BBS654
Data Mining

Pinar Duygulu

Slides are adapted from

J. Leskovec, A. Rajaraman, J. Ullman: Mining of Massive Datasets, http://www.mmds.org

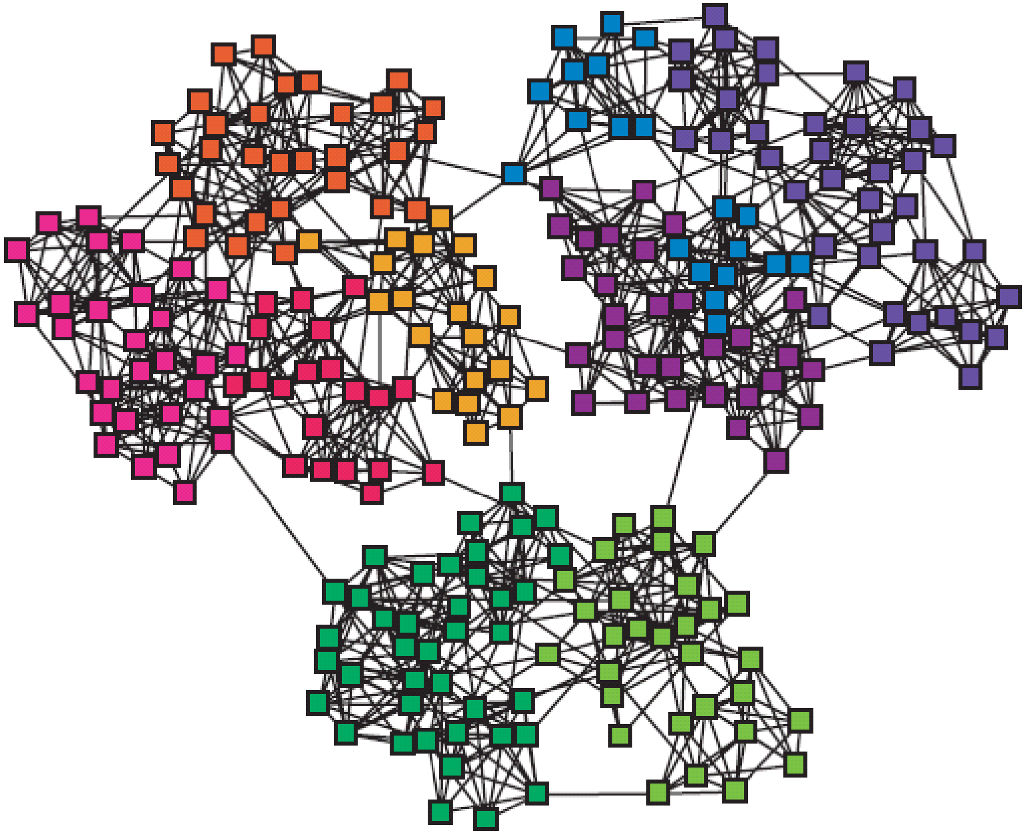
Analysis of Large Graphs: Community Detection

Mining of Massive Datasets
Jure Leskovec, Anand Rajaraman, Jeff Ullman
Stanford University

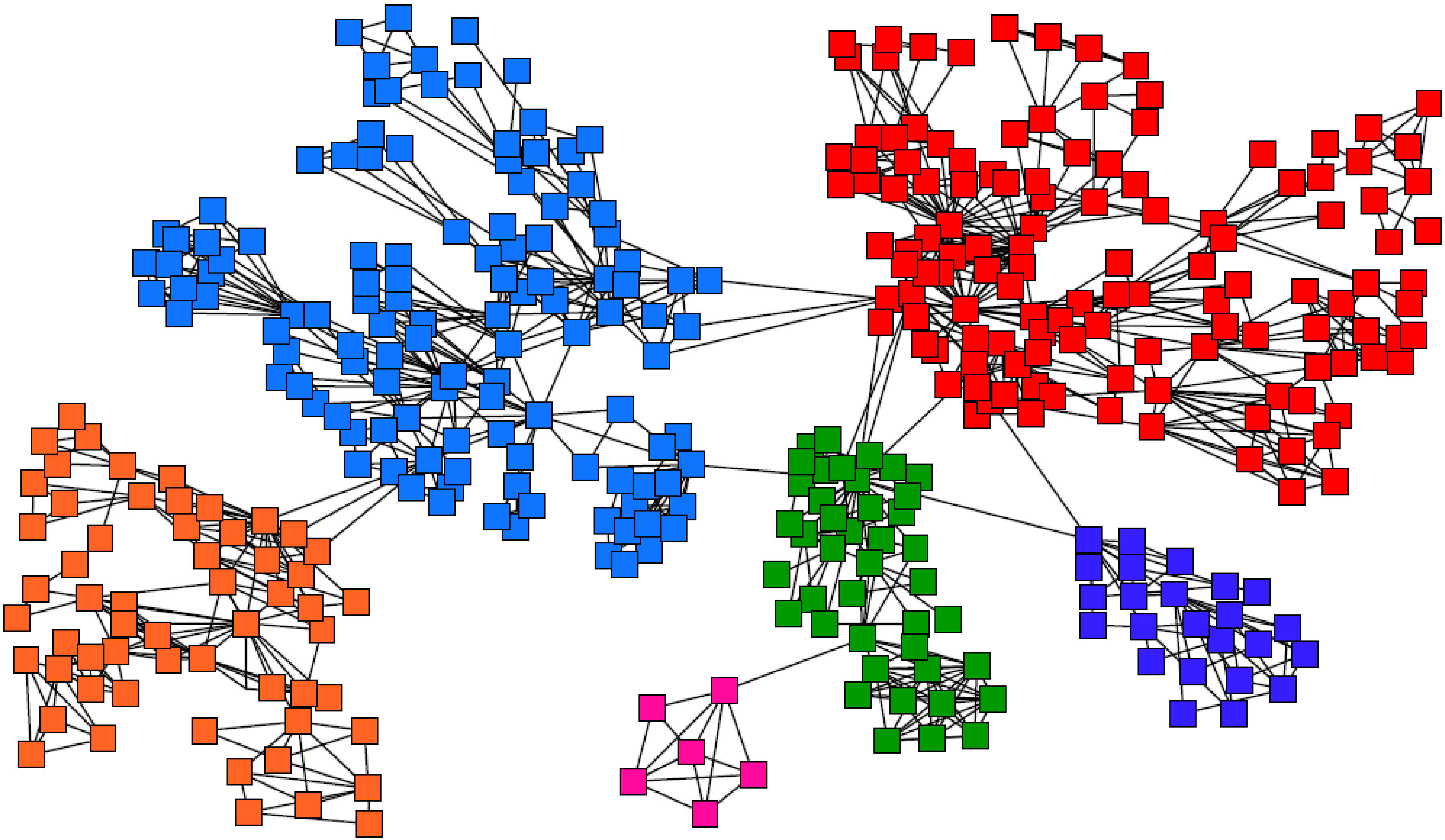
http://www.mmds.org

Networks & Communities

- We often think of networks being organized into modules, cluster, communities:

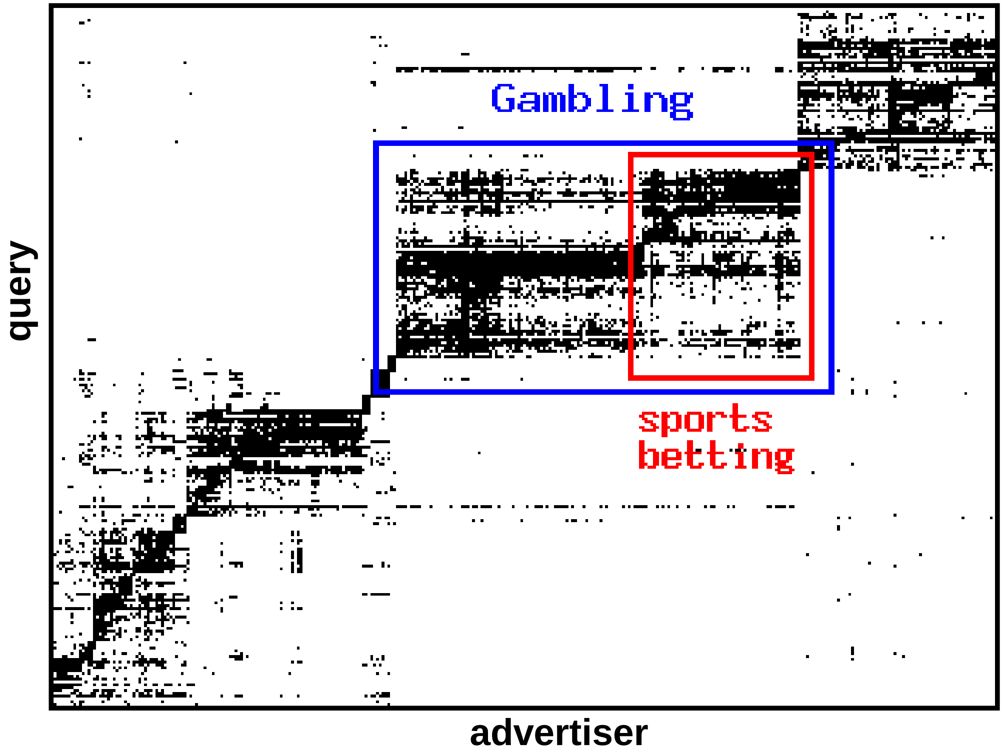


Goal: Find Densely Linked Clusters



Micro-Markets in Sponsored Search

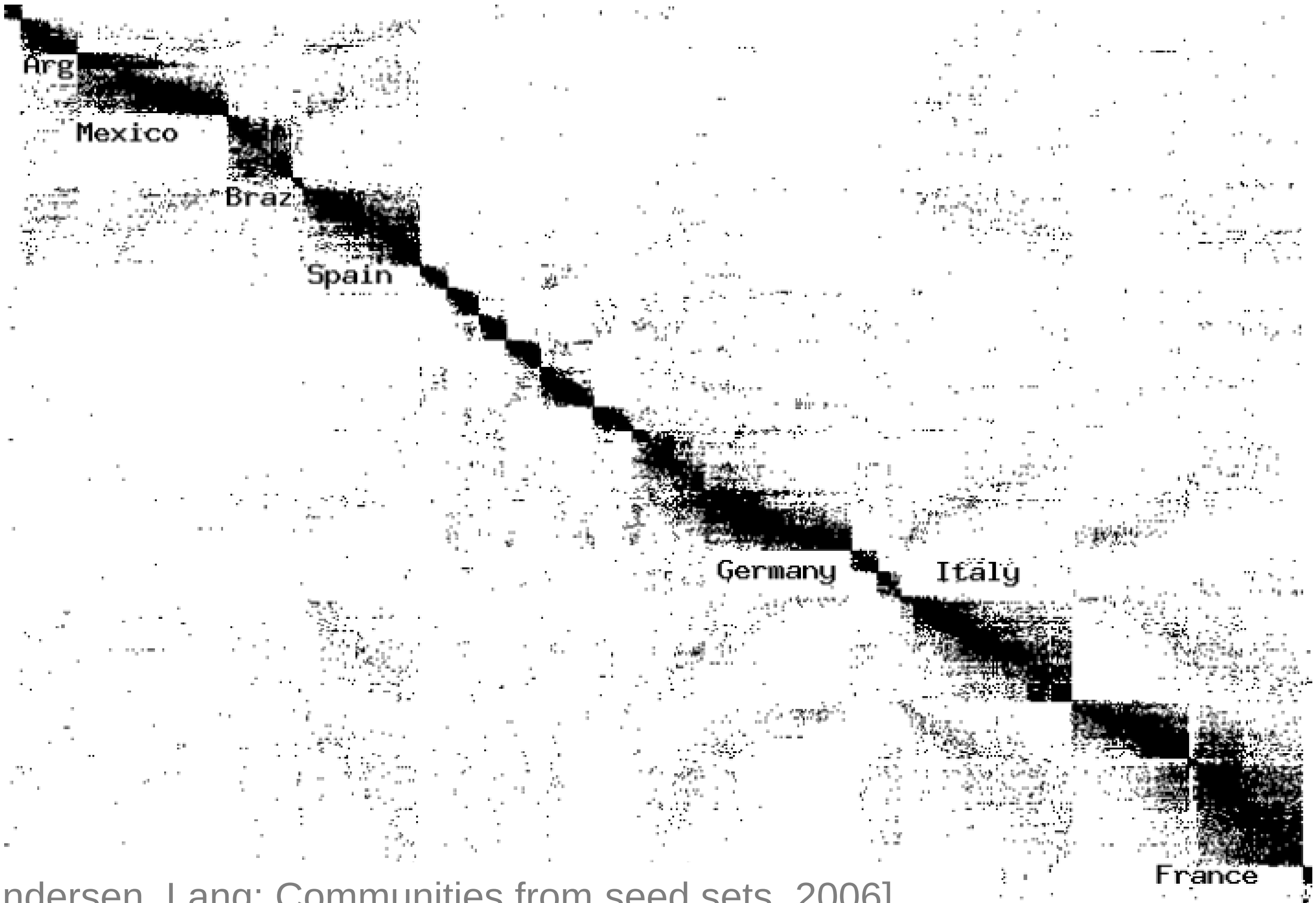
- Find micro-markets by partitioning the query-to-advertiser graph:



[Andersen, Lang: Communities from seed sets, 2006]

Movies and Actors

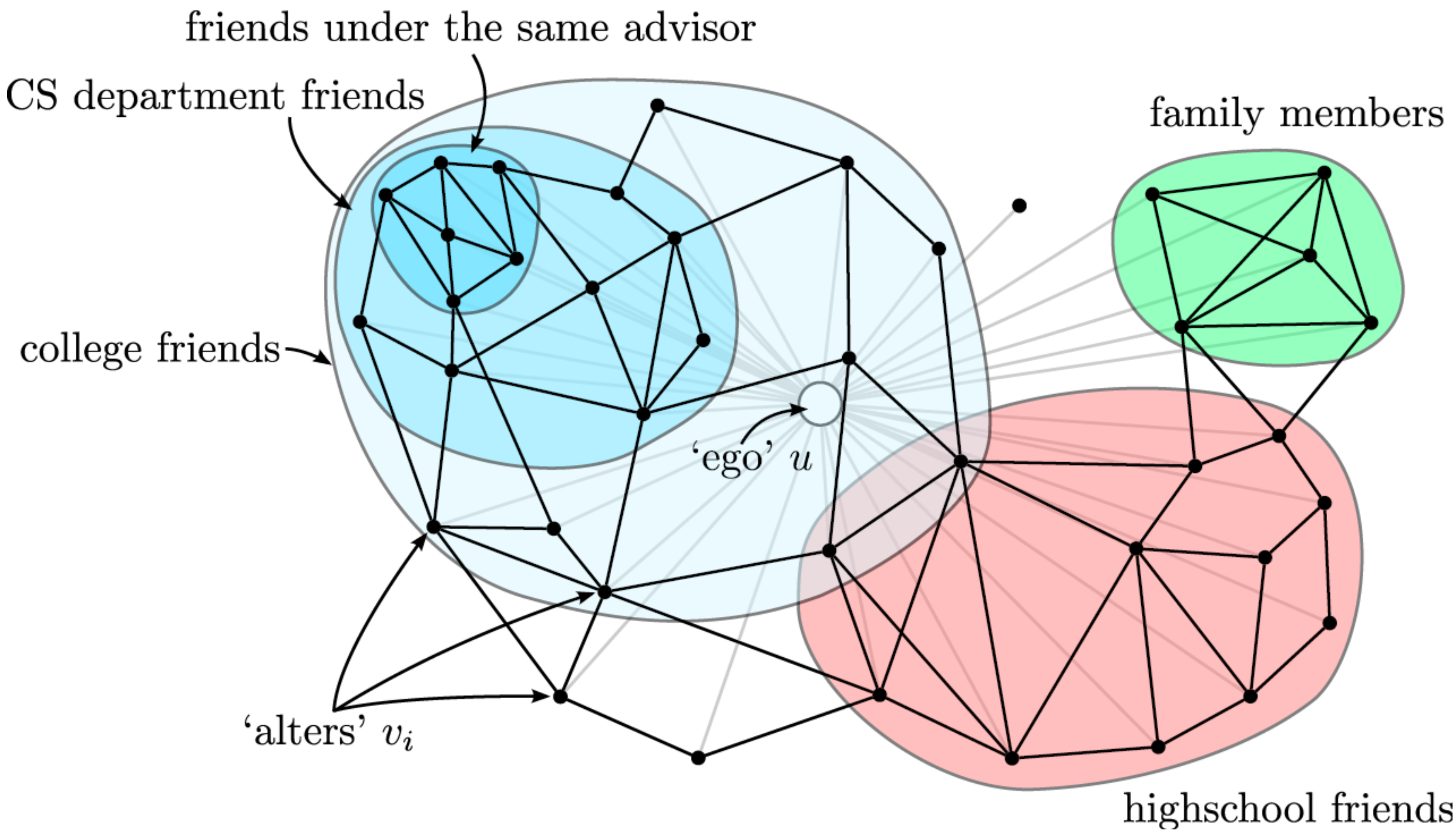
- **Clusters in Movies-to-Actors graph:**



[Andersen, Lang: Communities from seed sets, 2006]

Twitter & Facebook

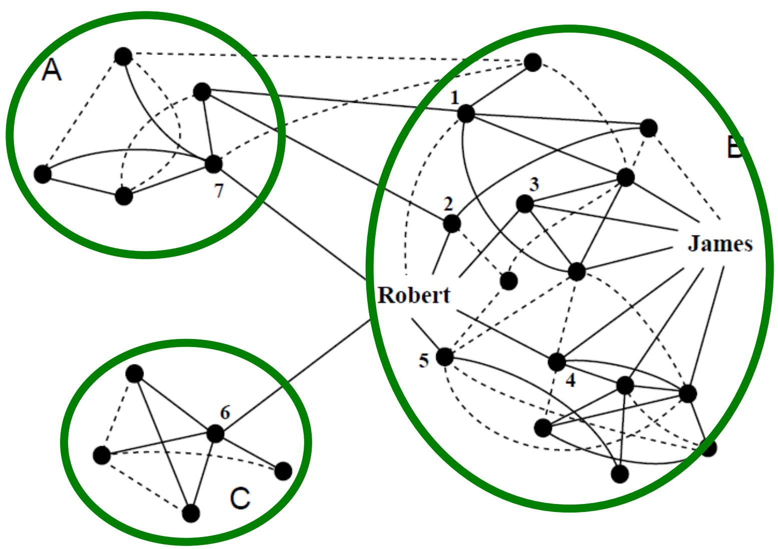
- **Discovering social circles, circles of trust:**

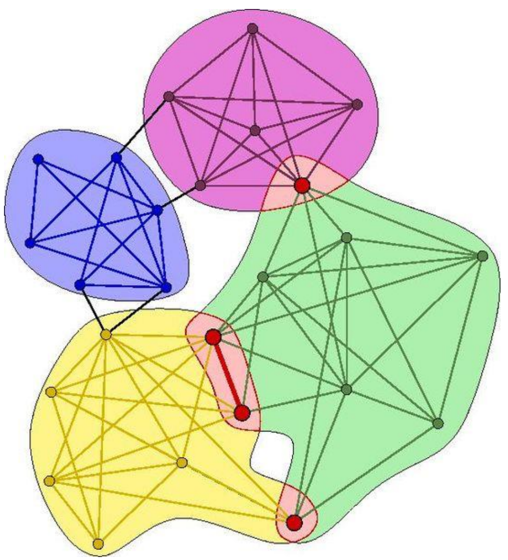


[McAuley, Leskovec: Discovering social circles in ego networks, 2012]

COMMUNITY DETECTION

How to find communities?

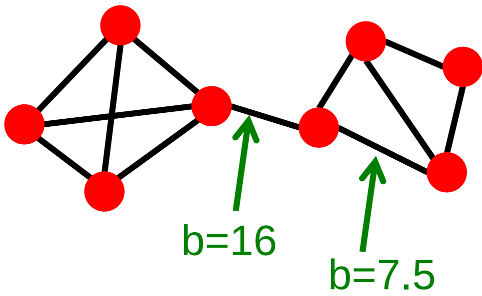


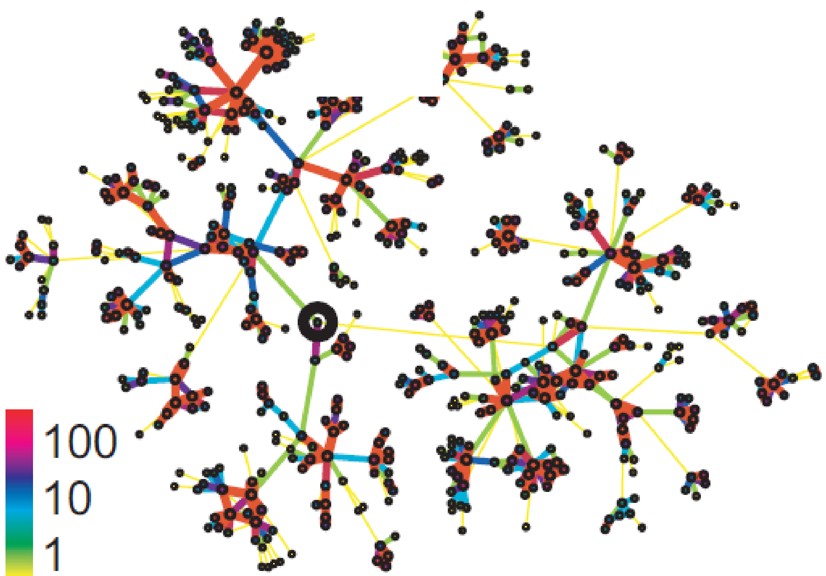


We will work with **undirected** (unweighted) networks

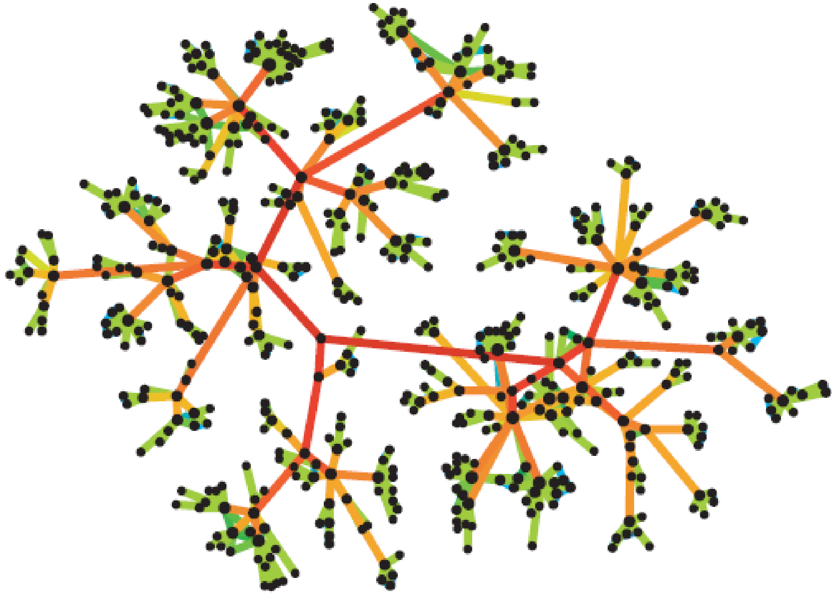
Method 1: Strength of Weak Ties

- **Edge betweenness:** Number of shortest paths passing over the edge
- **Intuition:**





Edge strengths (call volume) in a real network



Edge betweenness in a real network

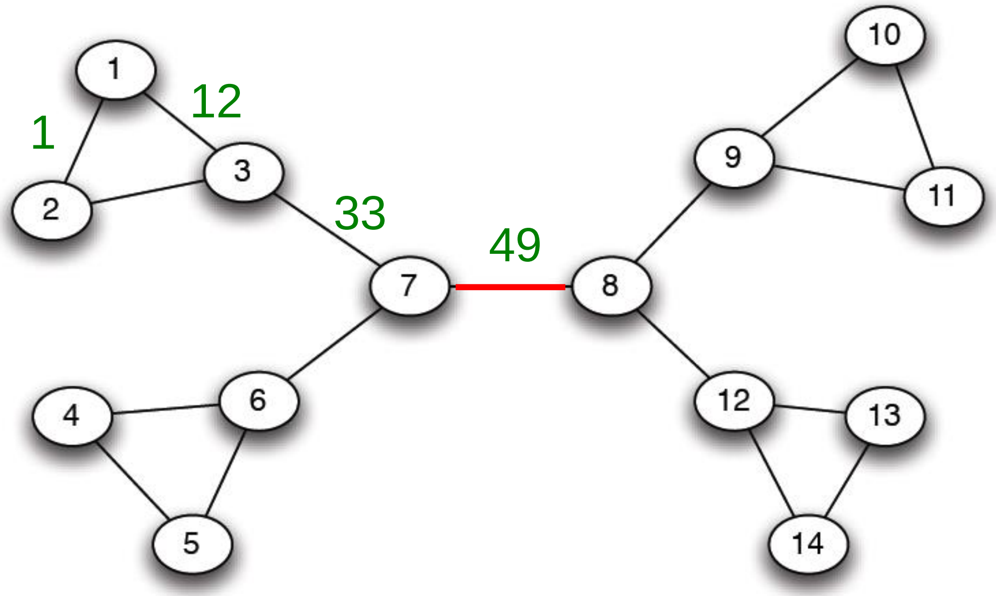
Method 1: Girvan-Newman

- Divisive hierarchical clustering based on the notion of edge **betweenness:**

  Number of shortest paths passing through the edge

- **Girvan-Newman Algorithm:**
  - » **Undirected unweighted networks**
  - **Repeat until no edges are left:**
    - Calculate betweenness of edges
    - Remove edges with highest betweenness
  - Connected components are communities
  - Gives a hierarchical decomposition of the network

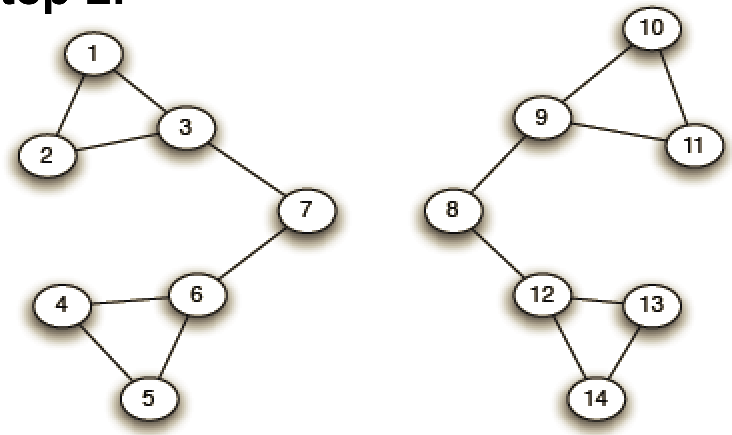
Girvan-Newman: Example



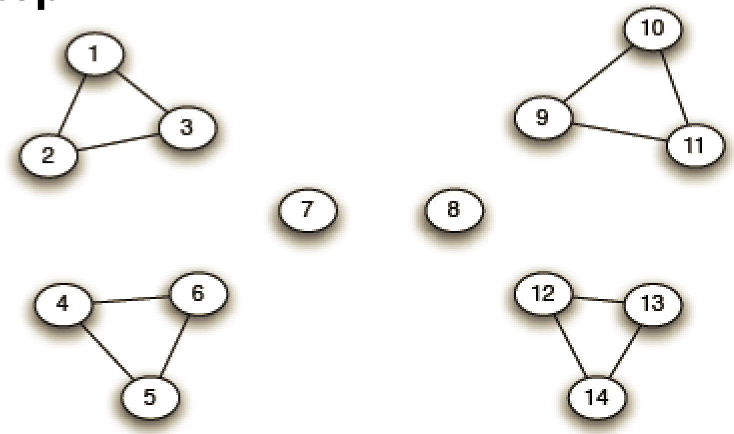
Need to re-compute betweenness at every step

Girvan-Newman: Example

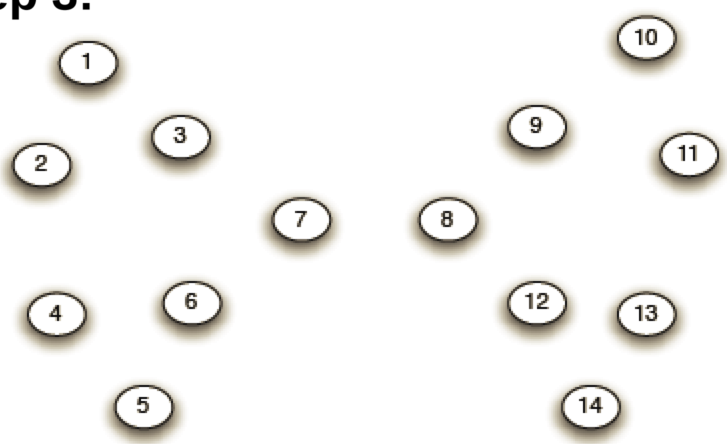
Step 1:



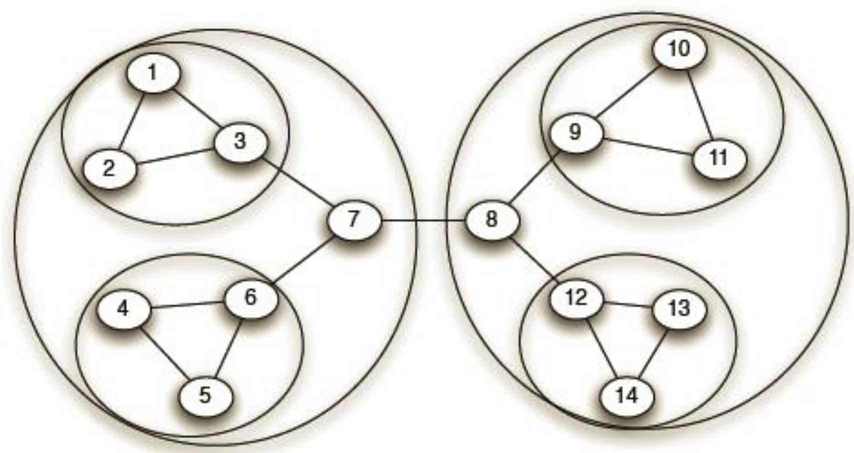
Step 2:



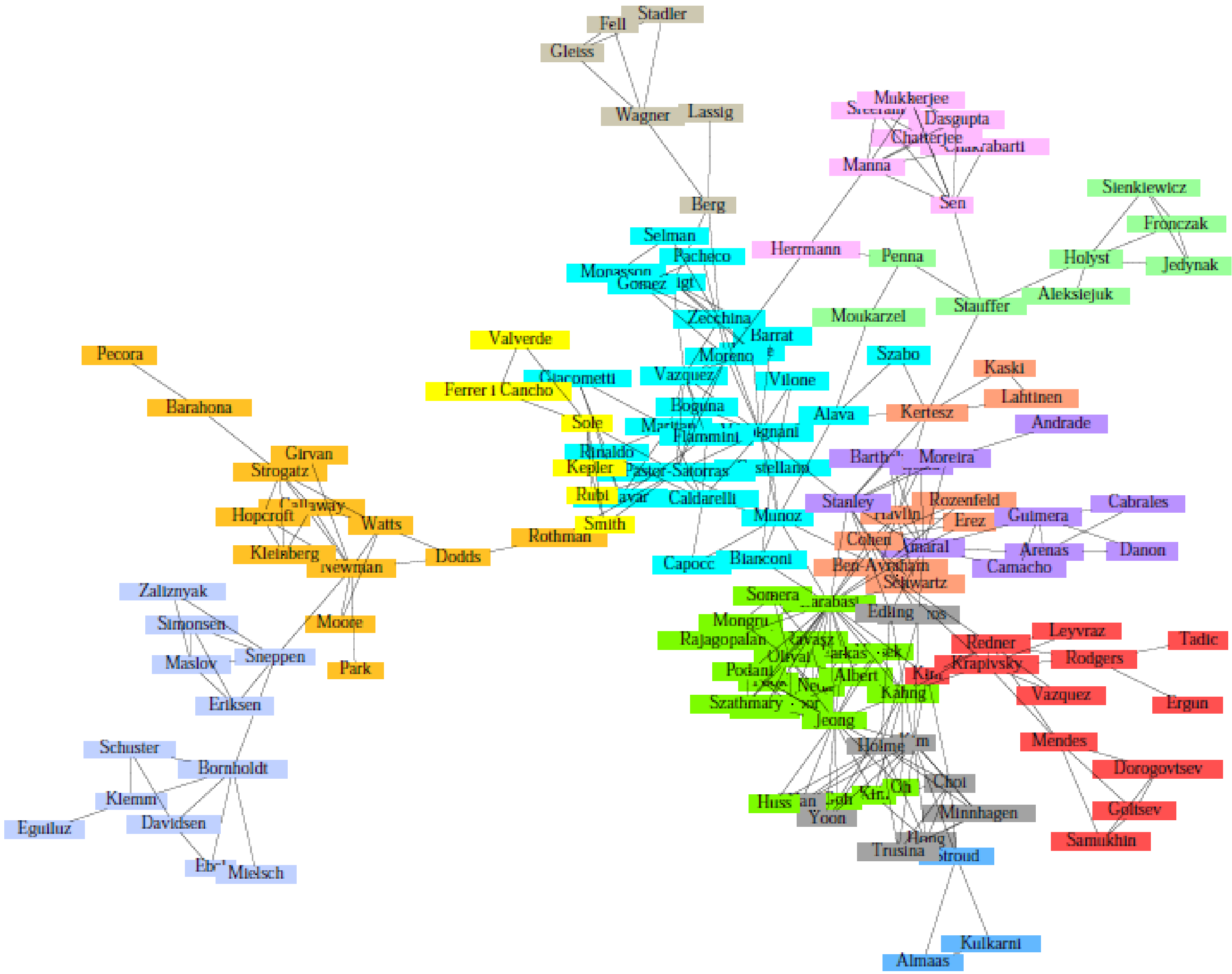
Step 3:



Hierarchical network decomposition:



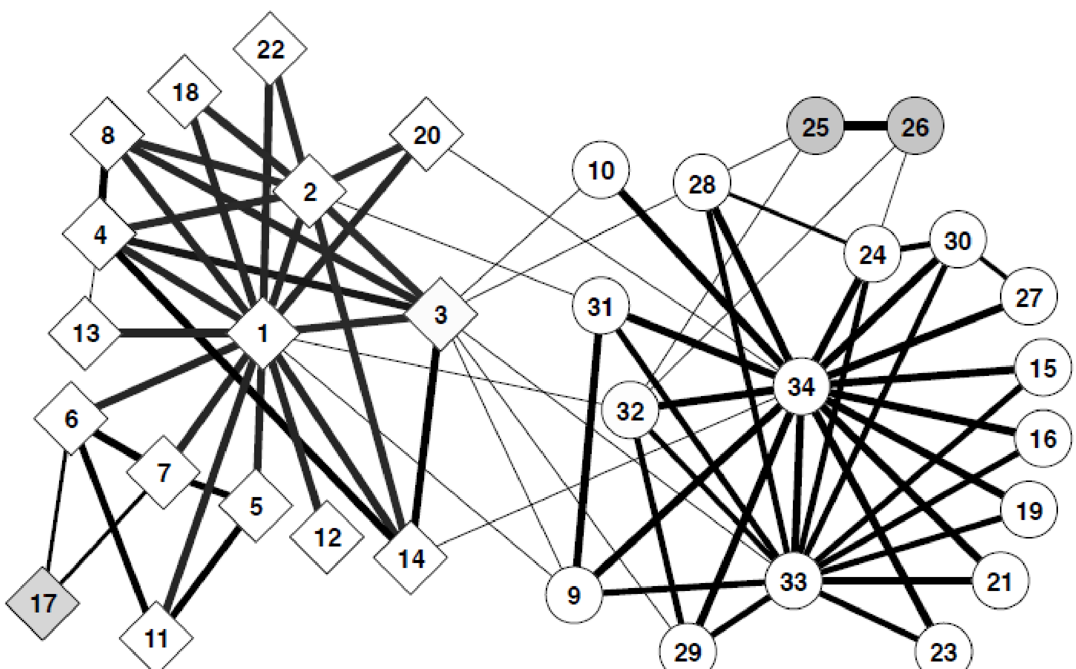
Girvan-Newman: Results

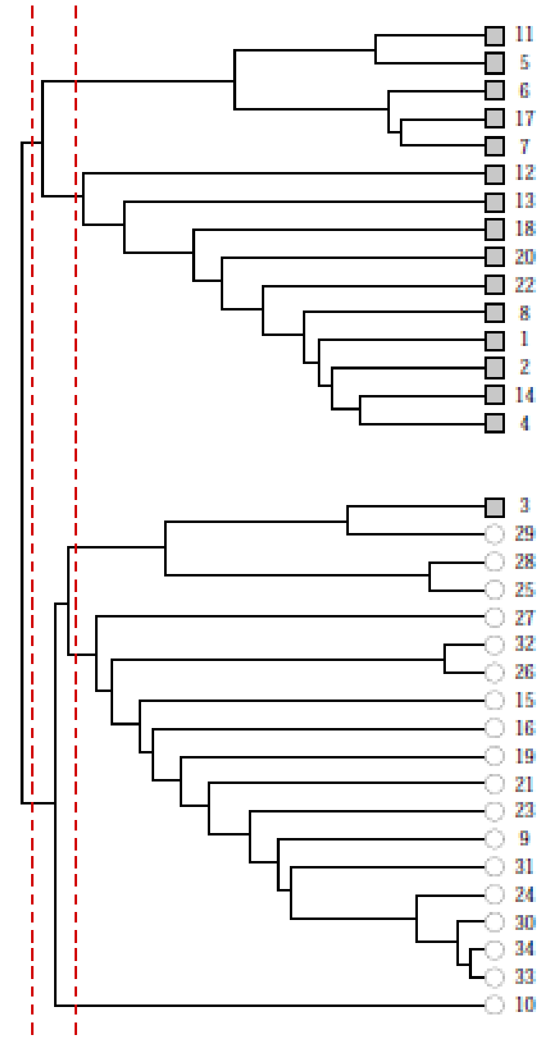


Communities in physics collaborations

Girvan-Newman: Results

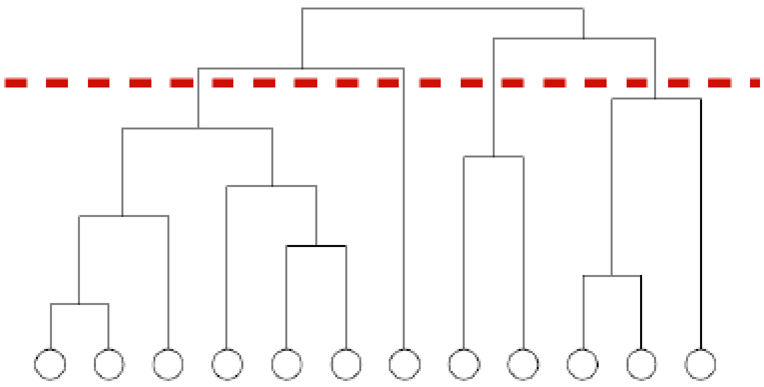
- **Zachary's Karate club:** Hierarchical decomposition





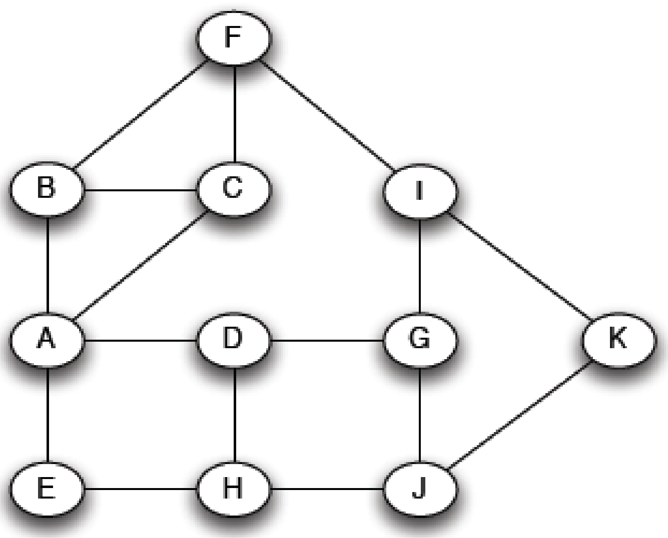
WE NEED TO RESOLVE 2 QUESTIONS

1. How to compute betweenness?
2. How to select the number of clusters?

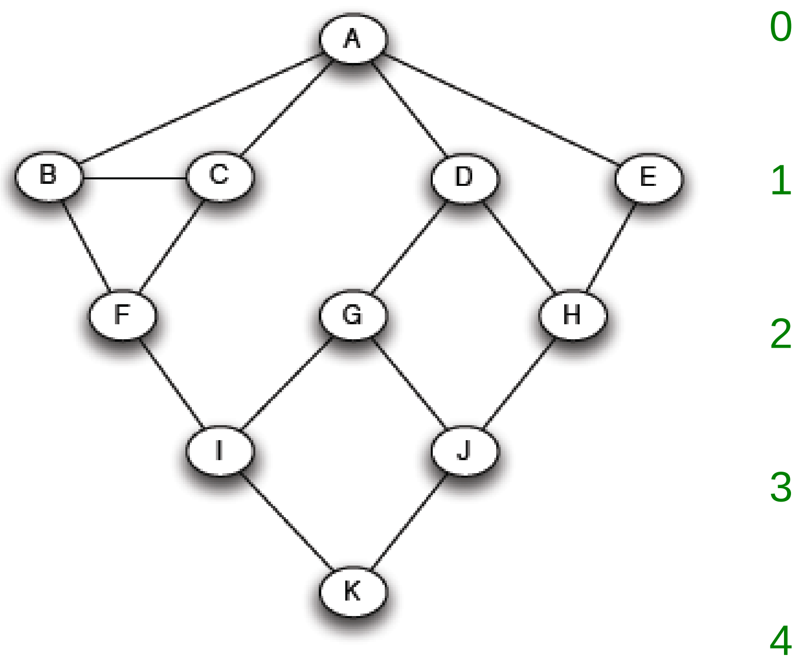


How to Compute Betweenness?

- **Want to compute betweenness of paths starting at node A**

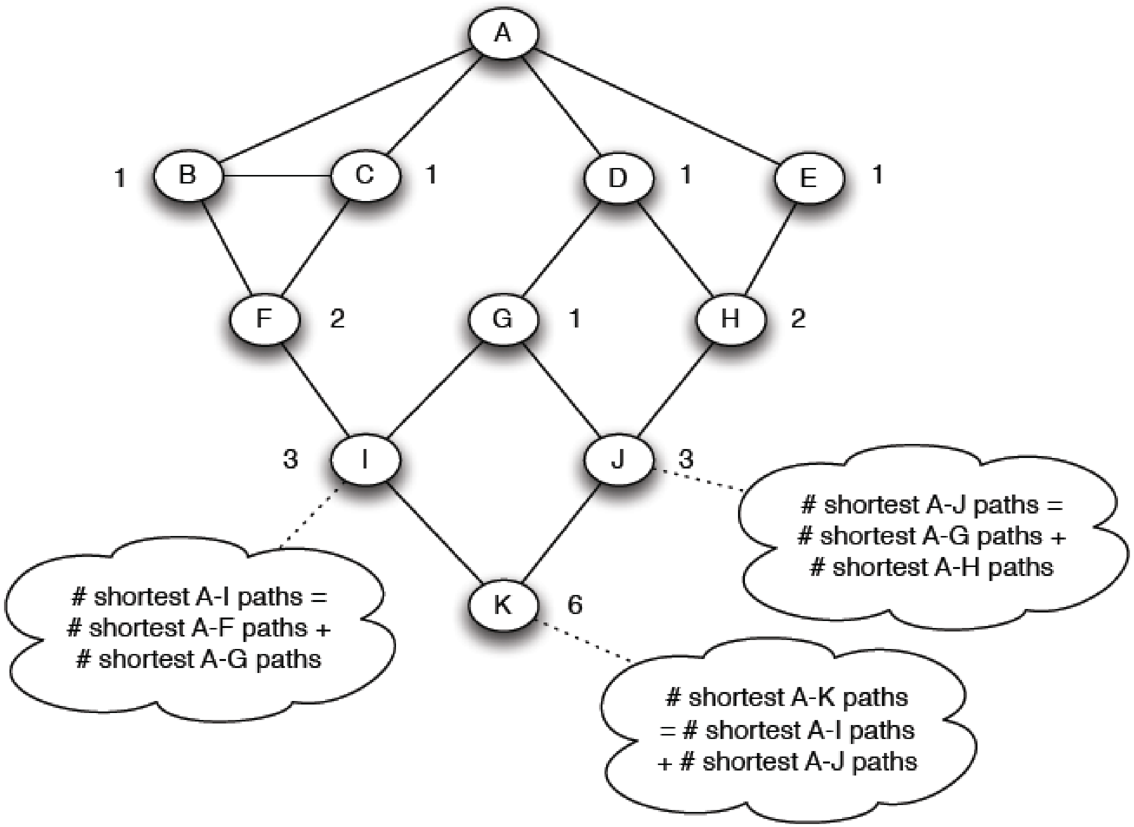


- **Breath first search starting from A:**



How to Compute Betweenness?

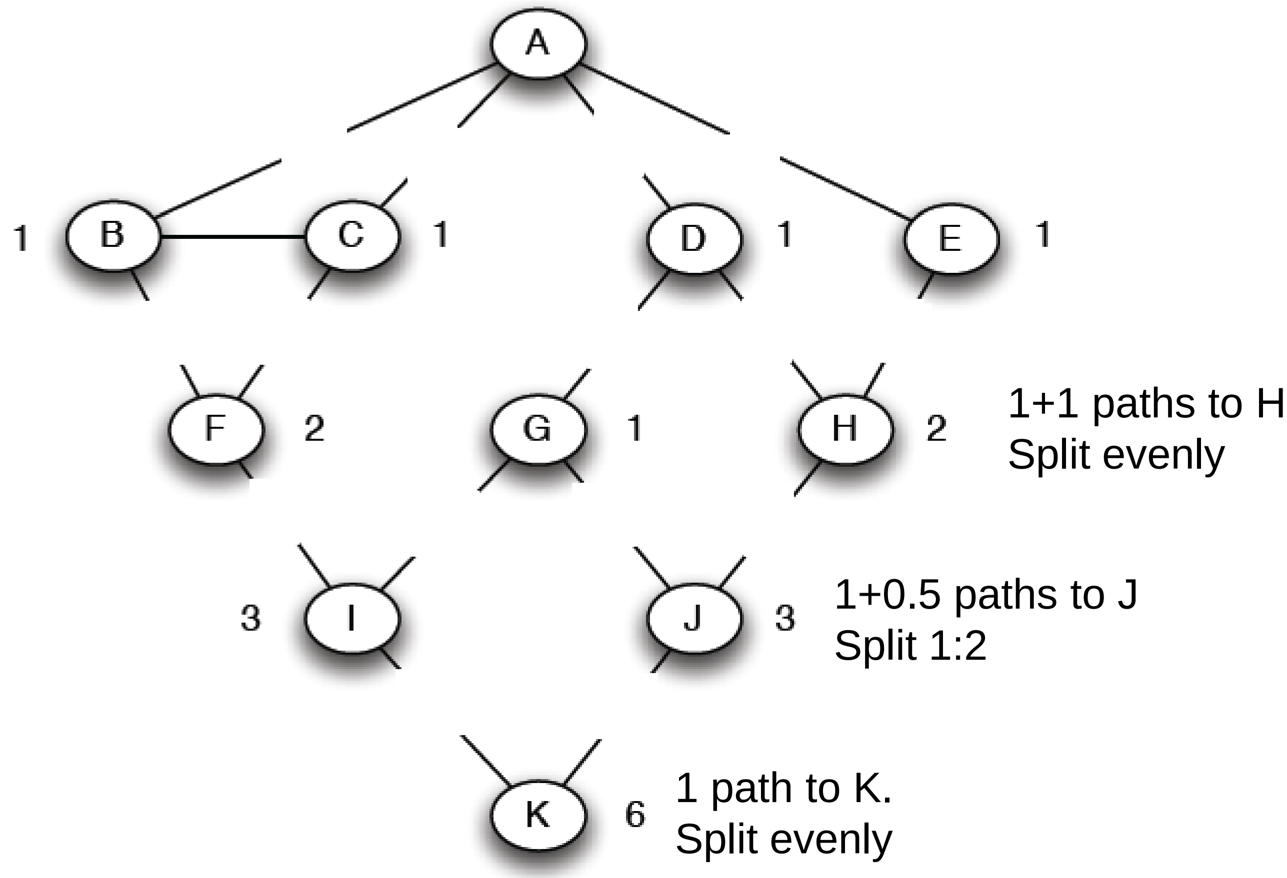
- Count the number of shortest paths from A to all other nodes of the network:



How to Compute Betweenness?

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

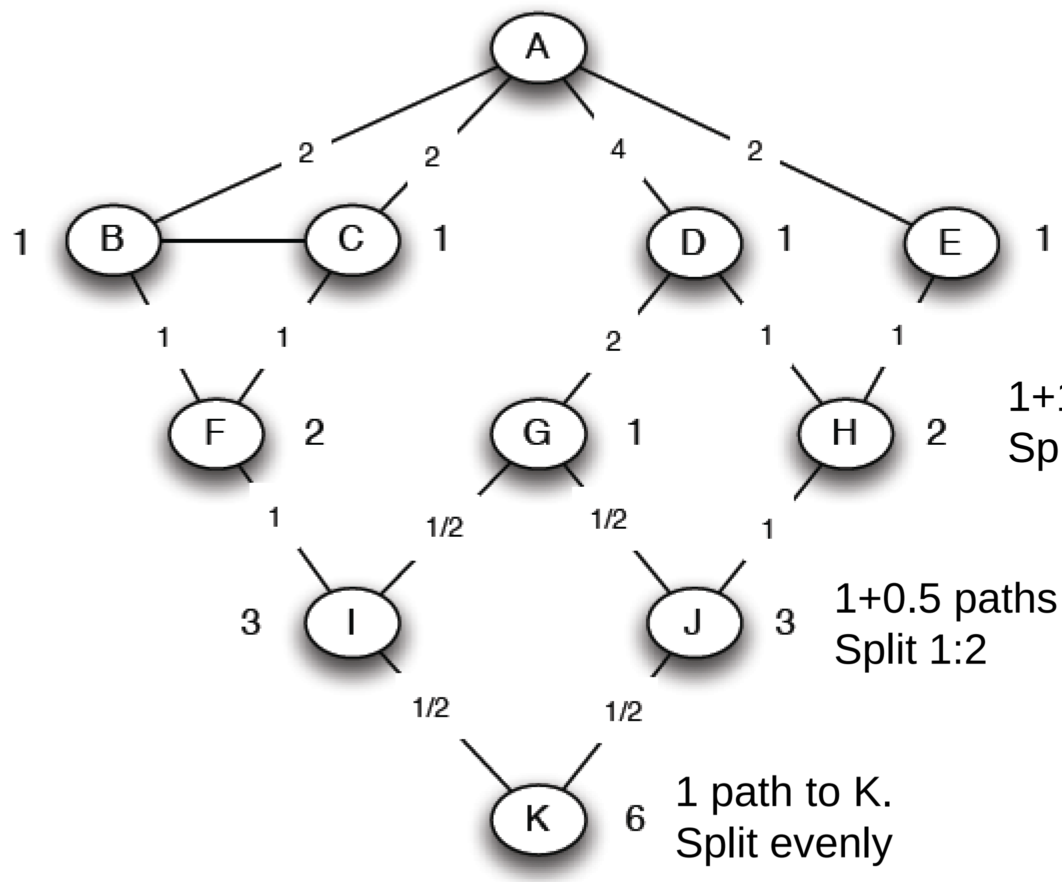
The algorithm:
•Add edge **flows**:
-- node flow = 1+∑child edges
-- split the flow up based on the parent value
• Repeat the BFS procedure for each starting node *U*



How to Compute Betweenness?

- **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*



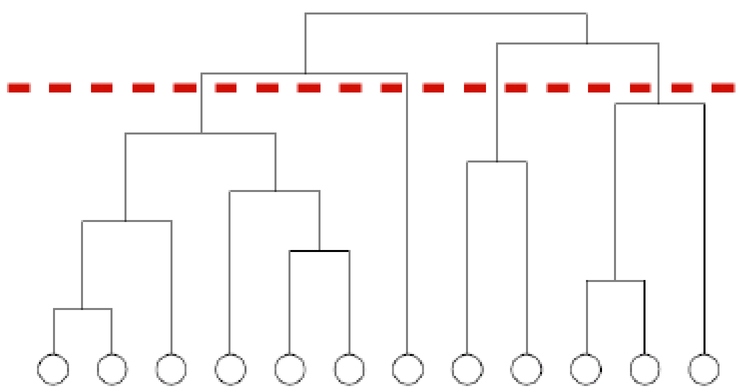
1+1 paths to H
Split evenly

1+0.5 paths to J
Split 1:2

1 path to K.
Split evenly

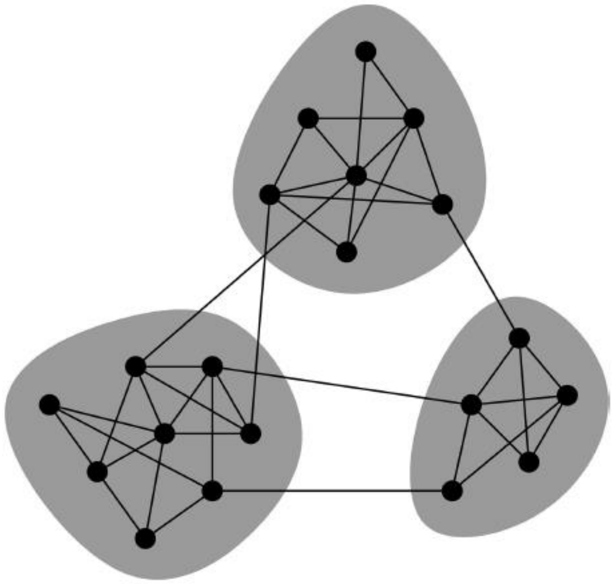
WE NEED TO RESOLVE 2 QUESTIONS

1. How to compute betweenness?
2. How to select the number of clusters?



Network Communities

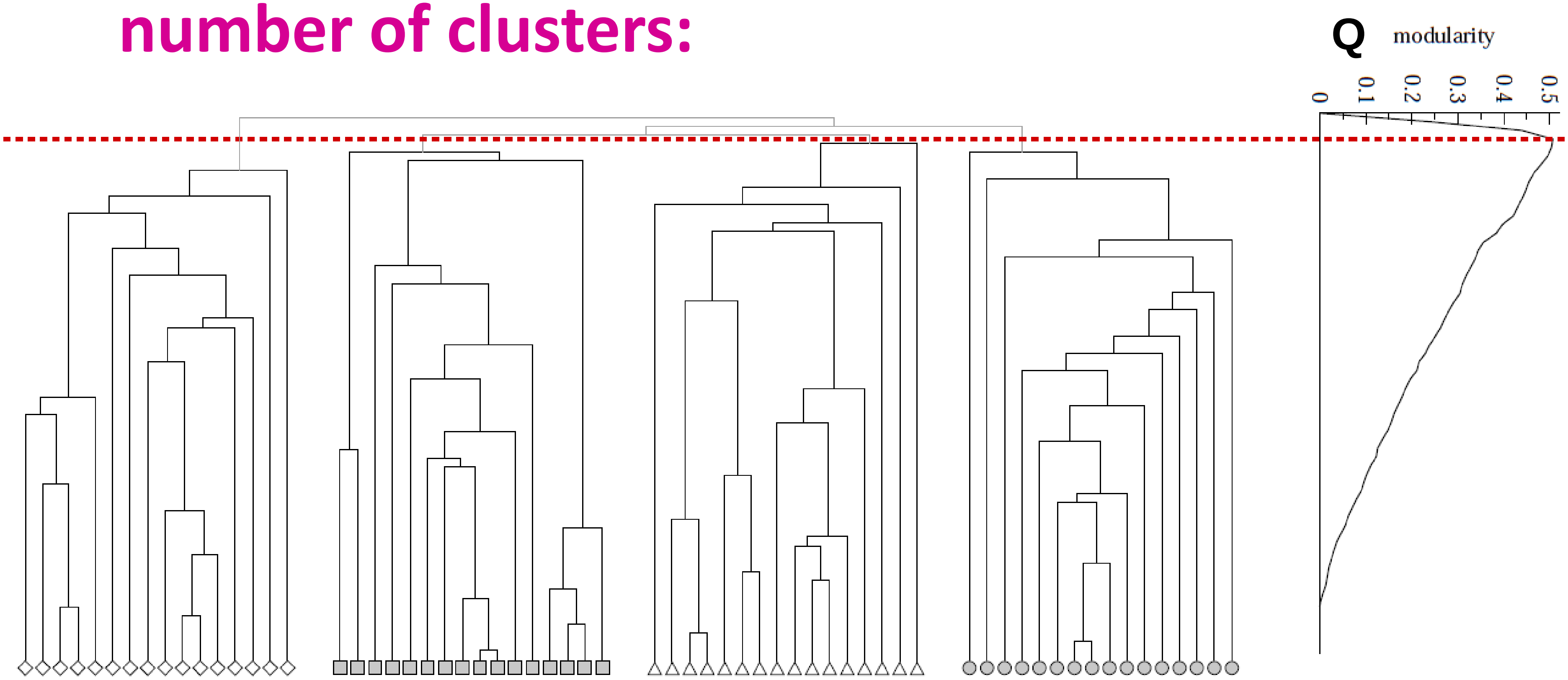
- **Communities:** sets of **tightly connected nodes**
- Define: **Modularity Q**
  - A measure of how well a network is partitioned into communities
  - Given a partitioning of the network into groups $s \in S$:



$$Q \propto \sum_{s \in S} [\ (\#\ \text{edges within group } s) - (\text{expected}\ \#\ \text{edges within group } s)\ ]$$

Modularity: Number of clusters

- Modularity is useful for selecting the number of clusters:

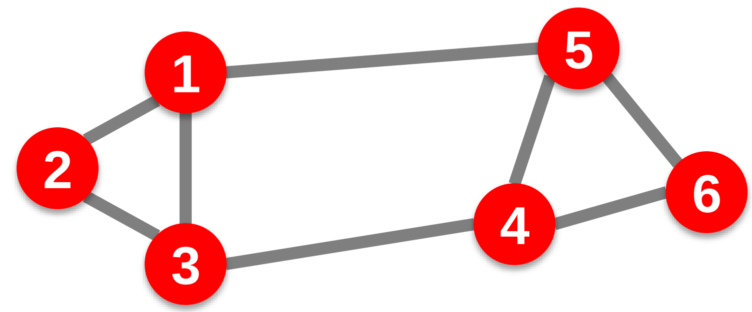


Next time: Why not optimize Modularity directly?

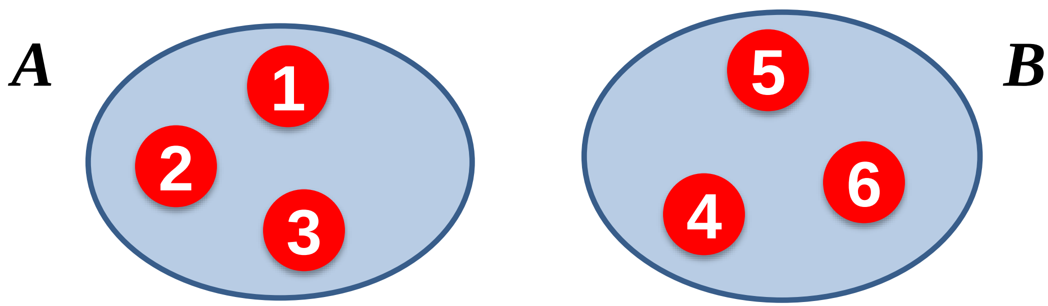
Spectral Clustering

Graph Partitioning

- **Undirected graph $G(V, E)$:**



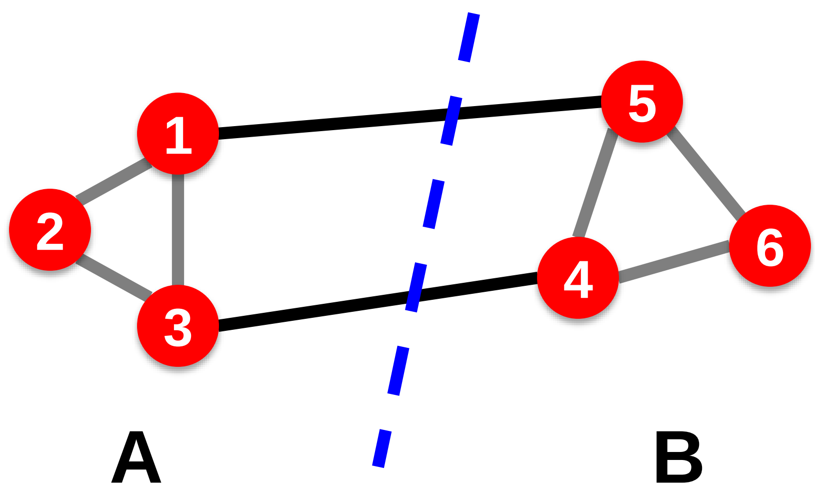
- **Bi-partitioning task:**
  - Divide vertices into two disjoint groups $\boldsymbol{A}, \boldsymbol{B}$



- **Questions:**
  - How can we define a "good" partition of $\boldsymbol{G}$?
  - How can we efficiently identify such a partition?

Graph Partitioning

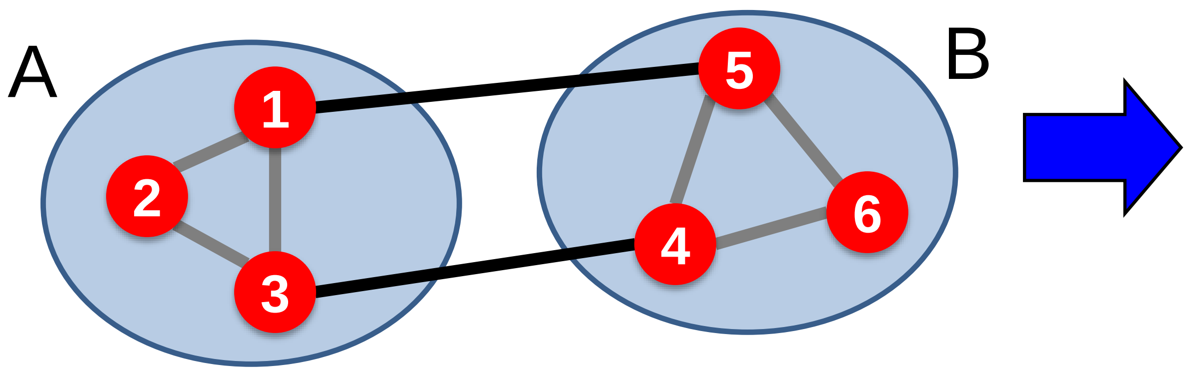
- **What makes a good partition?**
  - Maximize the number of within-group connections
  - Minimize the number of between-group connections



Graph Cuts

- **Express partitioning objectives as a function of the "edge cut" of the partition**
- **Cut:** Set of edges with only one vertex in a group:

$$cut(A,B) = \sum_{i \in A, j \in B} w_{ij}$$



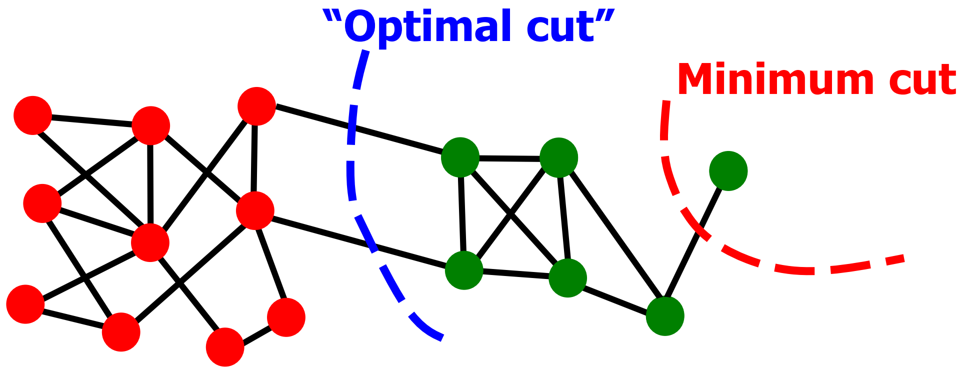
$cut(A,B) = 2$

Graph Cut Criterion

- **Criterion: Minimum-cut**
  - Minimize weight of connections between groups

$$\arg\min_{A,B} cut(A,B)$$

- **Degenerate case:**



- **Problem:**
  - Only considers external cluster connections
  - Does not consider internal cluster connectivity

Graph Cut Criteria

- **Criterion: Normalized-cut** [Shi-Malik, ’97]
  - Connectivity between groups relative to the density of each group

$$ncut(A,B) = \frac{cut(A,B)}{vol(A)} + \frac{cut(A,B)}{vol(B)}$$

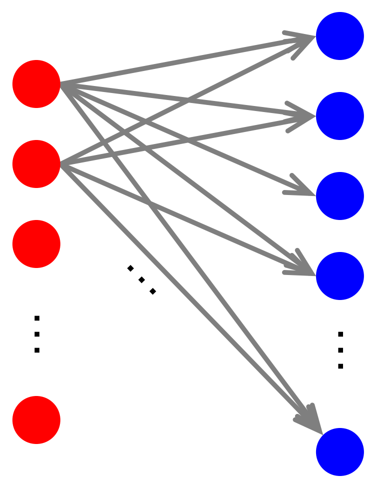
$\boldsymbol{vol(A)}$**:** total weight of the edges with at least one endpoint in $\boldsymbol{A}$: $\boldsymbol{vol(A)} = \sum_{\boldsymbol{i \in A}} \boldsymbol{k_i}$

- **Why use this criterion?**
  - Produces more balanced partitions
- **How do we efficiently find a good partition?**
  - **Problem:** Computing optimal cut is NP-hard

Analysis of Large Graphs: Trawling

Trawling

- **Searching for small communities in the Web graph**
- **What is the signature of a community / discussion in a Web graph?**



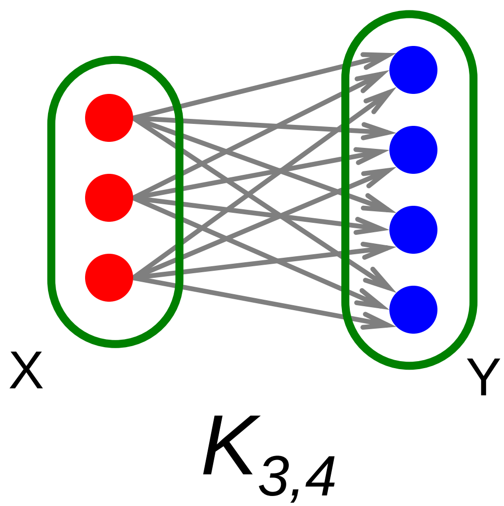
Dense 2-layer graph

Use this to define "topics":
What the same people on the left talk about on the right
Remember HITS!

Intuition: Many people all talking about the same things

Searching for Small Communities

- **A more well-defined problem:**
  Enumerate complete bipartite subgraphs $K_{s,t}$
  – Where $K_{s,t}$: s nodes on the "left" where each links to the same t other nodes on the "right"



$|X| = s = 3$
$|Y| = t = 4$

Fully connected

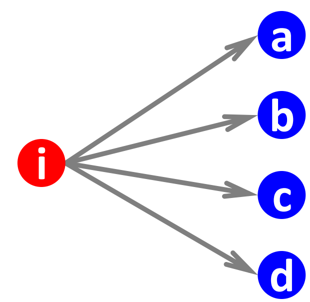
Frequent Itemset Enumeration

- **Market basket analysis.** Setting:
  - **Market:** Universe U of n items
  - **Baskets:** m subsets of U: $S_1, S_2, \ldots, S_m \subseteq U$ (S_i is a set of items one person bought)
  - **Support:** Frequency threshold f
- **Goal:**
  - **Find all subsets T s.t. $T \subseteq S_i$ of at least f sets S_i** (items in T were bought together at least f times)
- **What's the connection between the itemsets and complete bipartite graphs?**

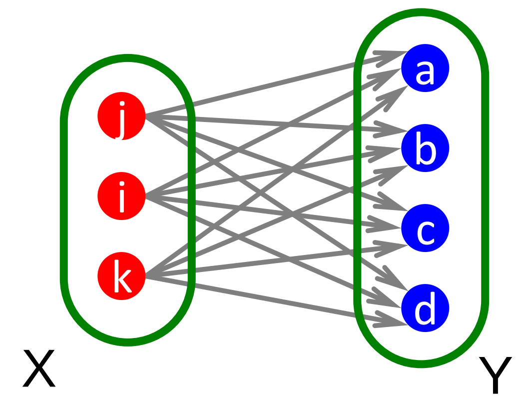
From Itemsets to Bipartite $K_{s,t}$

Frequent itemsets = complete bipartite graphs!

- **How?**
  - View each node $\boldsymbol{i}$ as a set $\boldsymbol{S_i}$ of nodes $\boldsymbol{i}$ points to
  - $\boldsymbol{K_{s,t}}$ = a set $\boldsymbol{Y}$ of size $\boldsymbol{t}$ that occurs in $\boldsymbol{s}$ sets $\boldsymbol{S_i}$
  - Looking for $\boldsymbol{K_{s,t}}$ → set of frequency threshold to $\boldsymbol{s}$ and look at layer $\boldsymbol{t}$ – all frequent sets of size $\boldsymbol{t}$



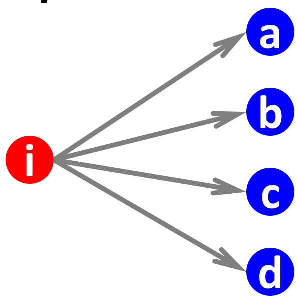
S_i={a,b,c,d}



s … minimum support (|X|=s)
t … itemset size (|Y|=t)

From Itemsets to Bipartite $K_{s,t}$

View each node ***i*** as a set $\boldsymbol{S_i}$ of nodes ***i*** points to

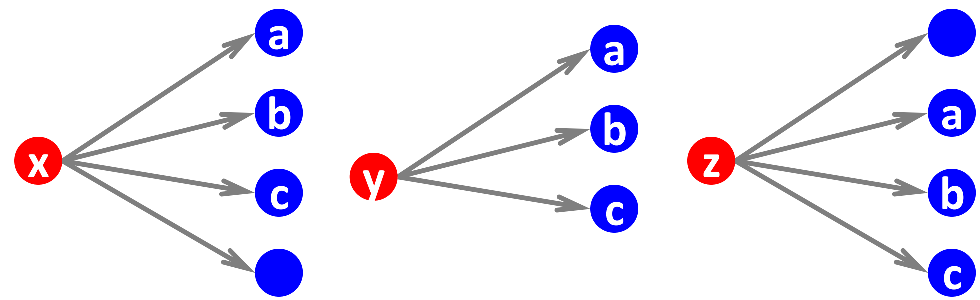


S_i={a,b,c,d}

Find frequent itemsets:
- **s** … minimum support
- **t** … itemset size

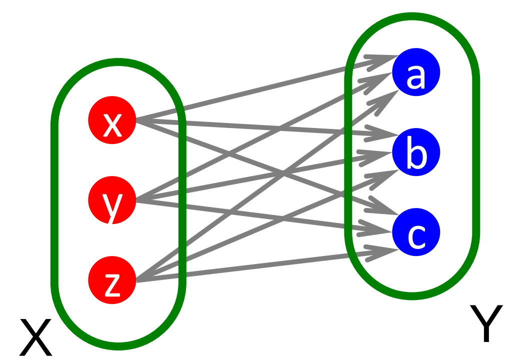
Say we find a **frequent itemset *Y={a,b,c}*** of supp ***s***

So, there are ***s*** nodes that link to all of **{a,b,c}**:

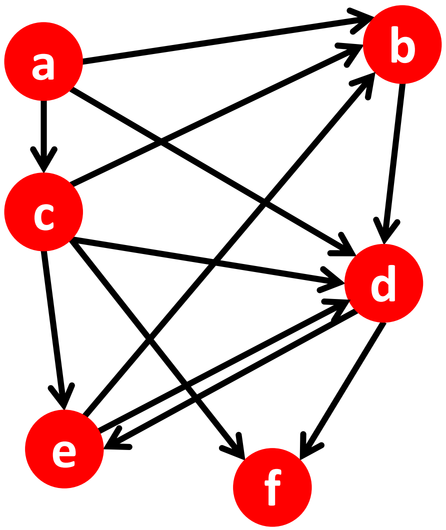


We found *$K_{s,t}$*!

$K_{s,t}$ = a set ***Y*** of size ***t*** that occurs in *s* sets $\boldsymbol{S_i}$



Example (1)



Itemsets:

a = {b,c,d}

b = {d}

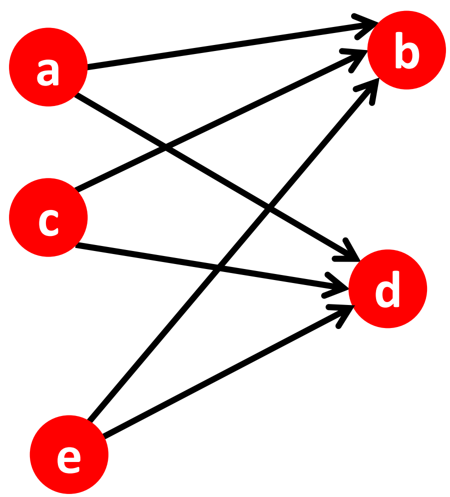
c = {b,d,e,f}

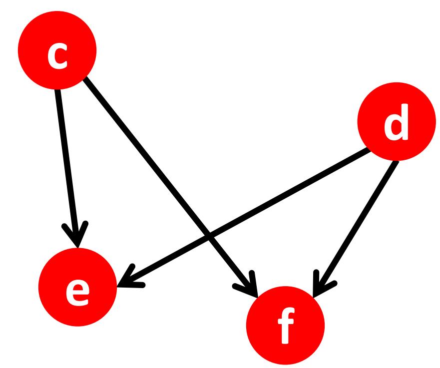
d = {e,f}

e = {b,d}

f = {}

- **Support threshold s=2**
  - **{b,d}**: support 3
  - **{e,f}**: support 2
- **And we just found 2 bipartite subgraphs:**





Example (2)

- **Example of a community from a web graph**

A community of Australian fire brigades

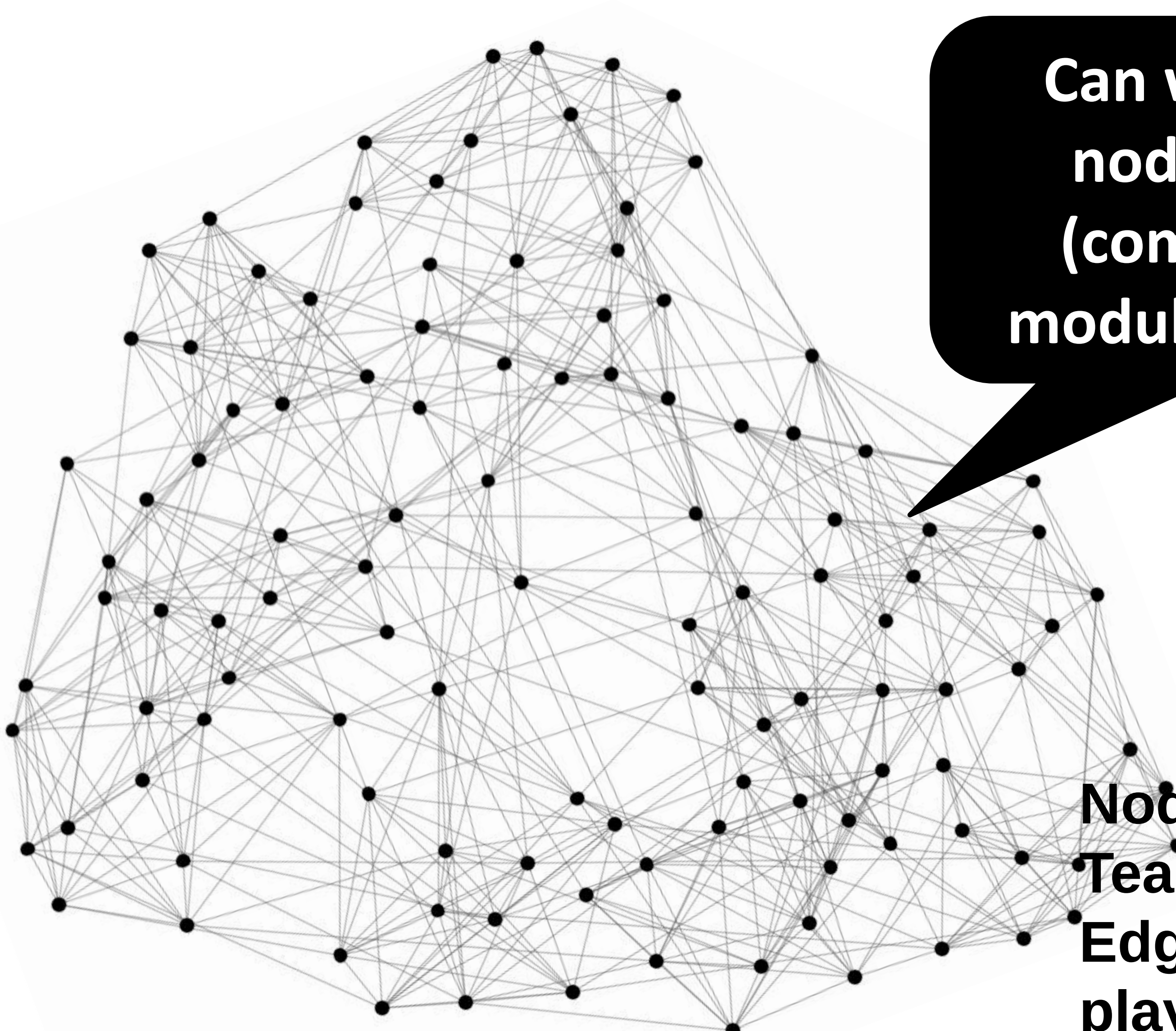
Nodes on the right	Nodes on the left
NSW Rural Fire Service Internet Site	New South Wales Fir...ial Australian Links
NSW Fire Brigades	Feuerwehrlinks Australien
Sutherland Rural Fire Service	FireNet Information Network
CFA: County Fire Authority	The Cherrybrook Rur...re Brigade Home Page
"The National Cente...ted Children's Ho...	New South Wales Fir...ial Australian Links
CRAFTI Internet Connexions-INFO	Fire Departments, F... Information Network
Welcome to Blackwoo... Fire Safety Serv...	The Australian Firefighter Page
The World Famous Guestbook Server	Kristiansand brannv...dens brannvesener...
Wilberforce County Fire Brigade	Australian Fire Services Links
NEW SOUTH WALES FIR...ES 377 STATION	The 911 F,P,M., Fir...mp; Canada A Section
Woronora Bushfire Brigade	Feuerwehrlinks Australien
Mongarlowe Bush Fire – Home Page	Sanctuary Point Rural Fire Brigade
Golden Square Fire Brigade	Fire Trails "l...ghters around the...
FIREBREAK Home Page	FireSafe – Fire and Safety Directory
Guises Creek Volunt...fficial Home Page...	Kristiansand Firede...departments of th...

[Kumar, Raghavan, Rajagopalan, Tomkins: Trawling the Web for emerging cyber-communities 1999]

Analysis of Large Graphs: Overlapping Communities

Mining of Massive Datasets
Jure Leskovec, Anand Rajaraman, Jeff Ullman
Stanford University
http://www.mmds.org

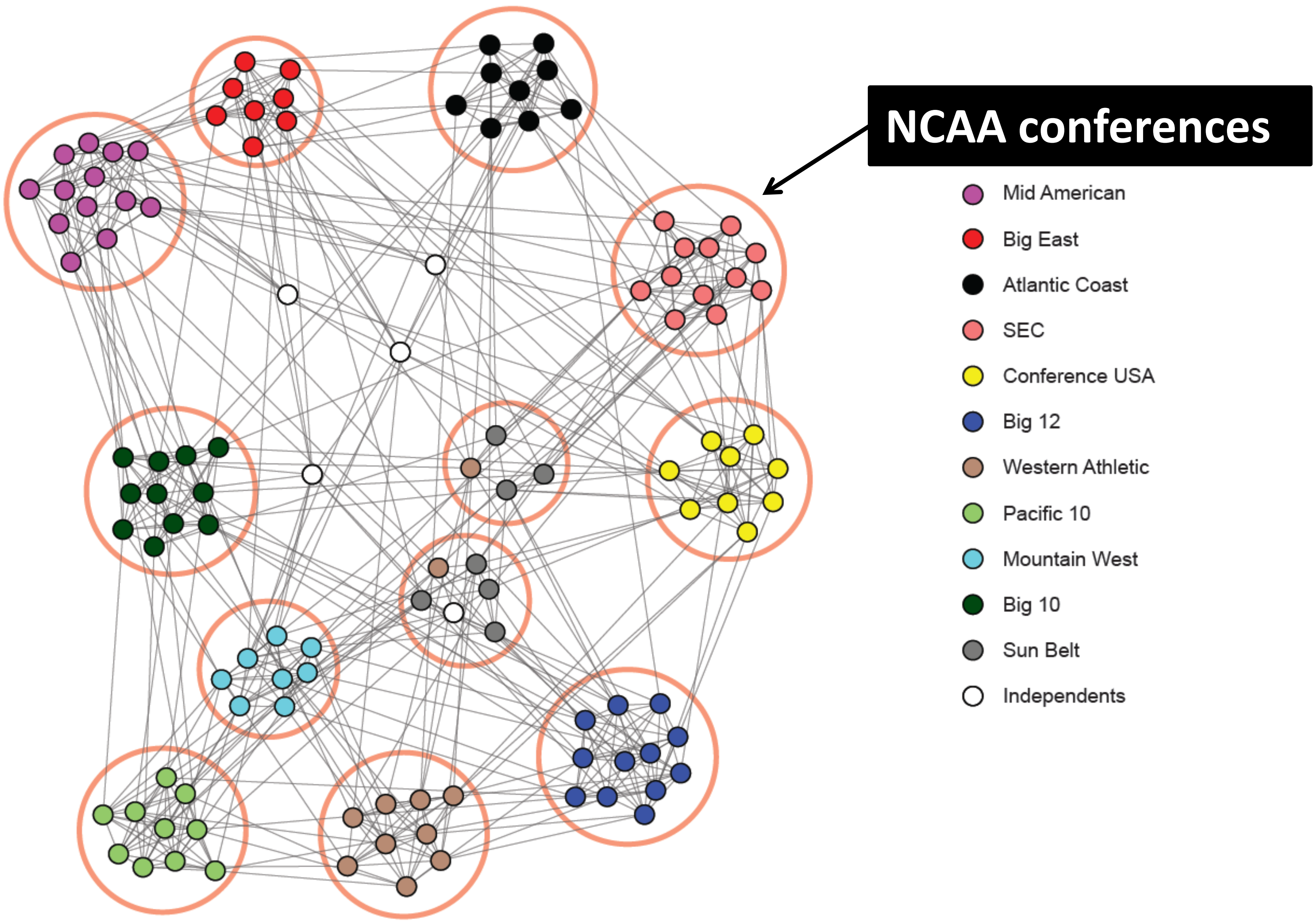
Identifying Communities



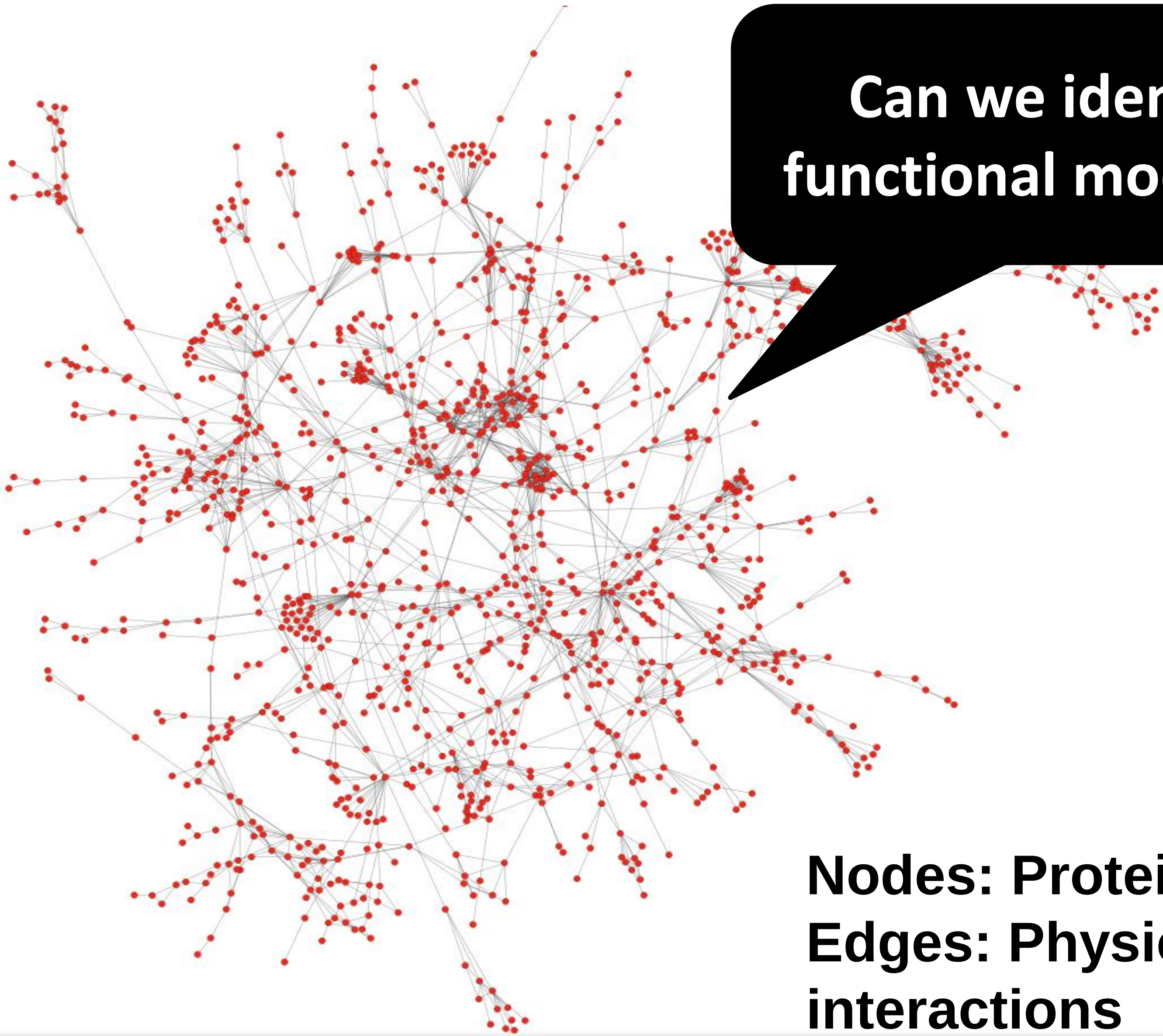
Can we identify node groups?
(communities, modules, clusters)

Nodes: Football Teams
Edges: Games played

NCAA Football Network



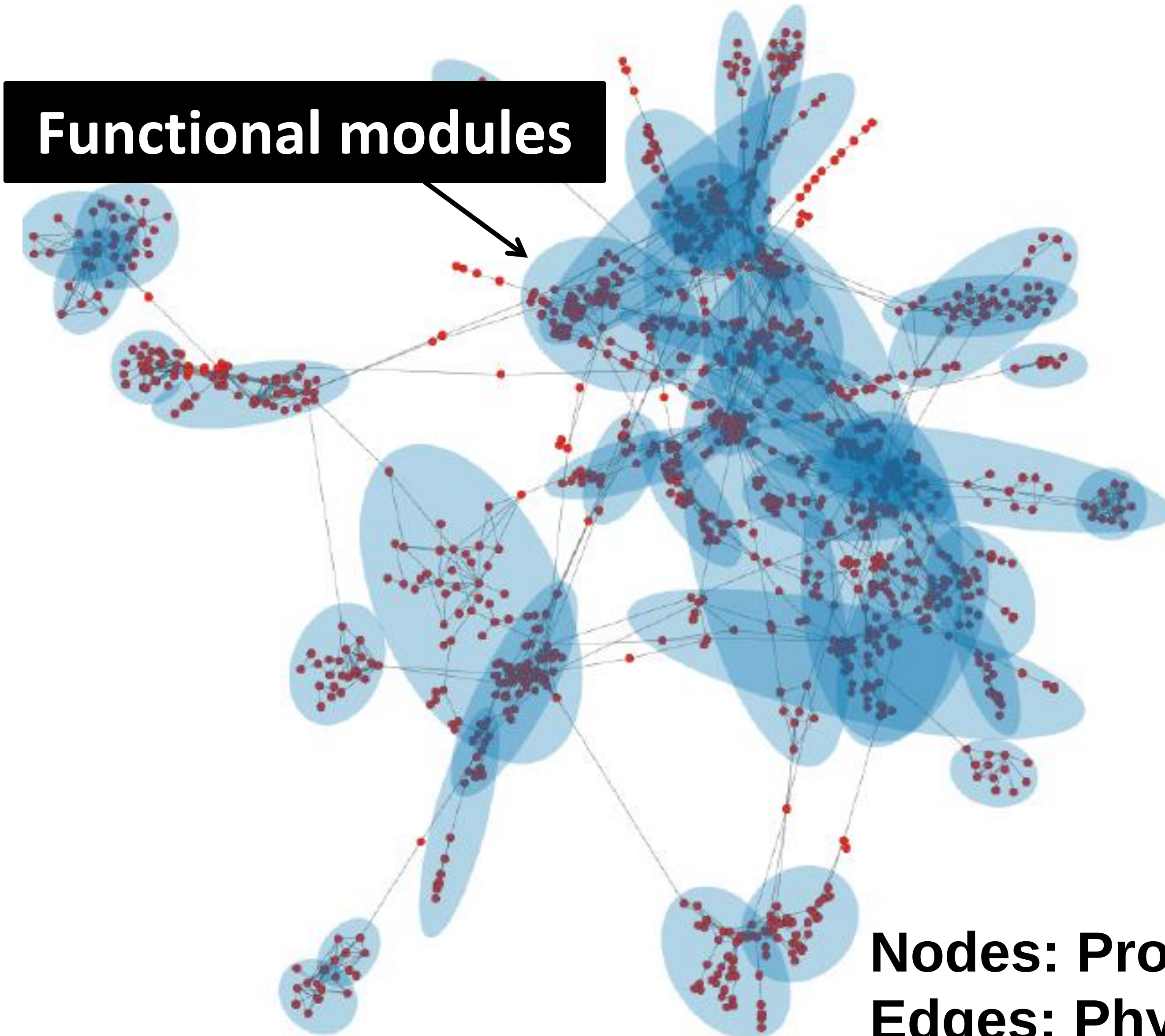
Protein-Protein Interactions



Nodes: Proteins
Edges: Physical interactions

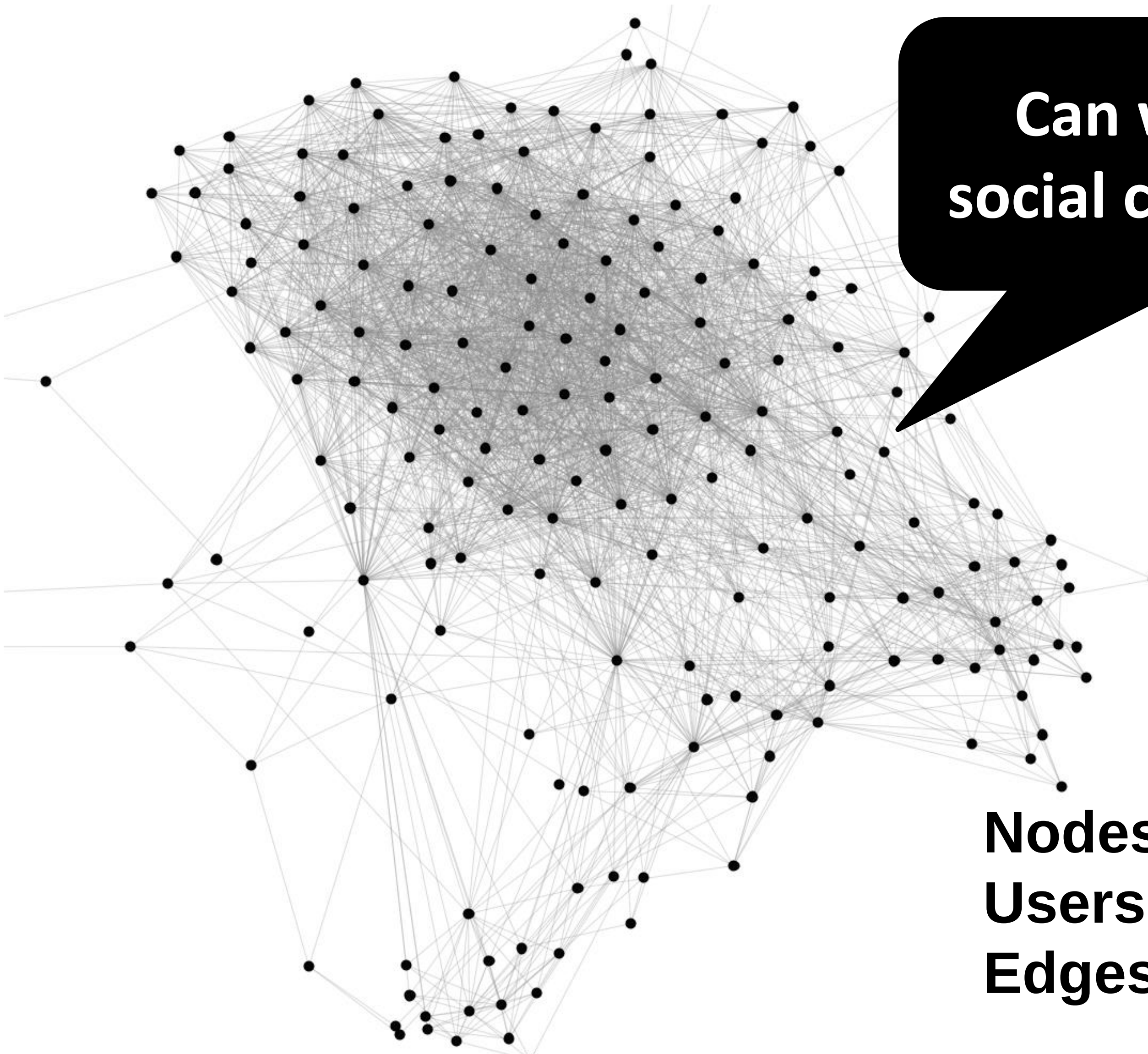
Protein-Protein Interactions

Functional modules



Nodes: Proteins
Edges: Physical

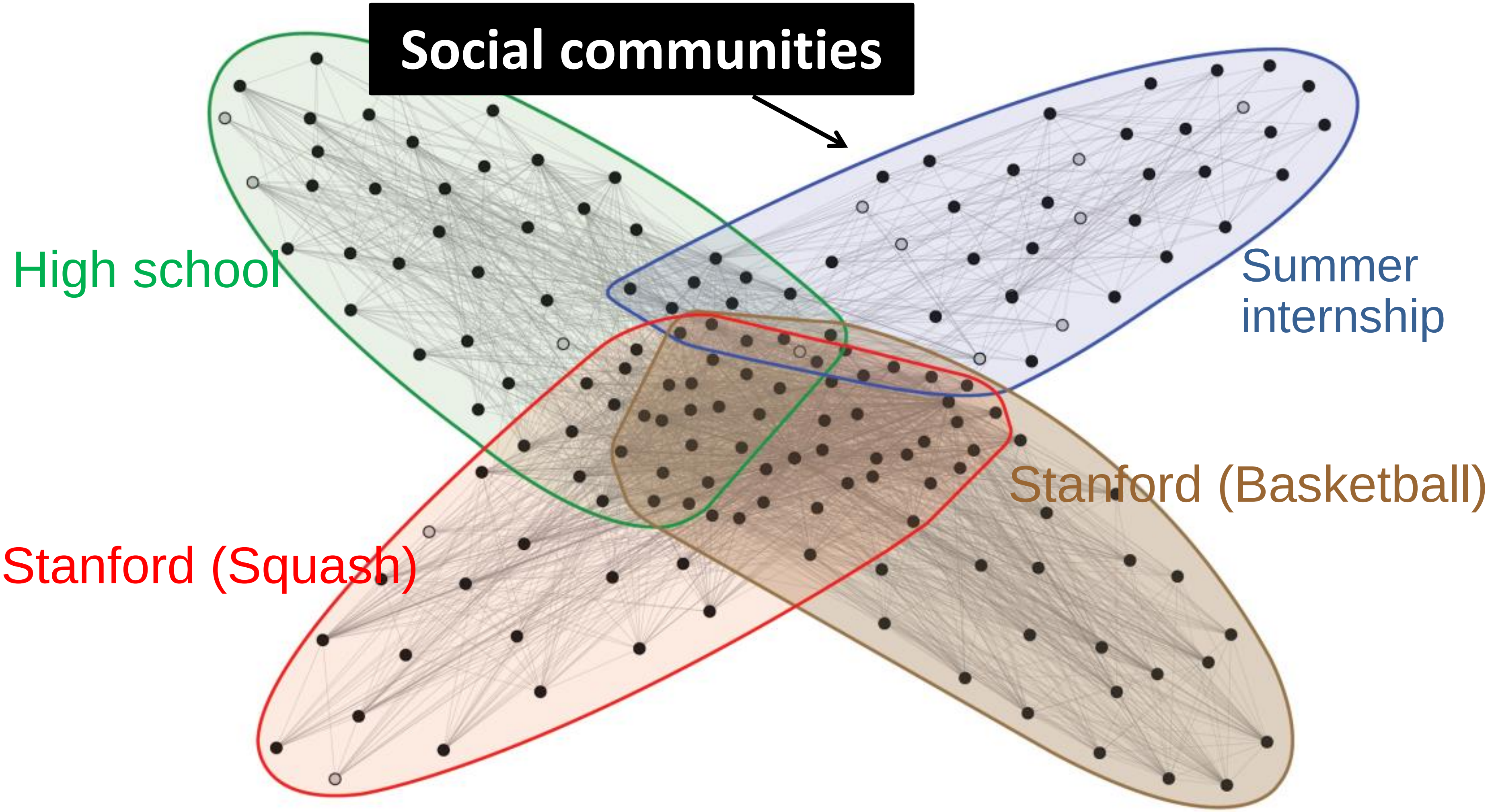
Facebook Network



Can we identify social communities?

Nodes: Facebook Users
Edges: Friendships

Facebook Network



More details at…

- Overlapping Community Detection at Scale: A Nonnegative Matrix Factorization Approach by J. Yang, J. Leskovec. *ACM International Conference on Web Search and Data Mining (WSDM)*, 2013.
- Detecting Cohesive and 2-mode Communities in Directed and Undirected Networks by J. Yang, J. McAuley, J. Leskovec. *ACM International Conference on Web Search and Data Mining (WSDM)*, 2014.
- Community Detection in Networks with Node Attributes by J. Yang, J. McAuley, J. Leskovec. *IEEE International Conference On Data Mining (ICDM)*, 2013.