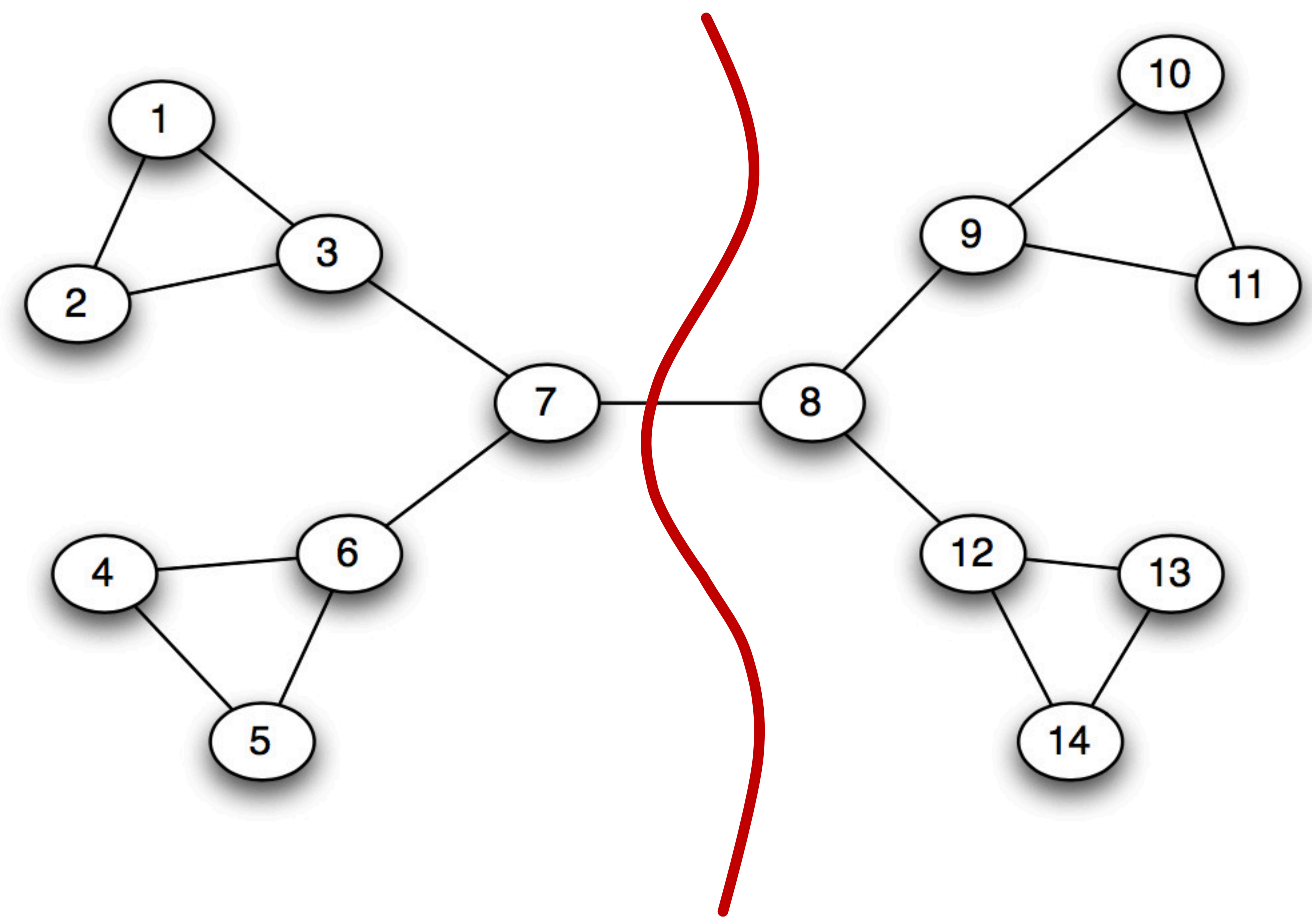


Where would you make the first cut?

Graph Partitioning

We'll do the first: start with the whole graph as a community and recursively split it up

Consider the following graph:

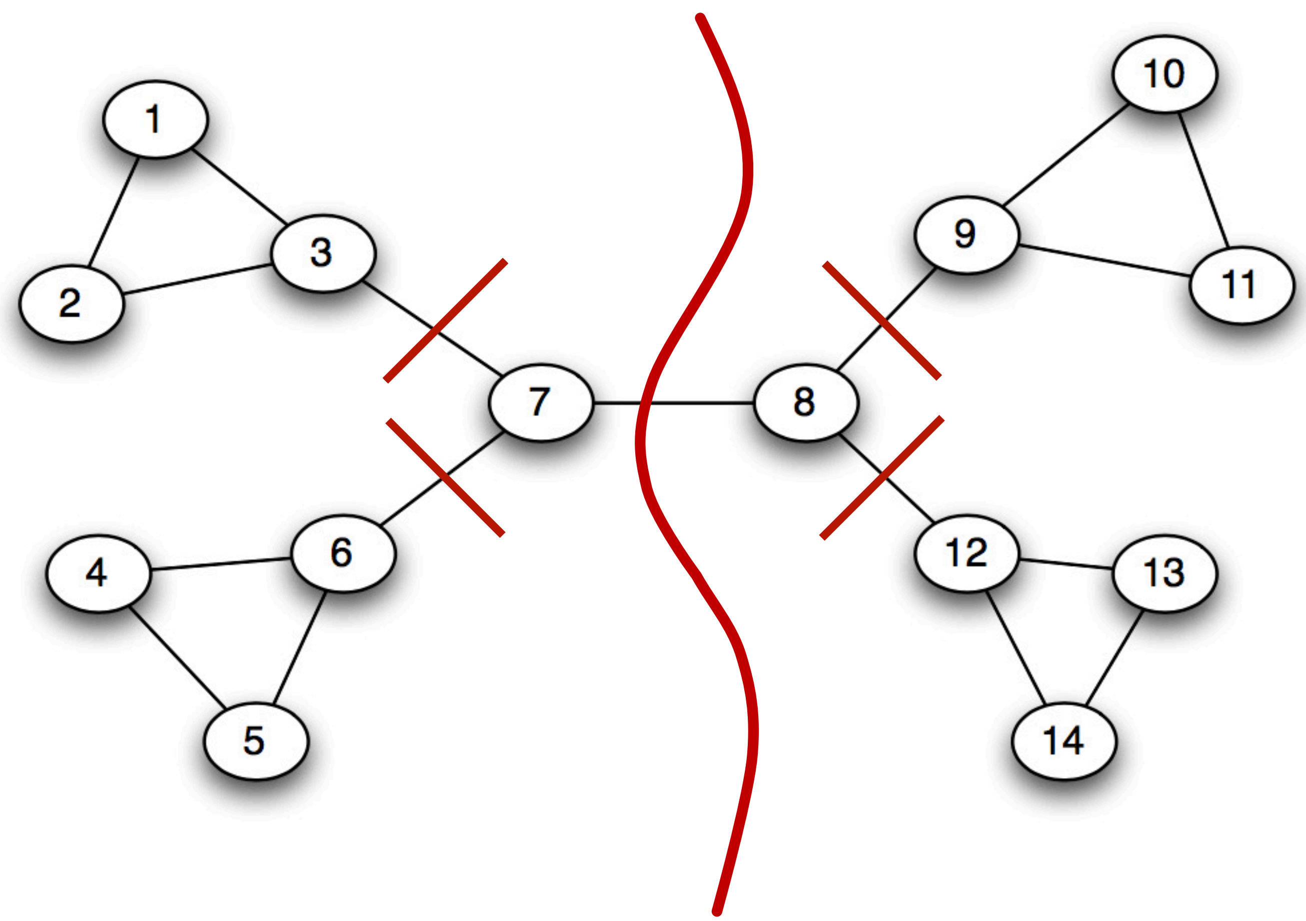


And now?

Graph Partitioning

We'll do the first: start with the whole graph as a community and recursively split it up

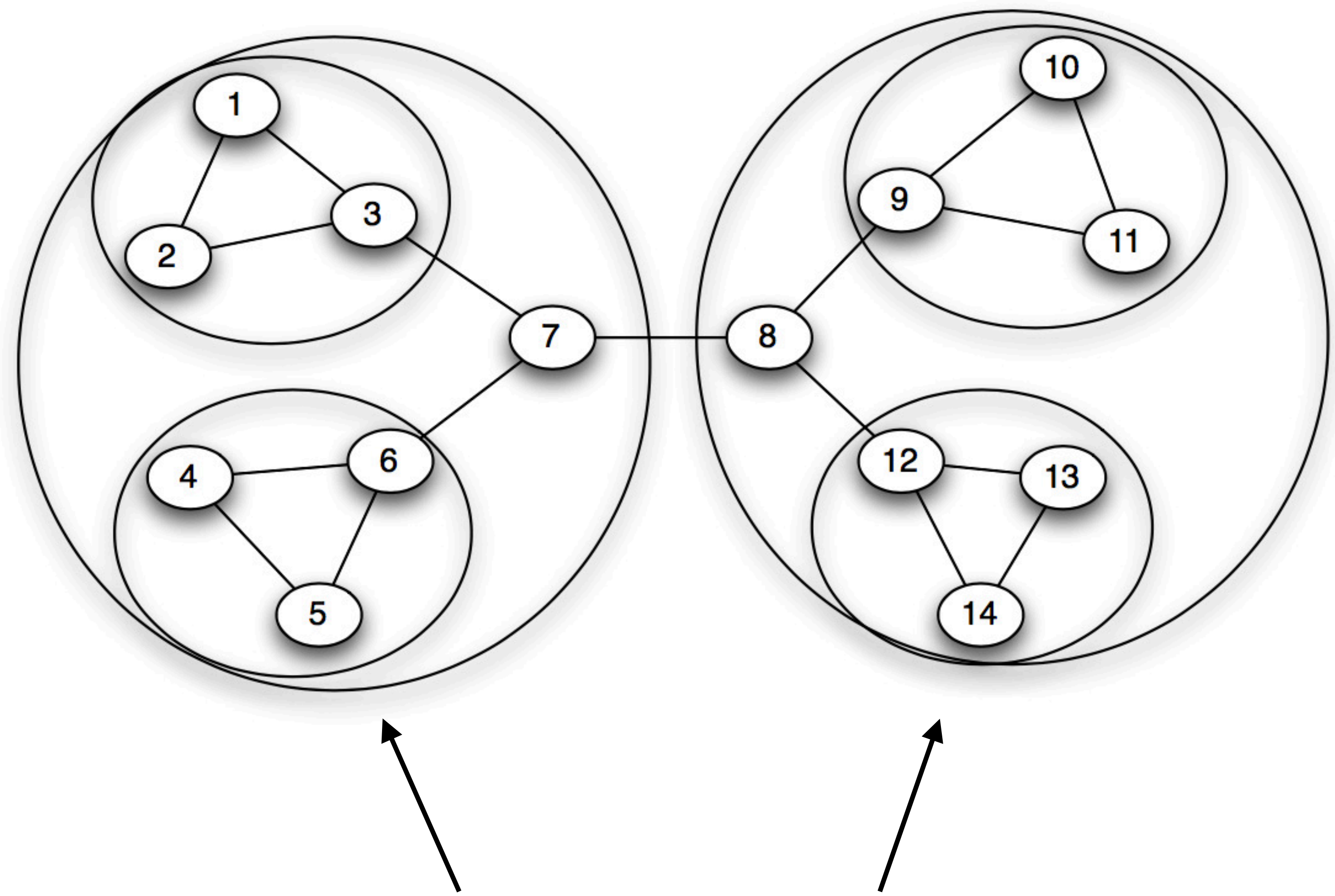
Consider the following graph:



Graph Partitioning

We'll do the first: start with the whole graph as a community and recursively split it up

Consider the following graph:

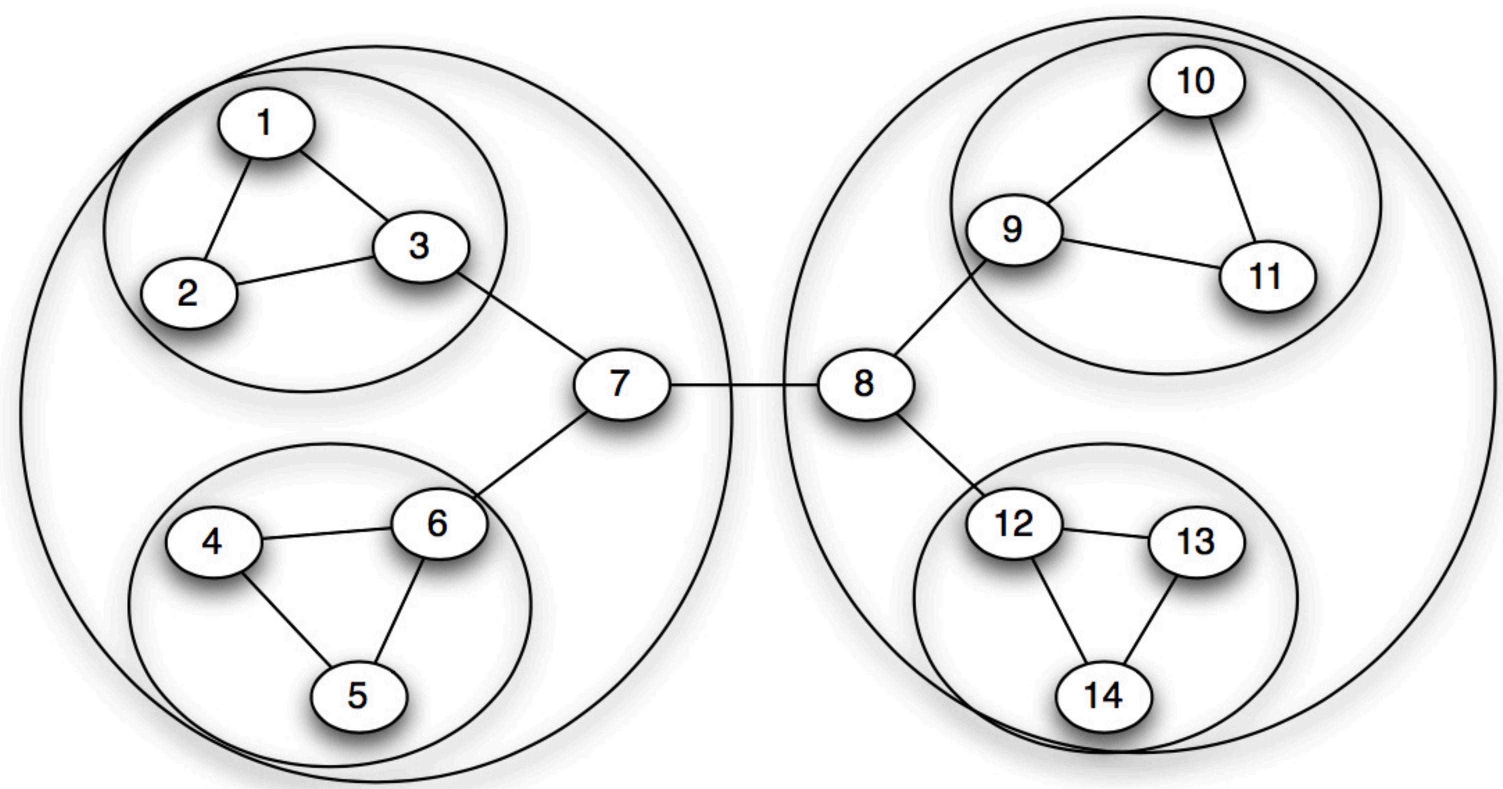


Tightly-knit regions

Graph Partitioning

This naturally produces **nested communities**

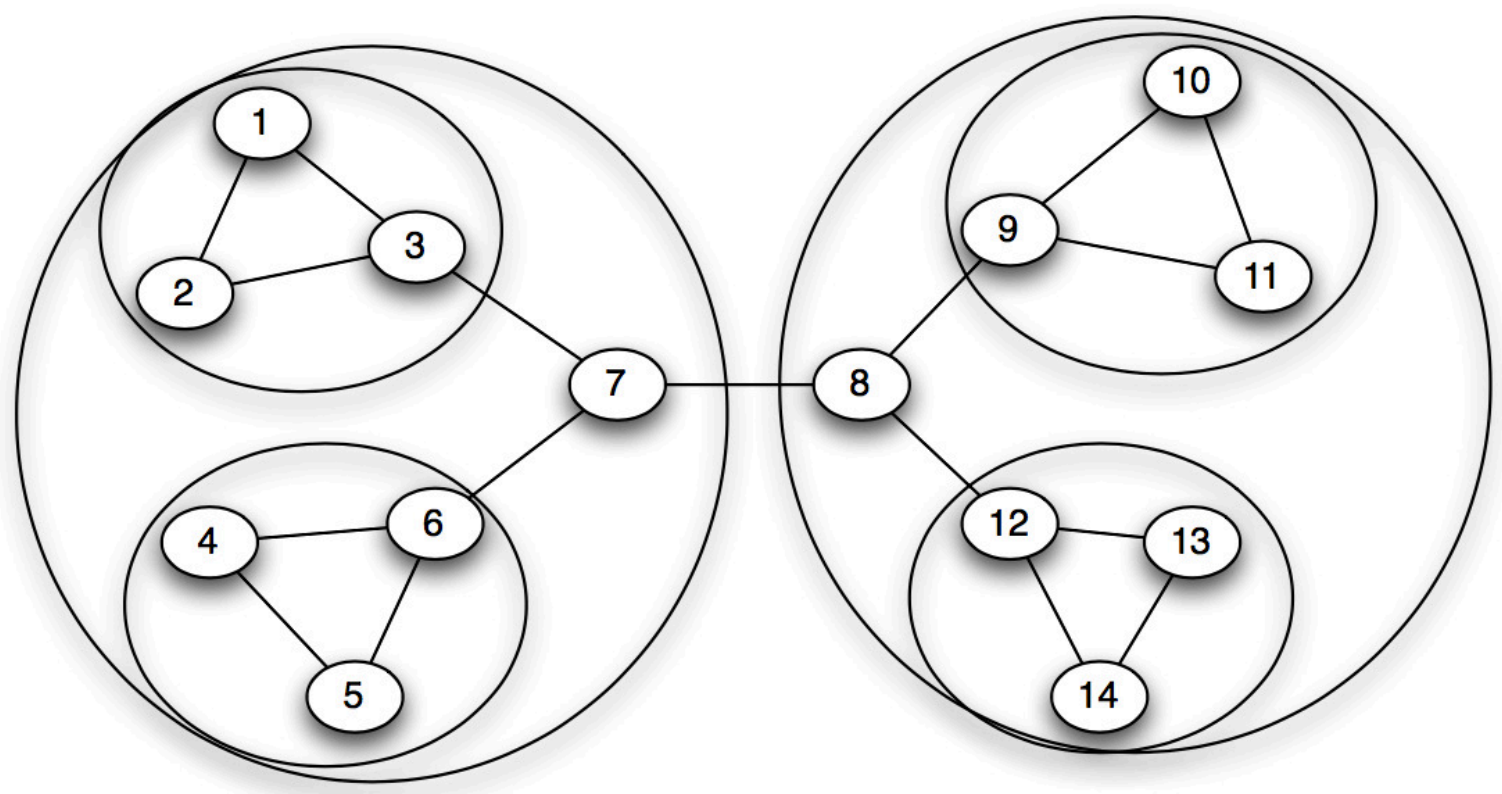
This is familiar from everyday life:
- Countries, provinces, cities…
- Sports, Arts, Business then teams, art forms, sectors



Nested structure!

Graph Partitioning

A number of **both** agglomerative and divisive clustering methods will **find this partitioning**
–Divisive will **delete 7-8 first**, etc.
–Agglomerative would **add 7-8 last**, etc.



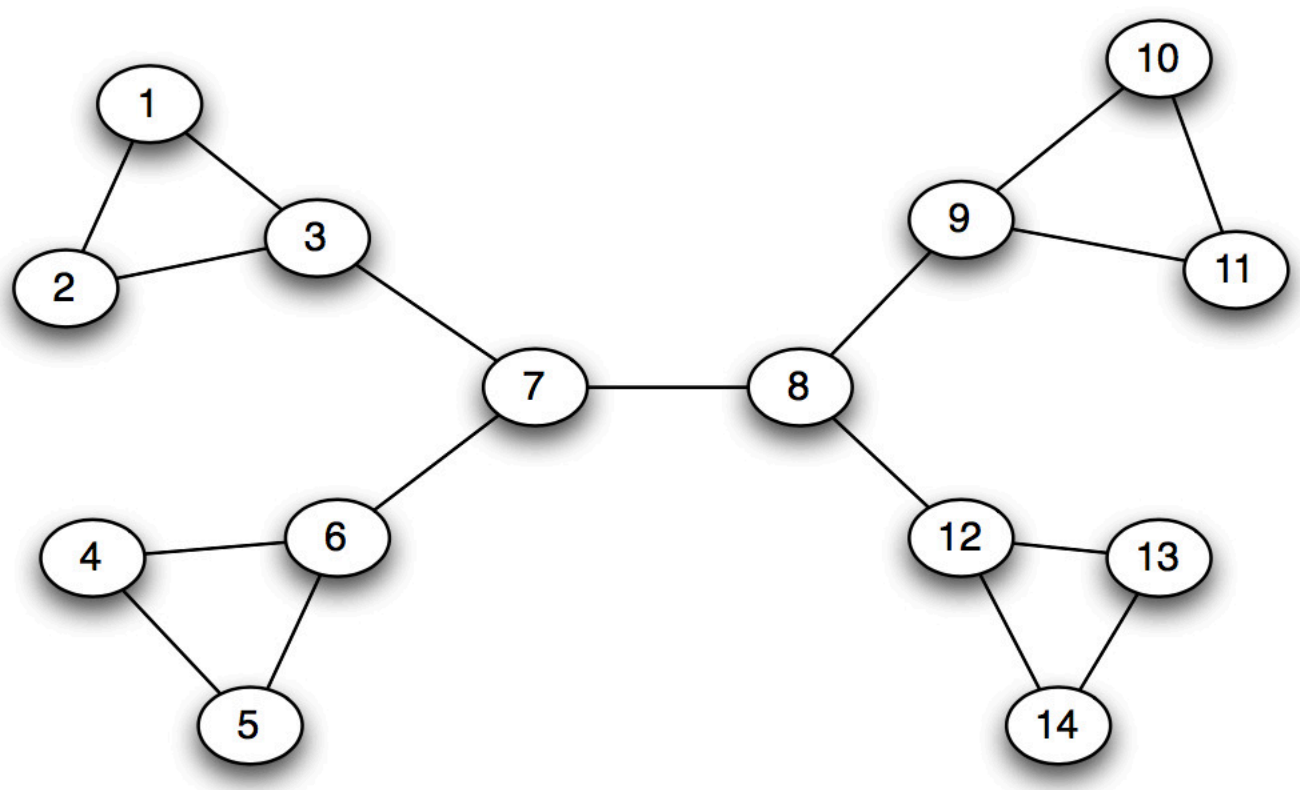
Nested structure!

Graph Partitioning

Back to divisive clustering: Why is 7-8 a good candidate for the first cut?

It is a **bridge**

Recall that a weak tie is defined as an edge that separates weakly-connected regions

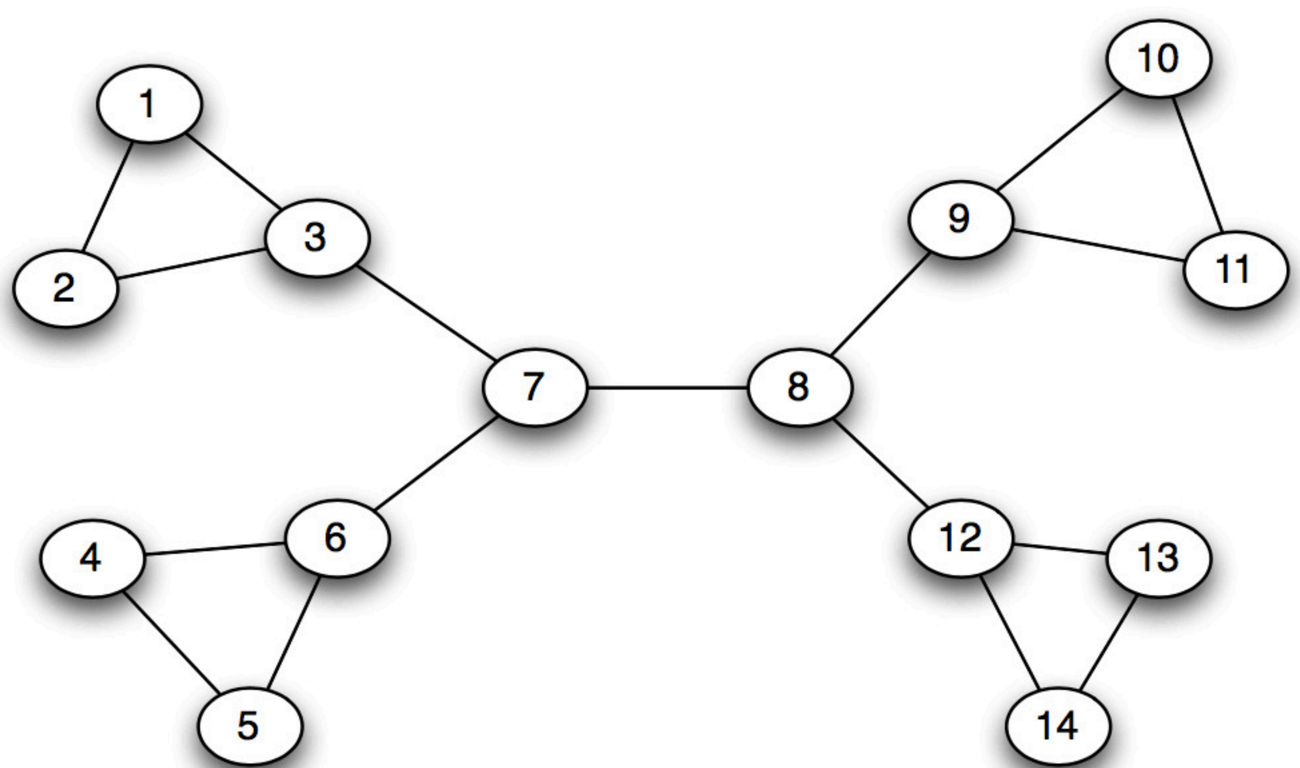


Graph Partitioning

Divisive clustering algorithm: Recursively remove bridges?

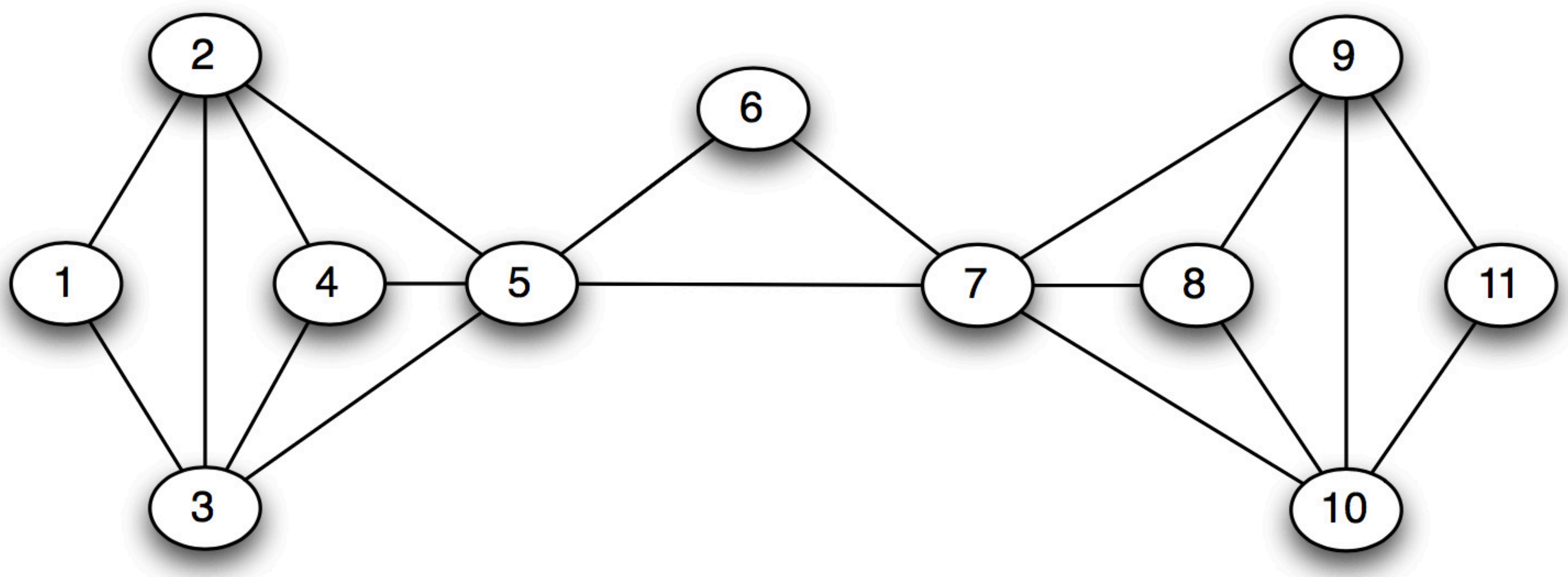
Right idea, but not strong enough: There are other bridges too (which ones?)

3-7, 6-7, 8-9, 8-12 are also bridges!



Graph Partitioning

Also, sometimes there are no bridges (or even no local bridges) **but "natural" communities still exist**

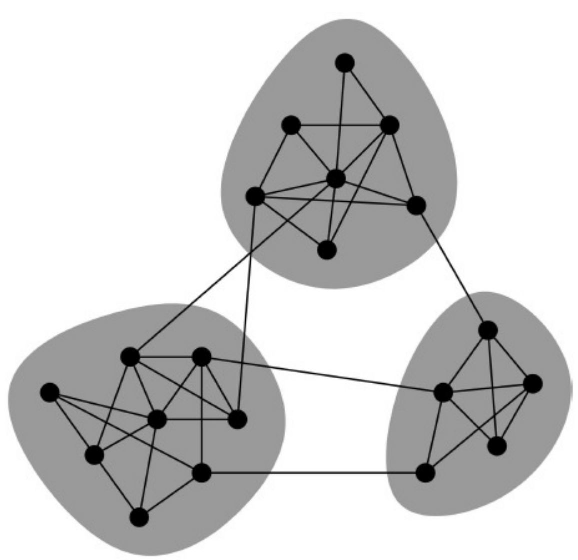


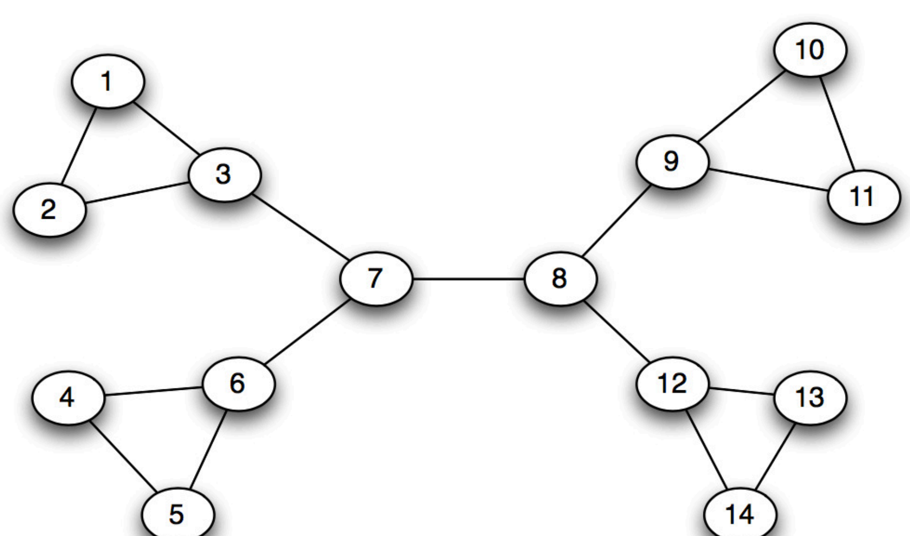
Graph Partitioning

Recall definition of a **bridge**: an edge that, if you remove it, disconnects its endpoints

Thus it is **an edge that carries a shortest path** (obviously the shortest, since it's also the only path)

Need a more nuanced definition to distinguish bridges and "bridge-like" edges from highly embedded edges

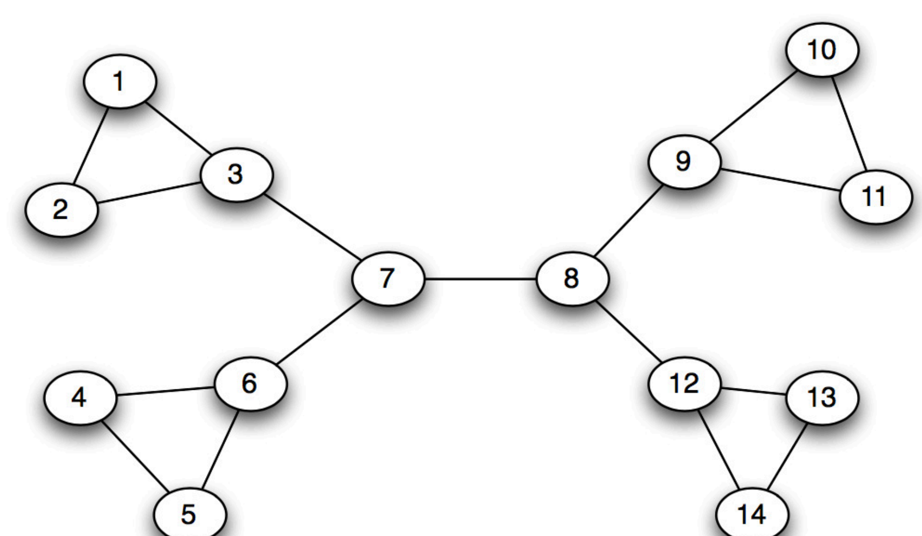




Graph Partitioning

Definition: the **betweenness** of an edge is how many (fractional) shortest paths travel through it

–For every pair of nodes A,B say there is one unit of "flow" along the edges from A to B
–Flow between A to B divides evenly among all shortest paths from A to B
–If k shortest paths, 1/k flow on each path

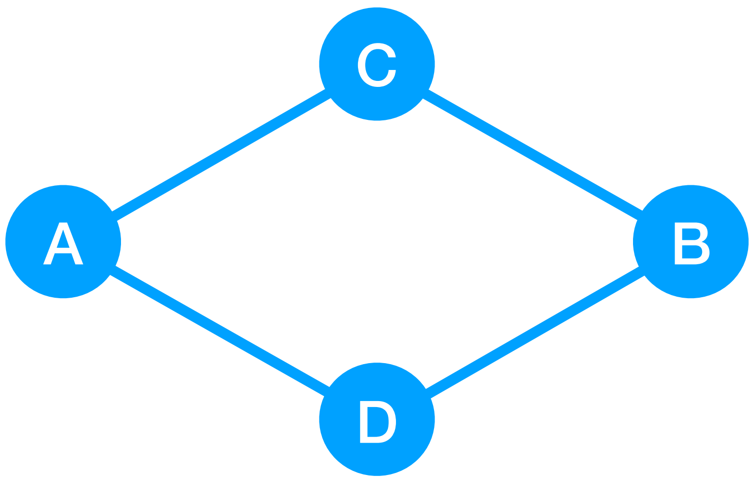


Graph Partitioning

One unit of flow from A to B

Betweenness(A–B) = 1

Graph Partitioning



One unit of flow from A to B

Two shortest paths from A to B, split evenly among them

So edges a–c, c–b, a–d, d–b get 1/2 flow each from the (A,B) pair

…and repeat for one unit of flow between every other pair of nodes:

(A,C), (A,D), (B,C), (B,D), (C,D)

Girvan-Newman algorithm

Divisive hierarchical clustering based on the notion of edge **betweenness** (Number of shortest paths passing through an edge)

Girvan-Newman Algorithm (on undirected unweighted networks)**:**

Repeat until no edges are left:

- –(Re)calculate betweenness of every edge
- –Remove edges with highest betweenness (if ties, remove all edges tied for highest)
- –Connected components are communities

Gives a hierarchical decomposition of the network

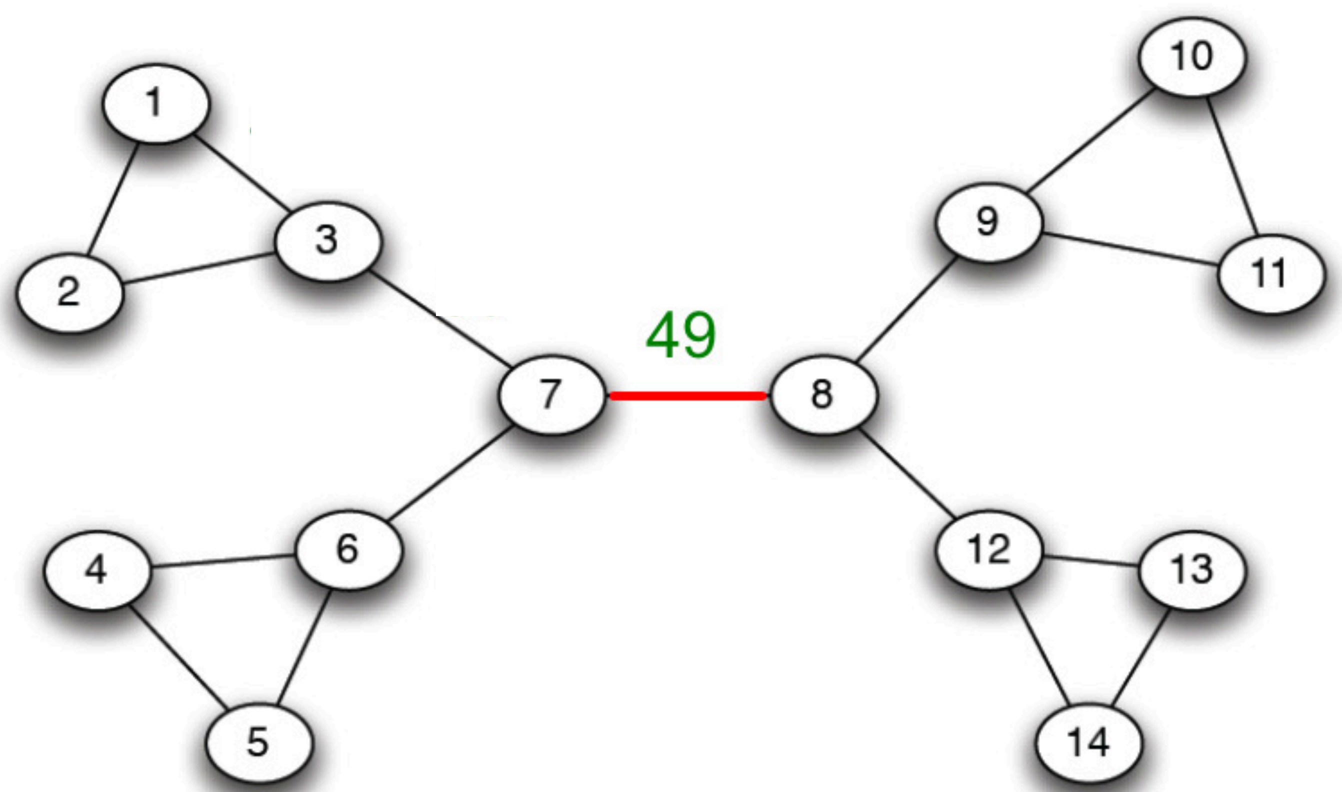
Girvan-Newman: Example

Consider edge 7-8:

–Each node A on left and node B on right has shortest path passing through 7-8

–No flow passing between nodes on same side passes through 7-8

–Betweenness(7-8) = 7x7 = 49



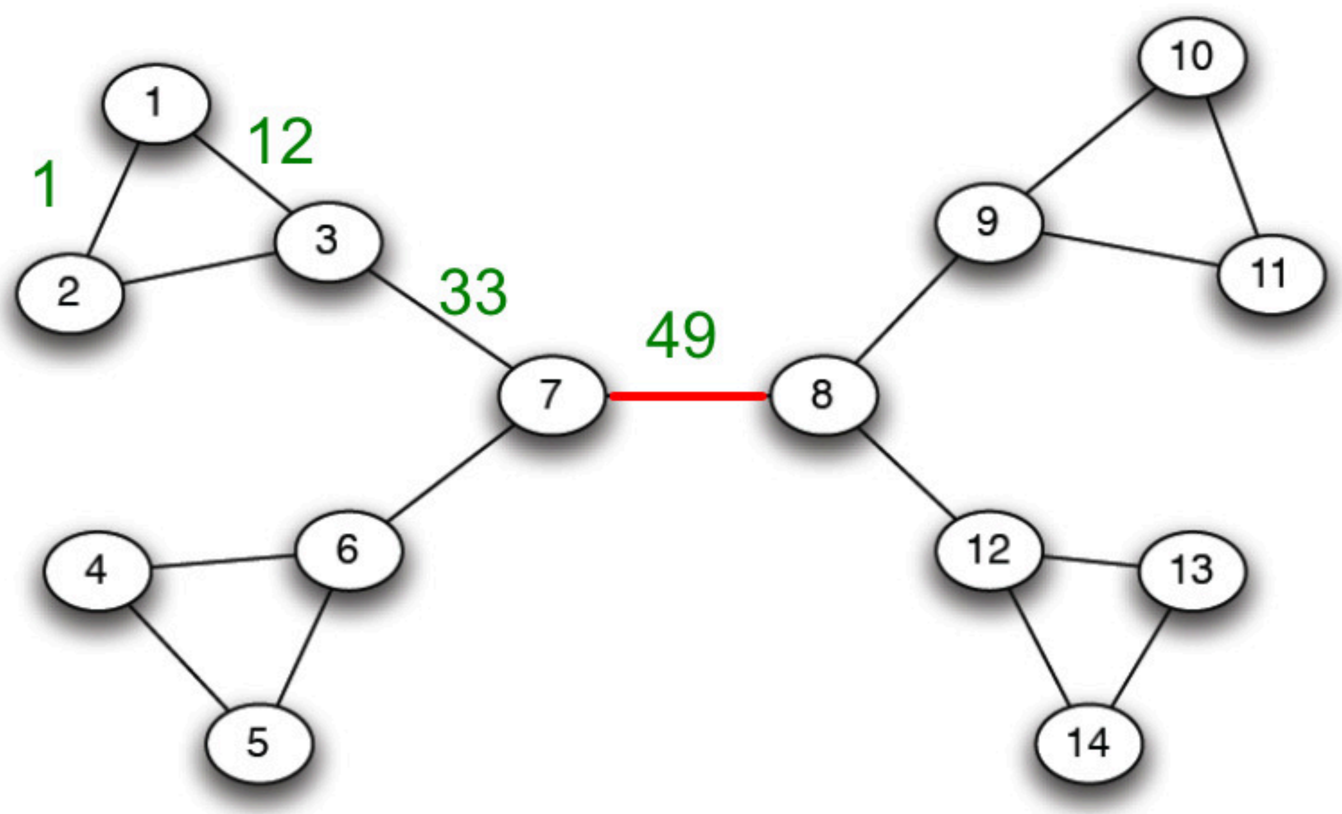
Girvan-Newman: Example

Other edges:

3-7 carries full flow from 1,2,3 to 4-14: 3x11=33

1-3 carries all flow from 1 to everyone else except 2: 1x12 = 12

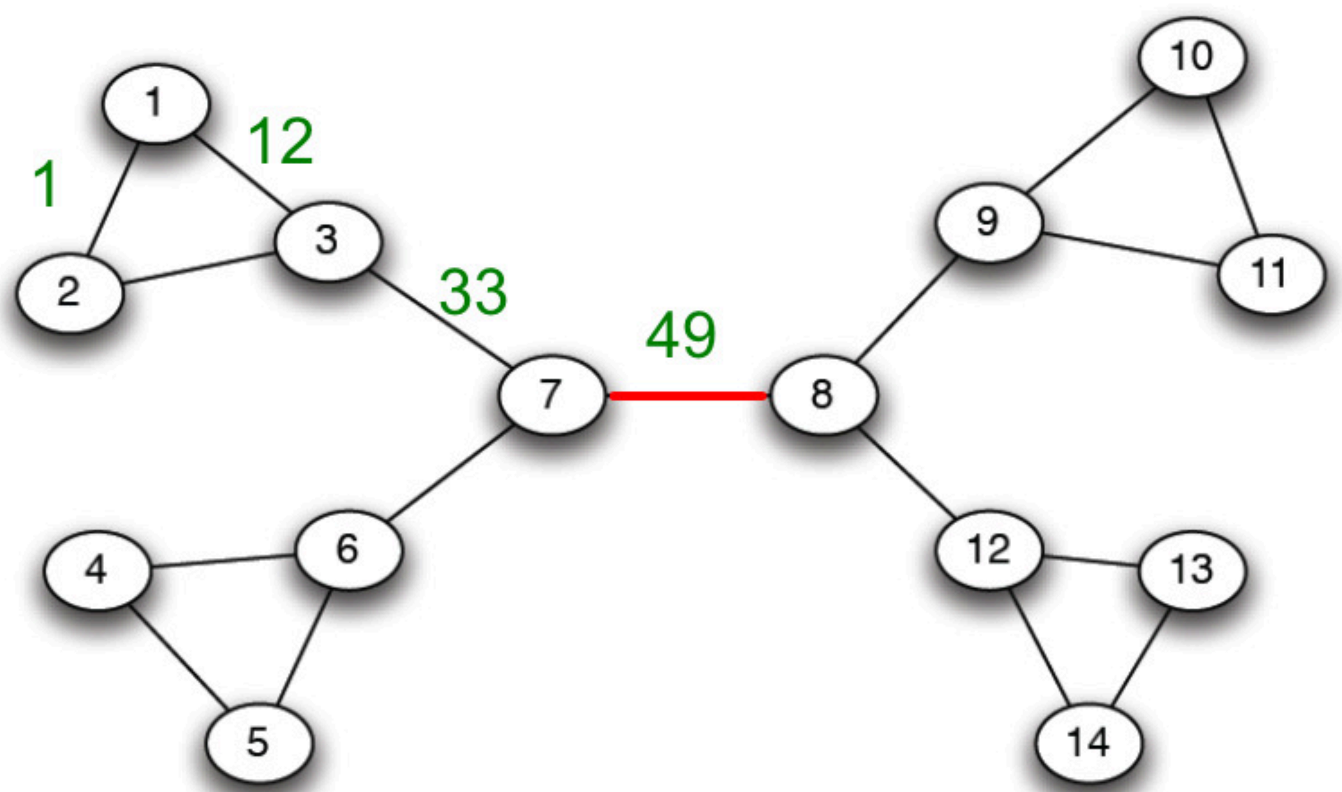
1-2 only carries flow from 1 to 2: 1x1 = 1



By symmetry, we know betweenness for all other nodes as well in this graph

Girvan-Newman: Example

Girvan-Newman method: Remove edge of highest betweenness (or multiple if there is a tie)

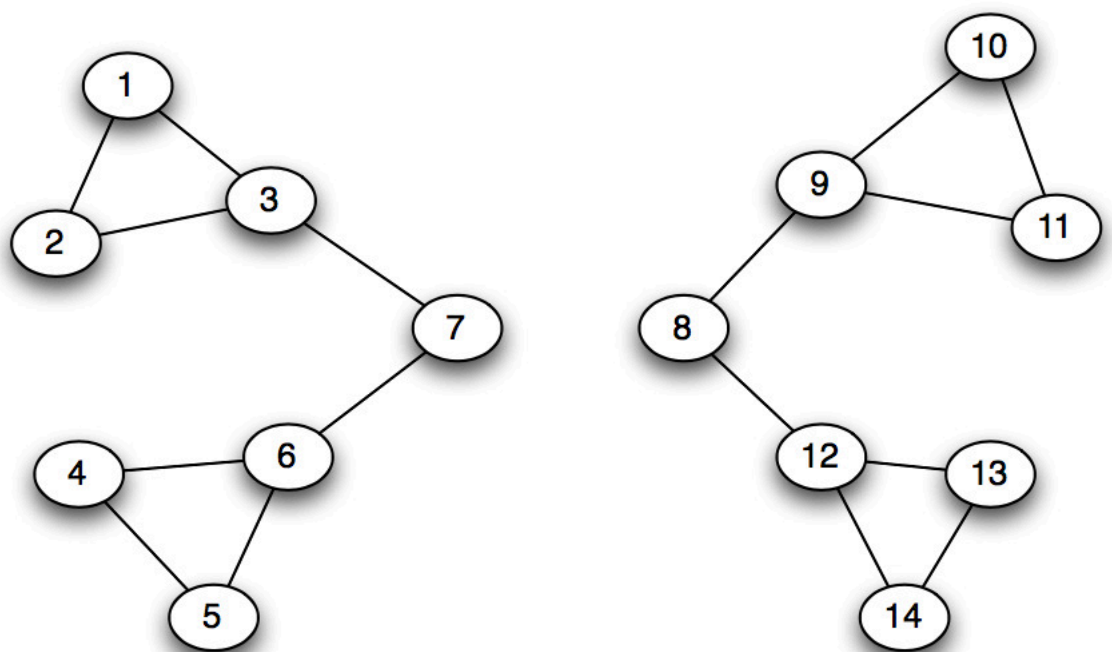


By symmetry, we know betweenness for all other nodes as well in this graph

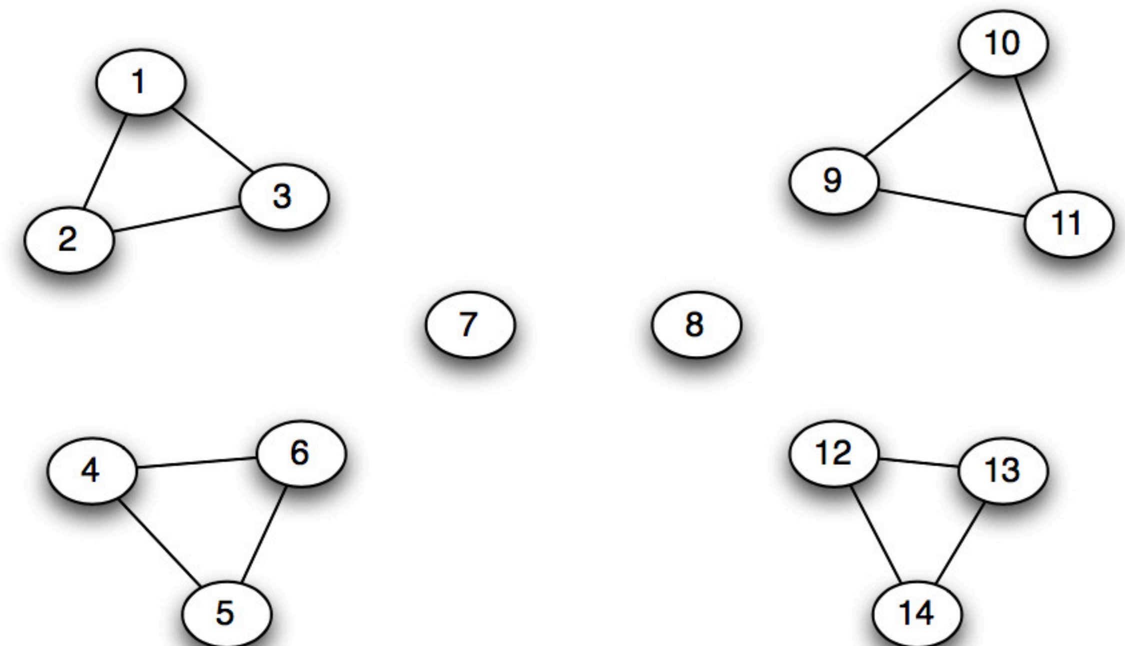
Girvan-Newman: Example

Need to re-compute
betweenness at every step

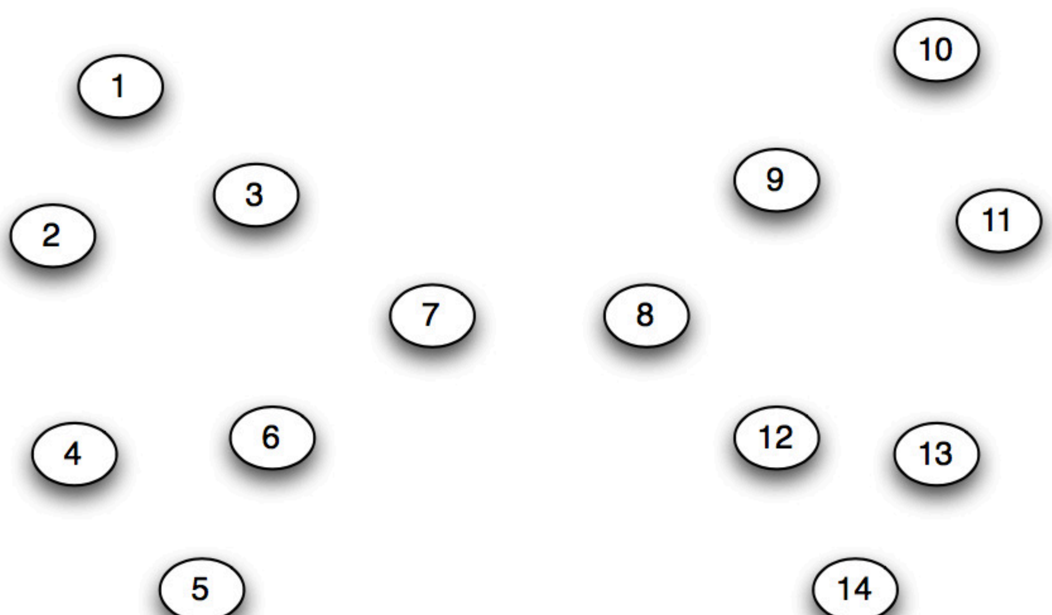
Step 1:



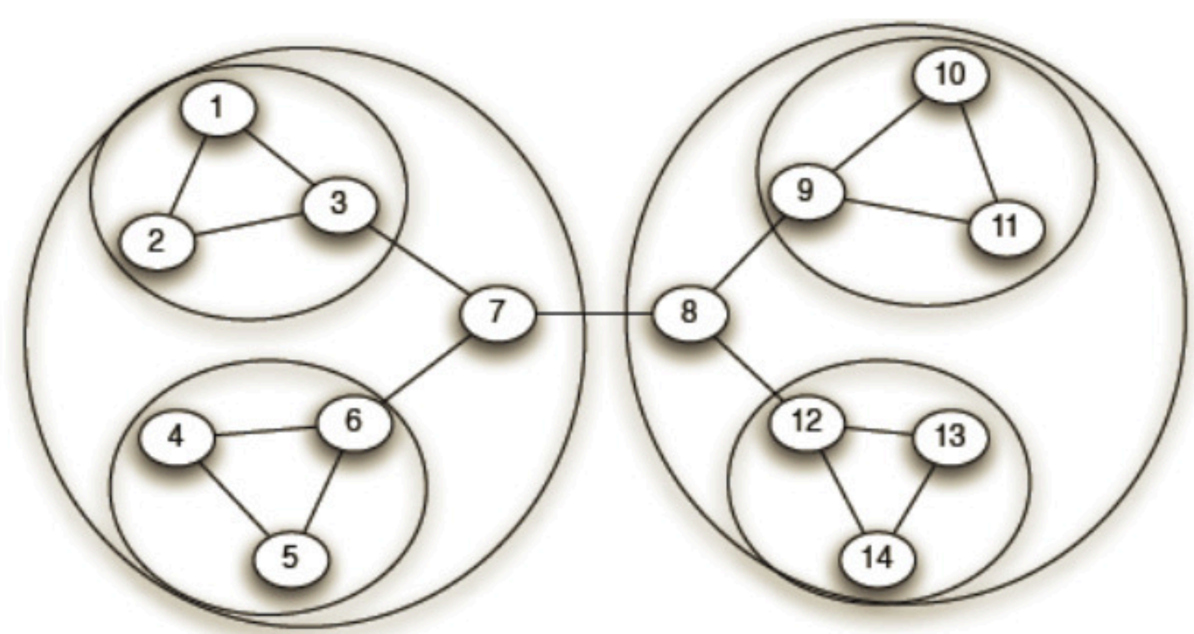
Step 2:



Step 3:



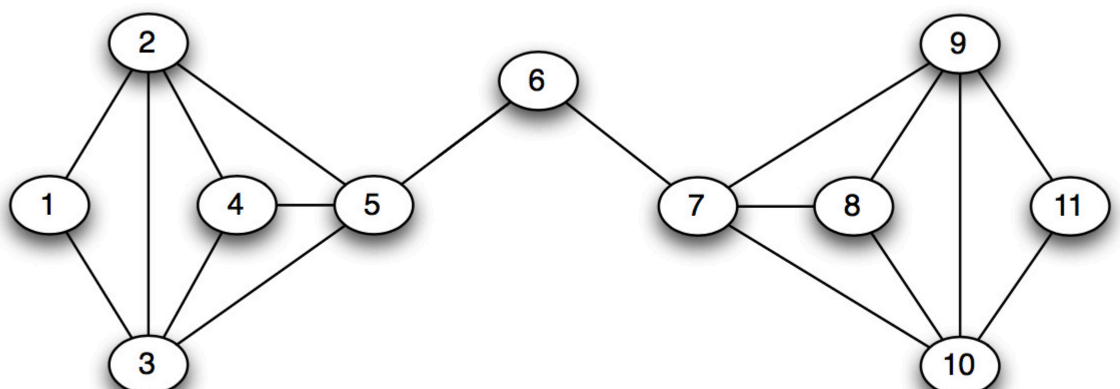
Hierarchical network decomposition:



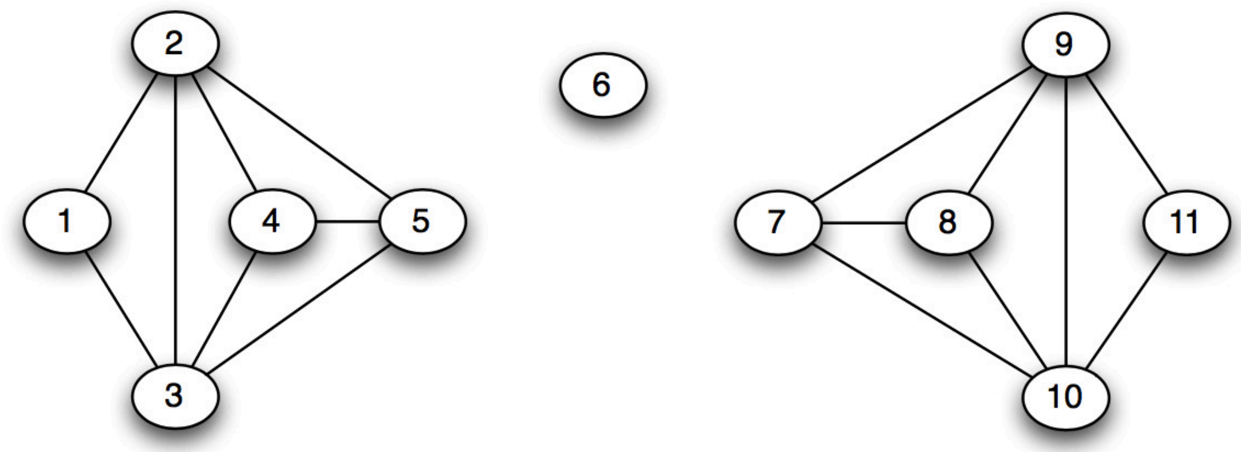
Girvan-Newman: Example

Need to re-compute betweenness at every step

Step 1:

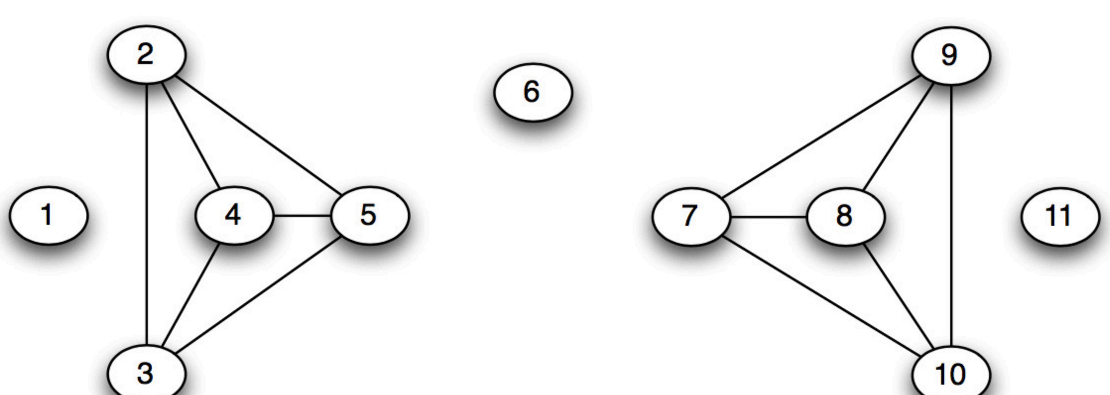


Step 2:

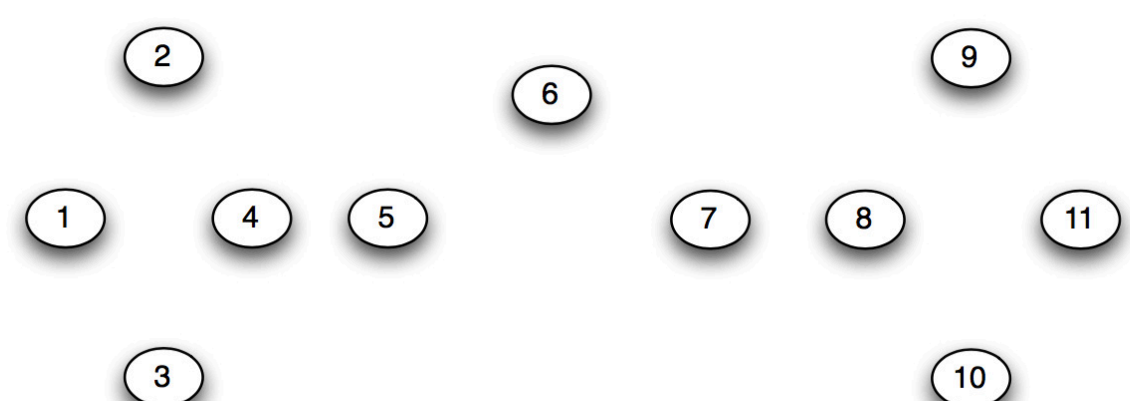


25 units that used to be on 5-7 get shifted to 5-6 and 6-7

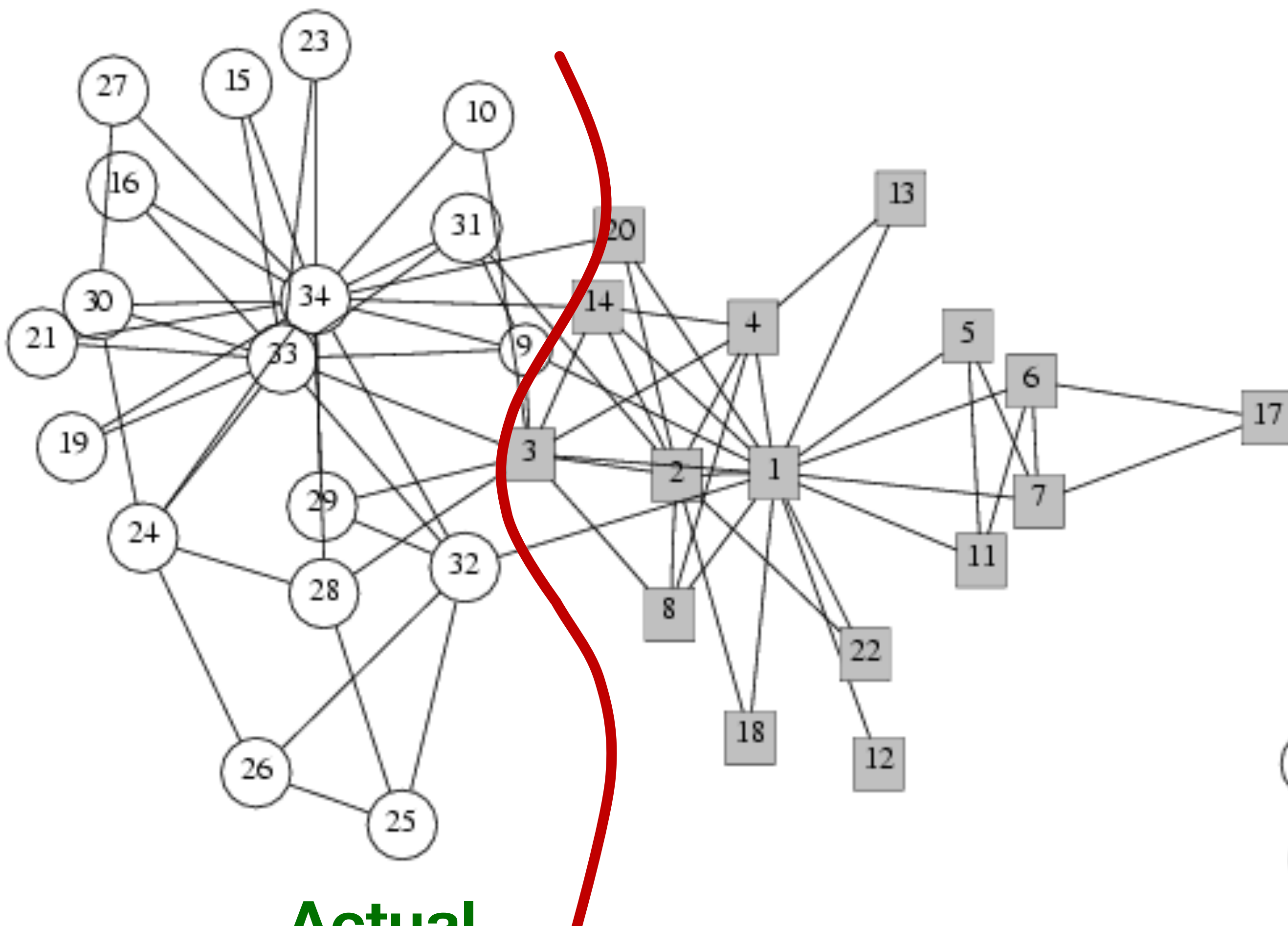
Step 3:



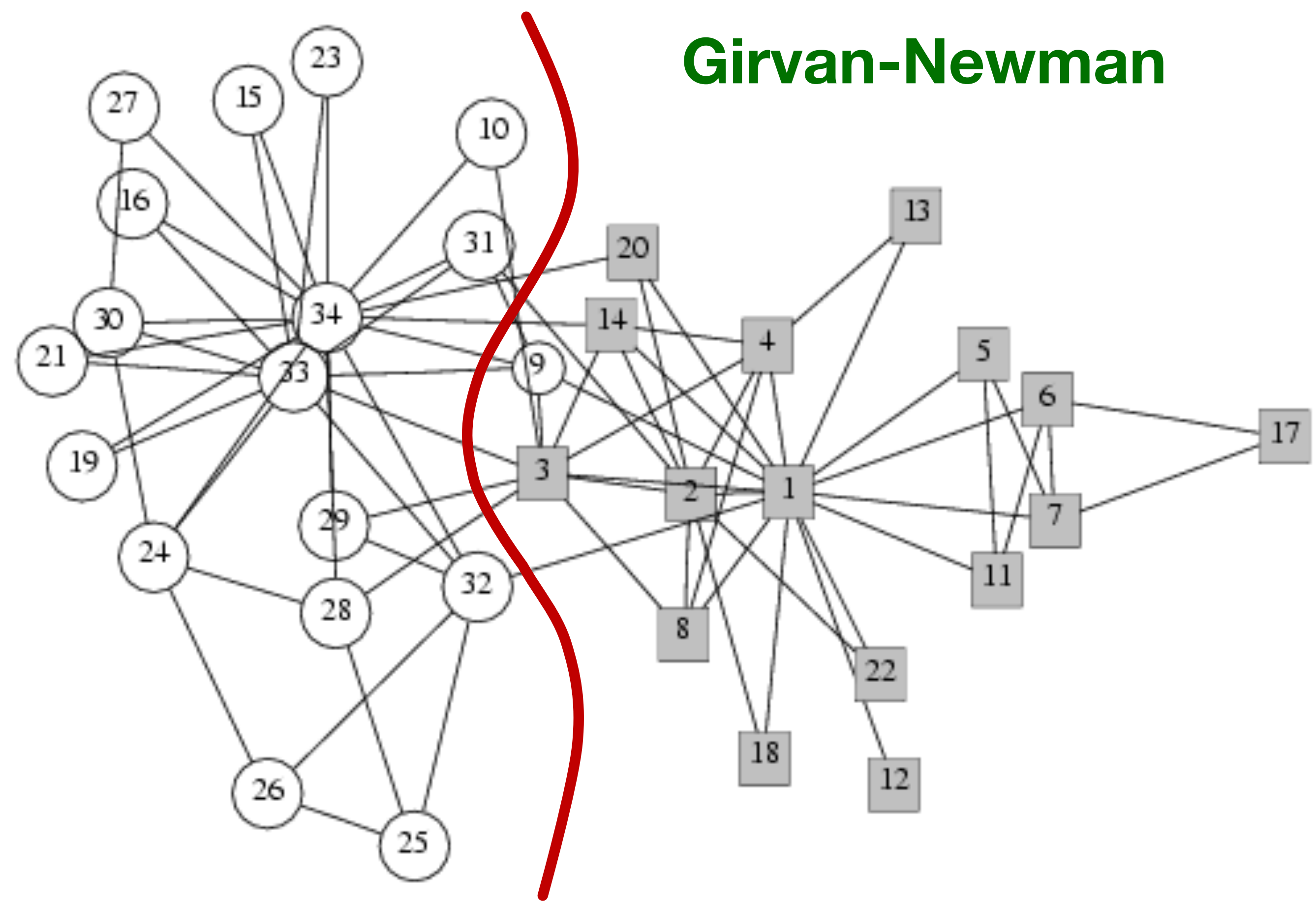
Step 4:



Zachary Karate Club



Actual



Girvan-Newman

Visualizing Hierarchical Clusters

Dendrogram

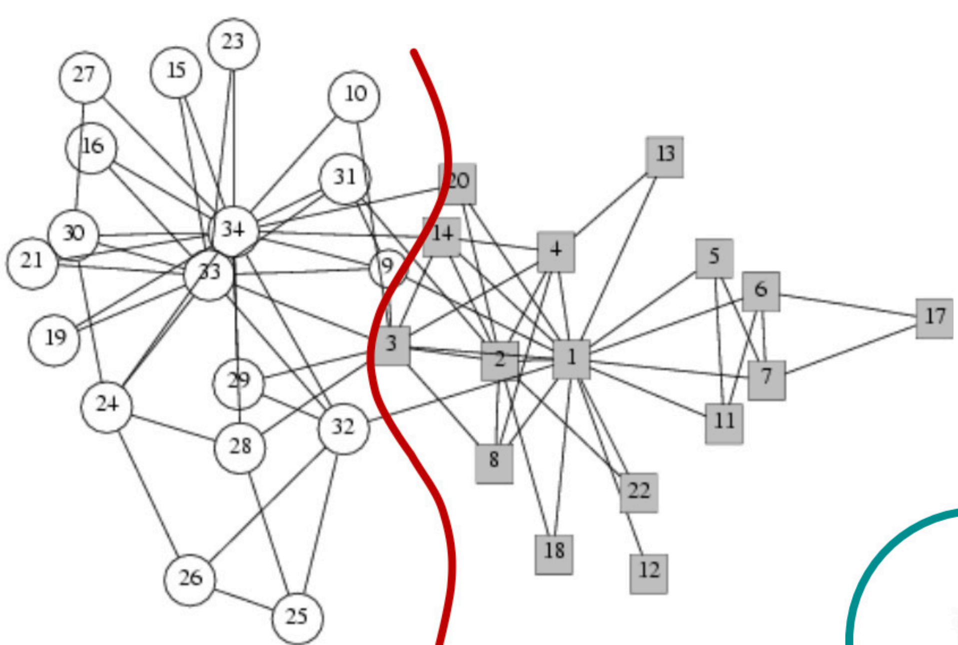
Graphical depiction of the hierarchical clustering splits done at every step

"First AB/CDEF, then C/DEF, then D/EF, then A/B, then E/F"

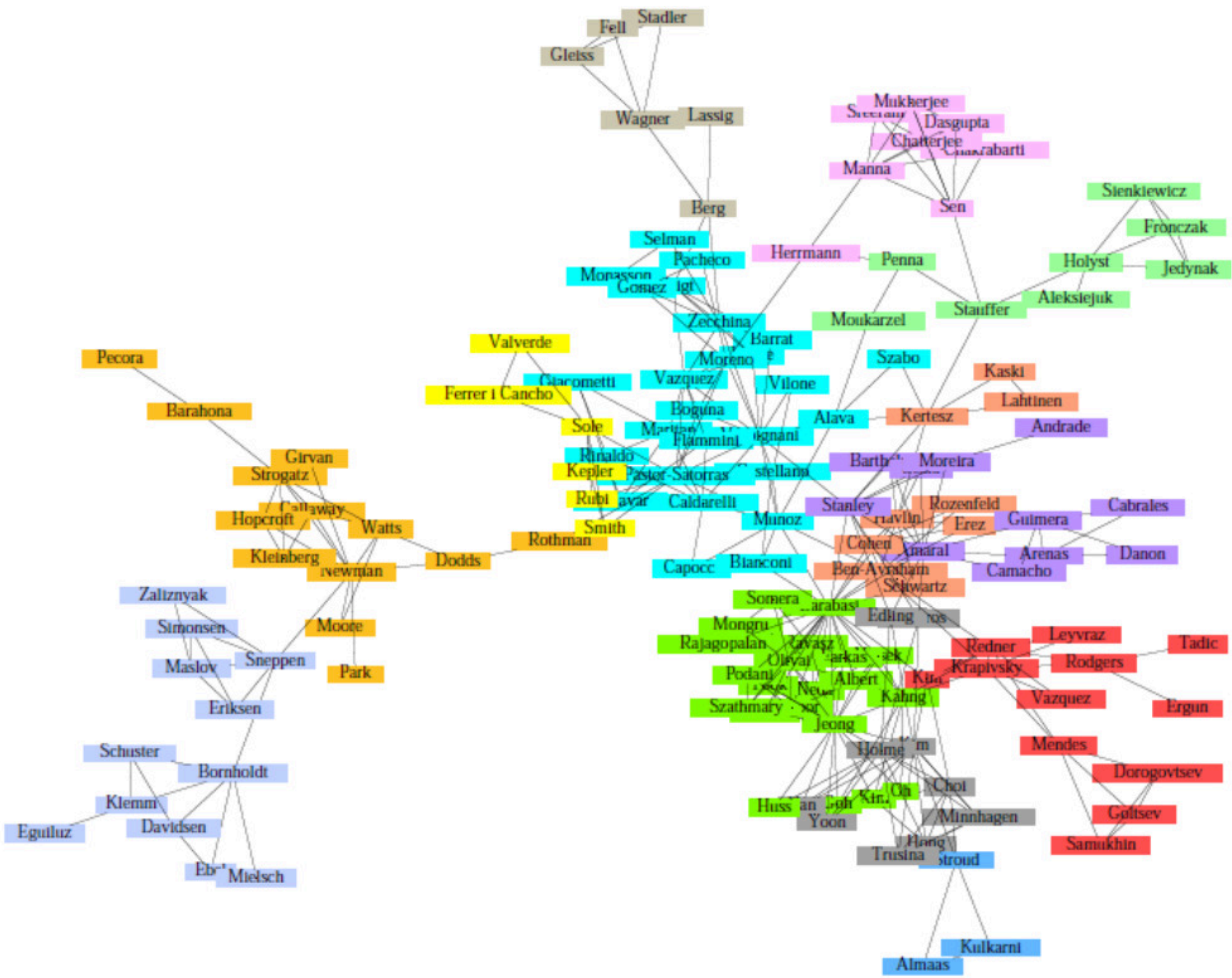
Zachary Karate Club

Dendrogram

Graphical depiction of the hierarchical clustering splits done at every step



Girvan-Newman: Results

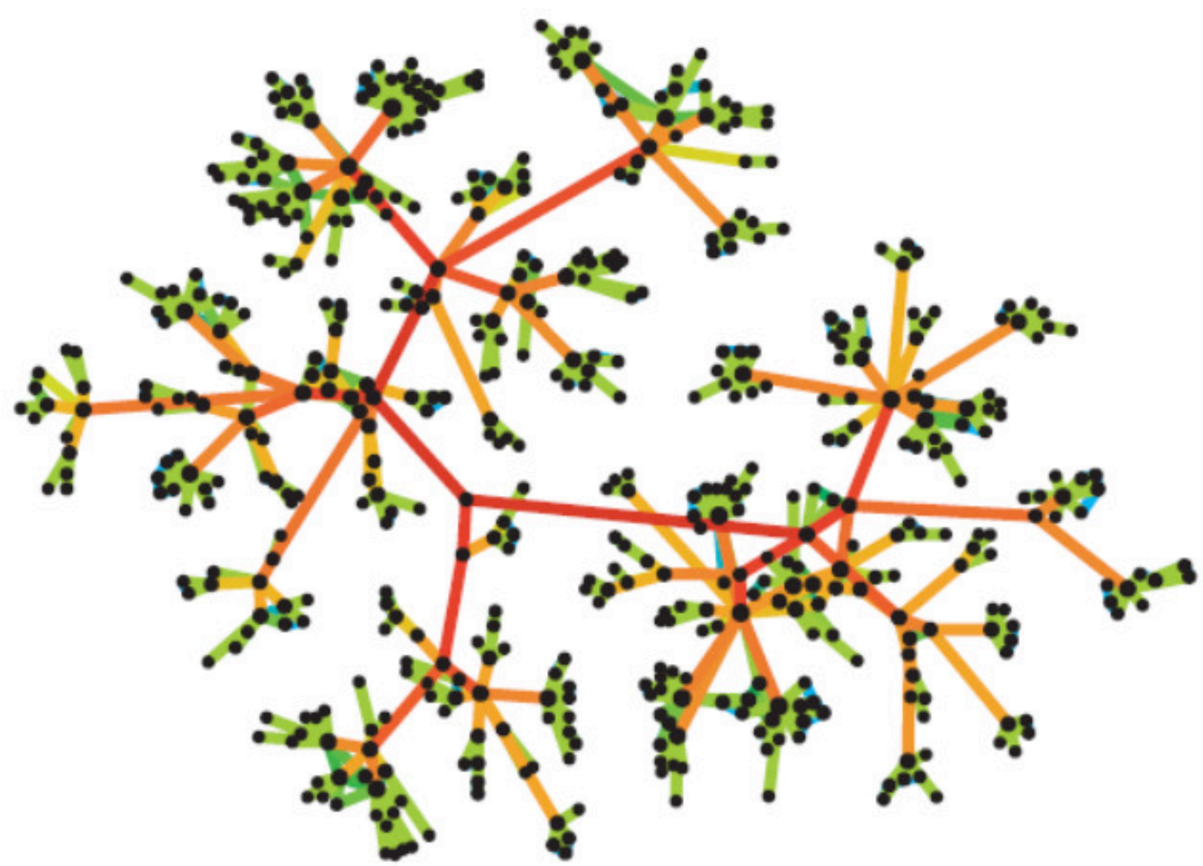


Communities in physics collaborations

We need to resolve a question

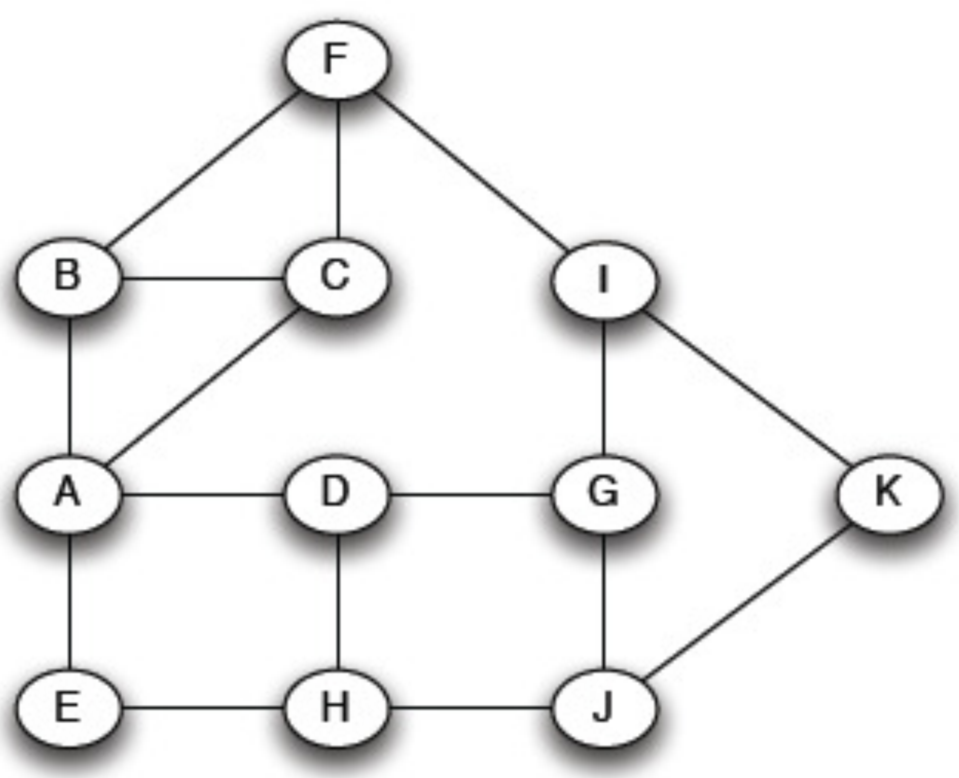
How to compute betweenness?

Counting all pairs of shortest paths for every edge is **computationally challenging**!

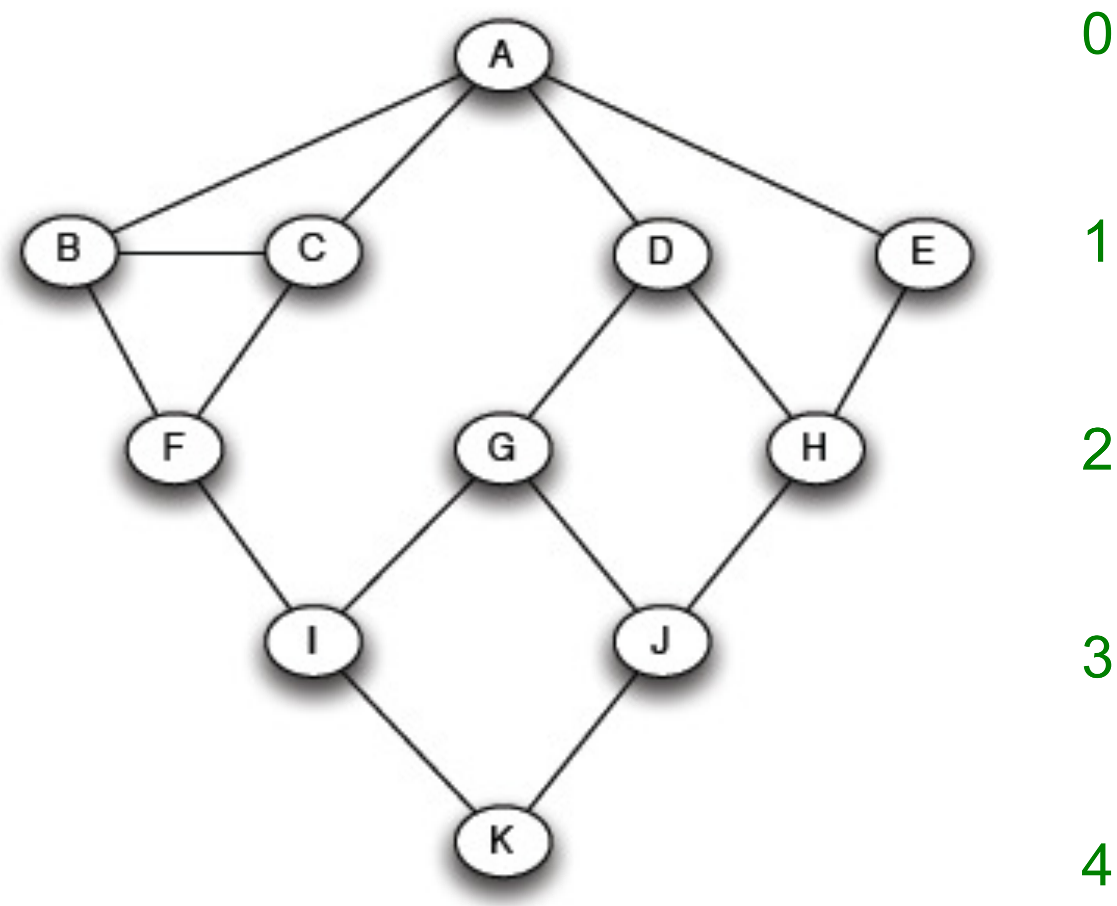


How to Compute Betweenness?

Want to compute betweenness of paths starting at node A



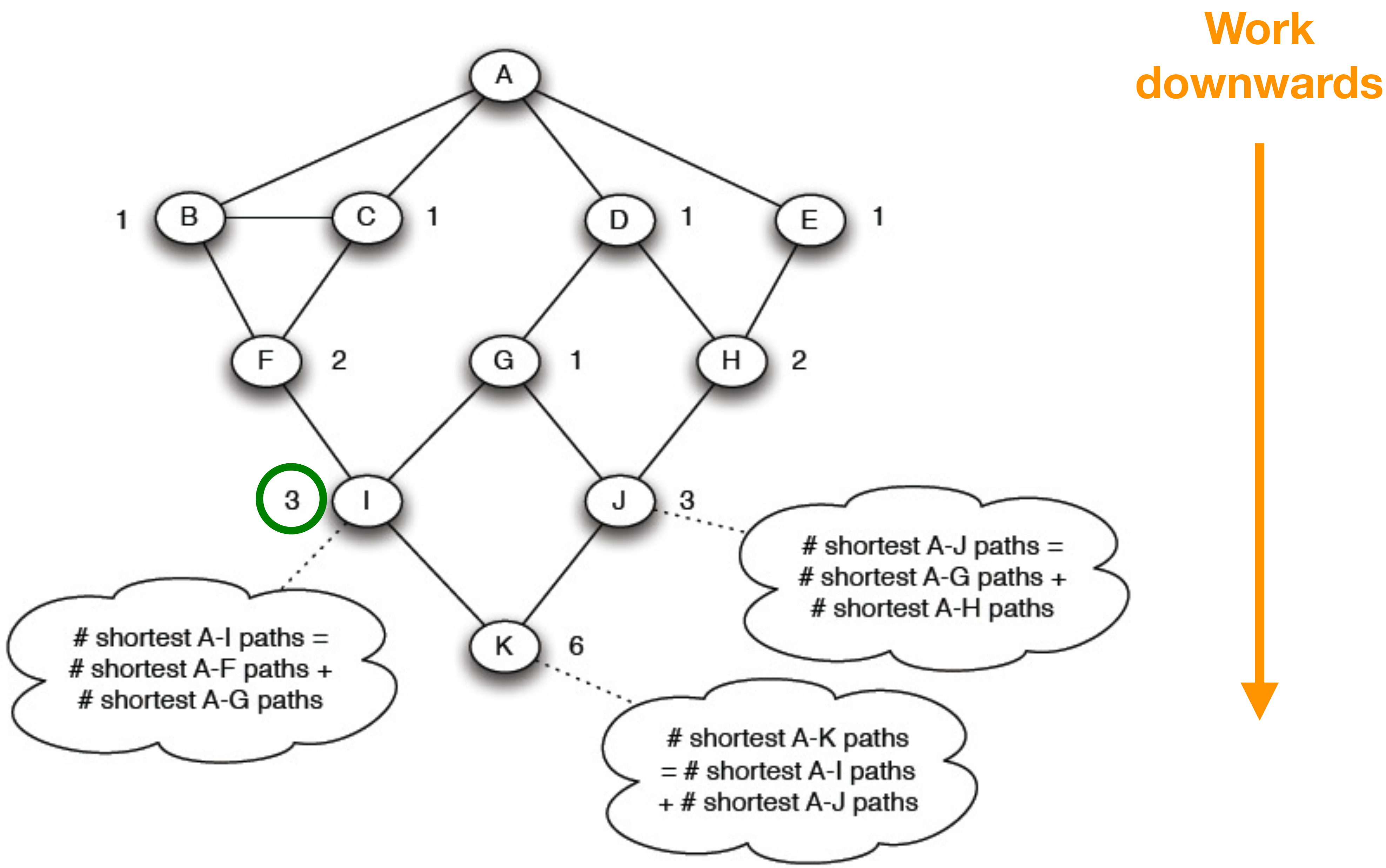
BFS starting from A:



Recall BFS goes layer-by-layer

How to Compute Betweenness?

Count the number of shortest paths from A to all other nodes in the graph:



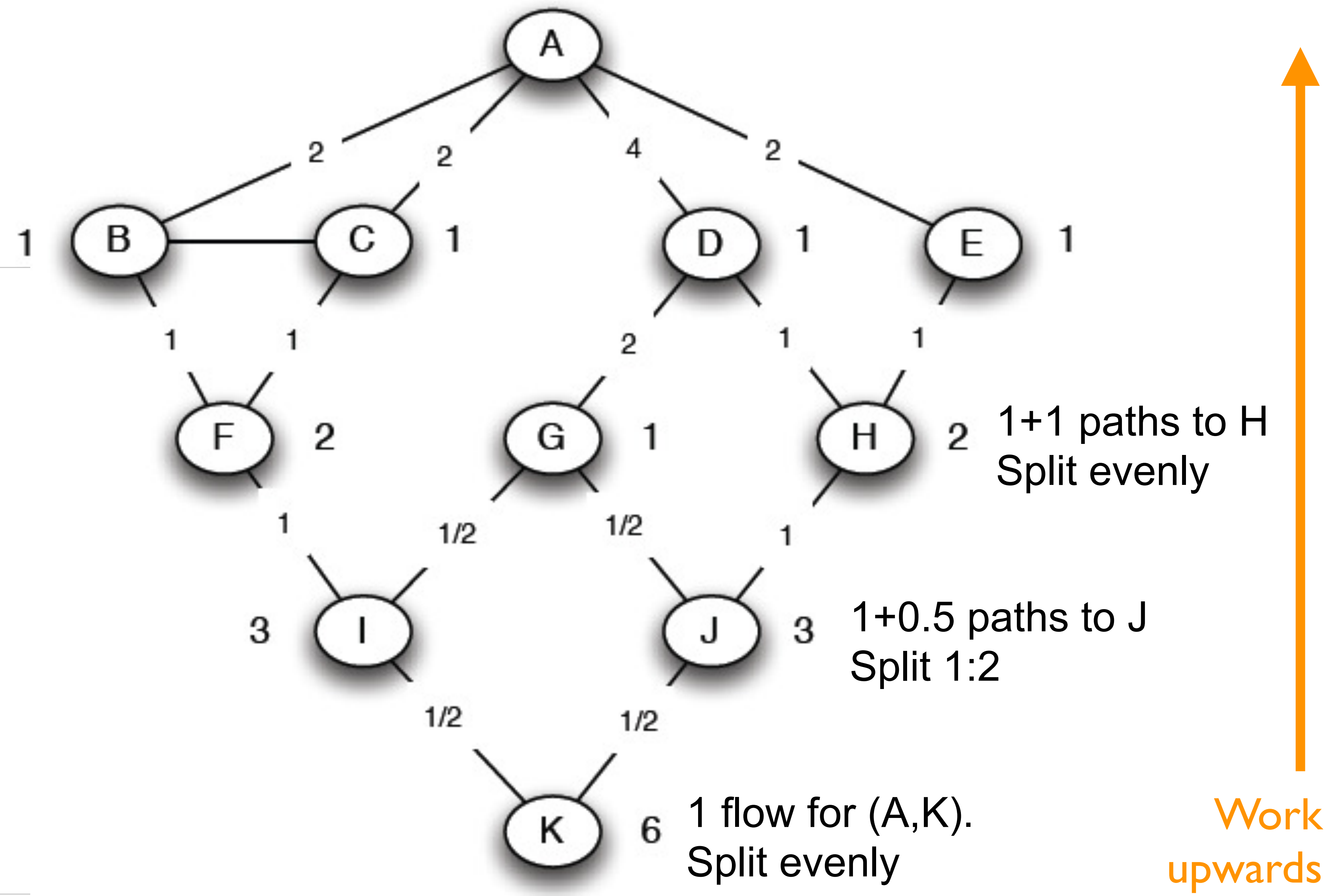
How to Compute Betweenness?

How much flow goes from A to other nodes?

Compute betweenness by working up the tree: If there are multiple paths count them fractionally

The algorithm:

- Add edge **flows**:
  - -- node flow = 1+∑child edges
  - -- split the flow up based on the parent value
- Repeat the BFS procedure for each starting node U



Girvan-Newman

–Repeat for each node in the graph, add up the edge scores that edges receive in these computations

–For each edge (u,v), must divide by 2 because we counted it once for u and once for v

–Works on moderately-sized graphs

–To scale to big data, still expensive, and requires approximations or related more efficient methods

Granovetter's Explanation

- Granovetter makes a connection between social and structural role of an edge
- **First point: Structure**
  - Structurally embedded edges are also socially strong
  - Long-range edges spanning different parts of the network are socially weak
- **Second point: Information**
  - Long-range edges allow you to gather information from different parts of the network and get a job
  - Structurally embedded edges are heavily redundant in terms of information access