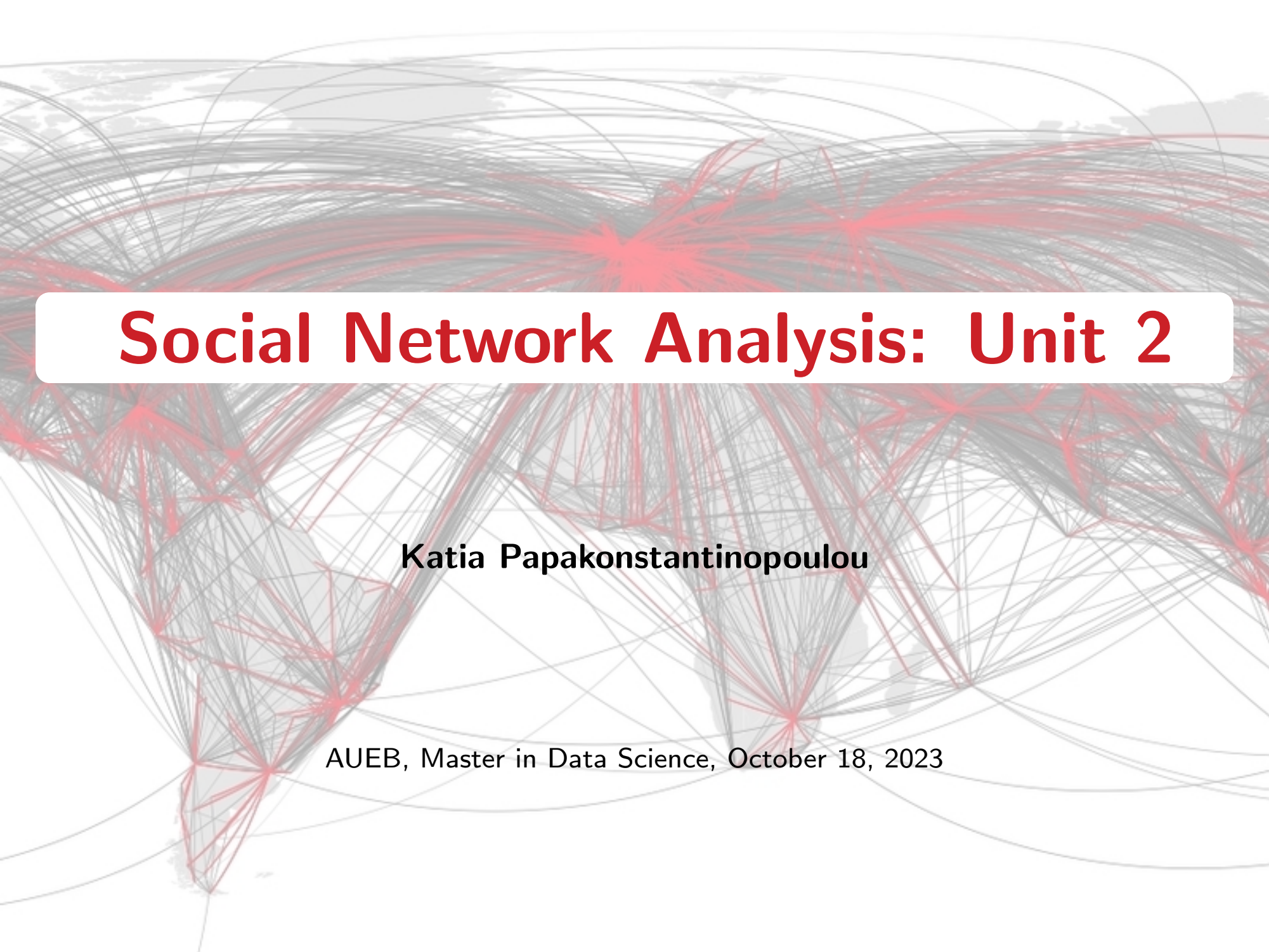


Social Network Analysis: Unit 2

Katia Papakonstantinopoulou

AUEB, Master in Data Science, October 18, 2023

Unit 2: Social Network Structure

Outline:

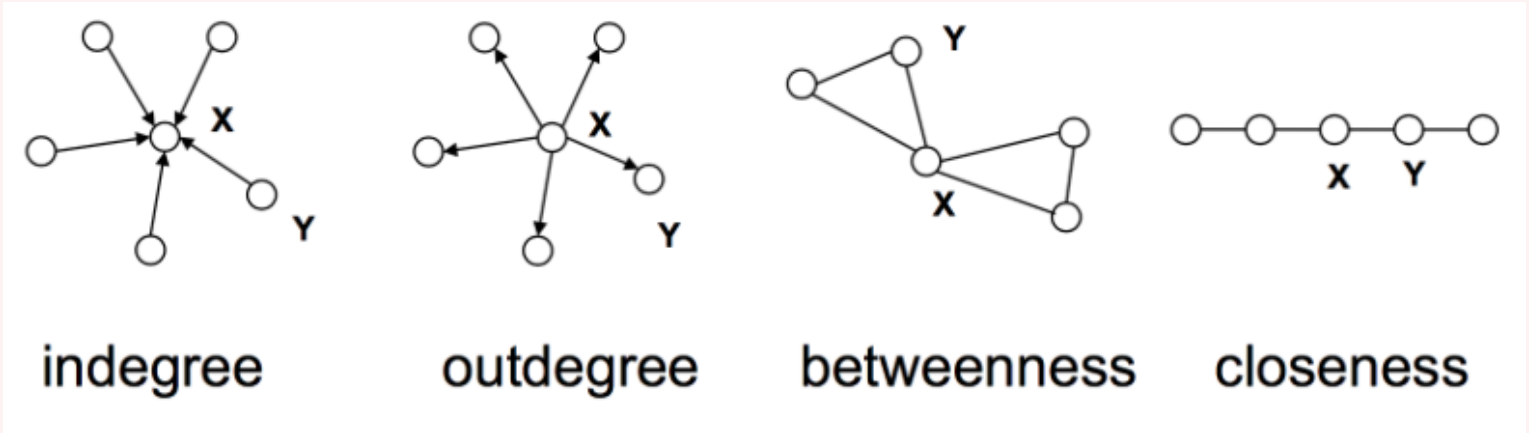
- Centrality measures, Betweenness, Closeness
- Community detection
- Overlapping community detection
- LAB SESSION: SNAP

Centrality Measures

- In many applications we are interested in finding the most *central* nodes. However, relying on their degree is not enough!
- Different notions of *centrality*
  - Degree centrality (in-degree, out-degree)
  - Eigenvector centrality, PageRank centrality
  - Closeness centrality
  - Betweenness centrality

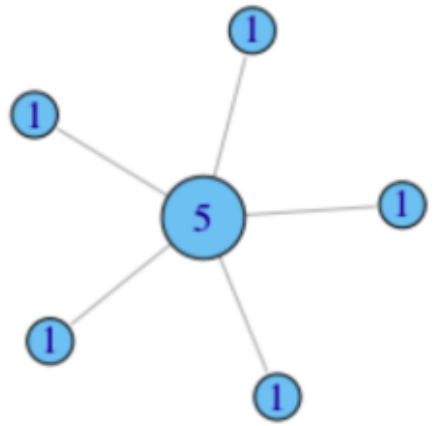
Examples

In each of the following networks, X has higher centrality than Y according to a particular measure:



Undirected degree

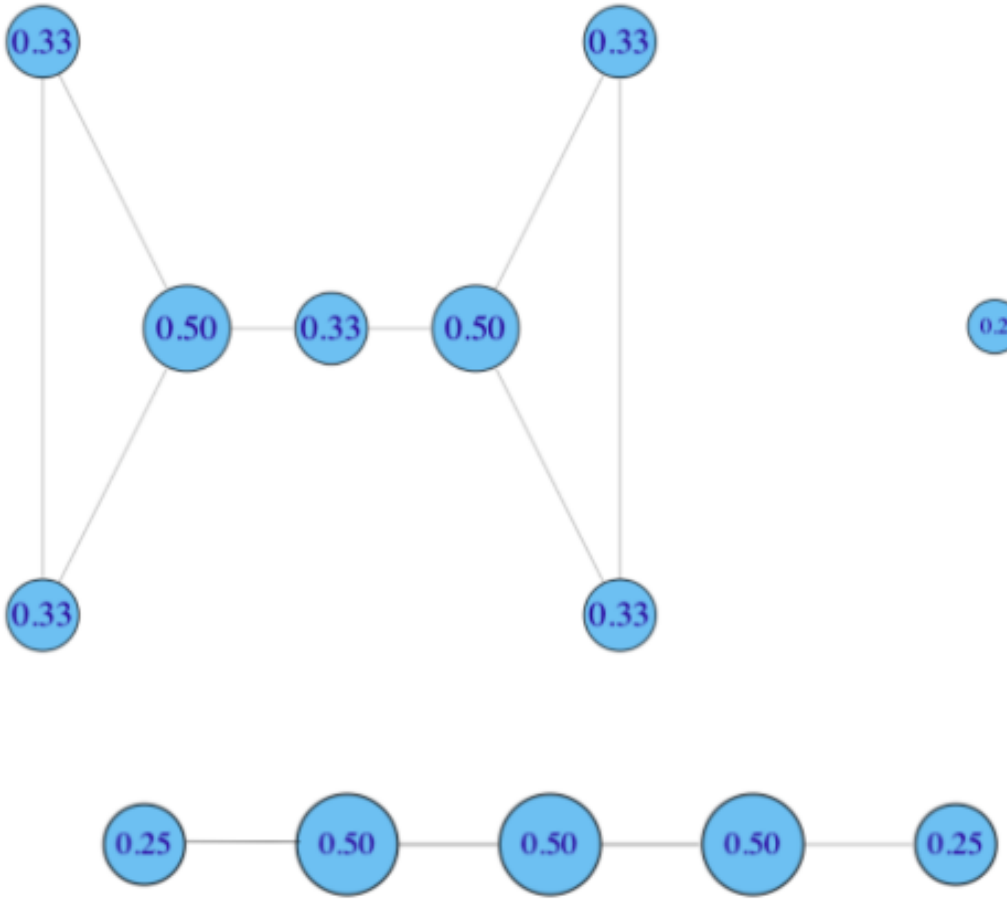
Idea: Nodes with more friends are more central



Assumption: We do not care about our friends' connections!

Degree normalization

We normalize the degrees by dividing each degree with the maximum possible degree, i.e., $n - 1$:



Degree Centralization

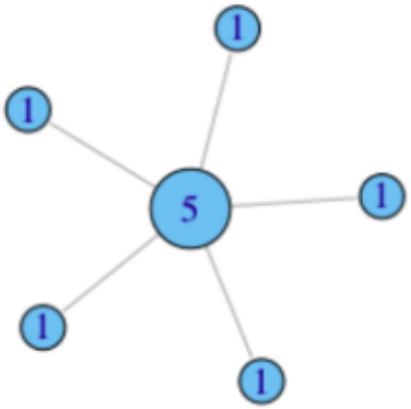
How much variation is there in the centrality scores among the nodes?

Considering Freeman's general formula for centralization,

$$C_D = \frac{\sum_{i=1}^{g}[C_D(n^*) - C_D(i)]}{(N-1)(N-2)}$$

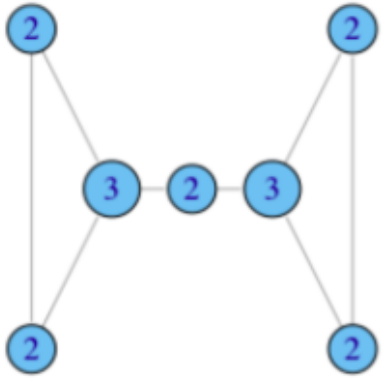
where $C_D(n^*)$ is the maximum value in the network.

Degree Centralization Examples



$C_D = 1.0$

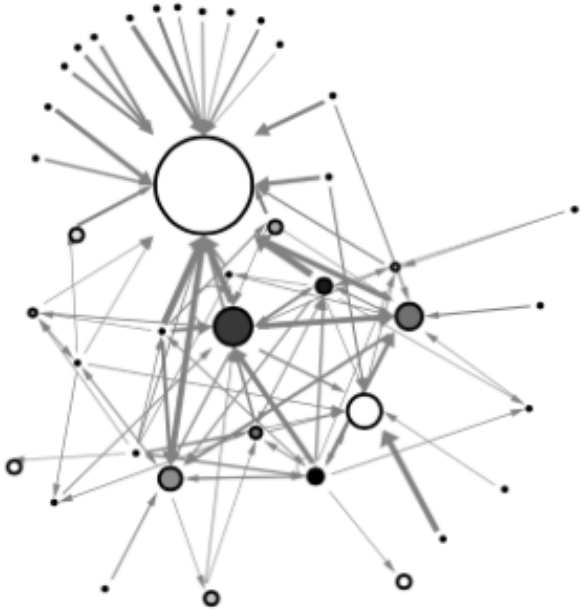
$C_D = 0.167$



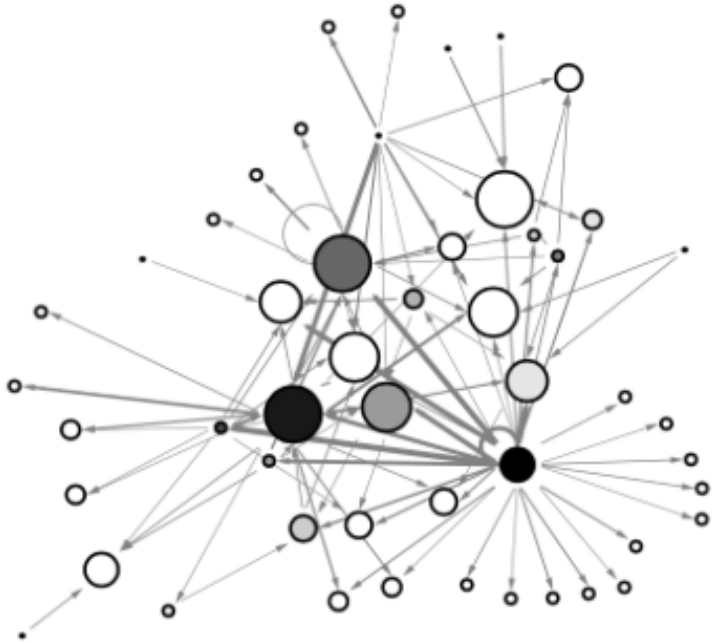
$C_D = 0.167$

Real-World Degree Centralization Examples

Similar networks may exhibit varying degree of in-centralization.
Example financial trading networks:



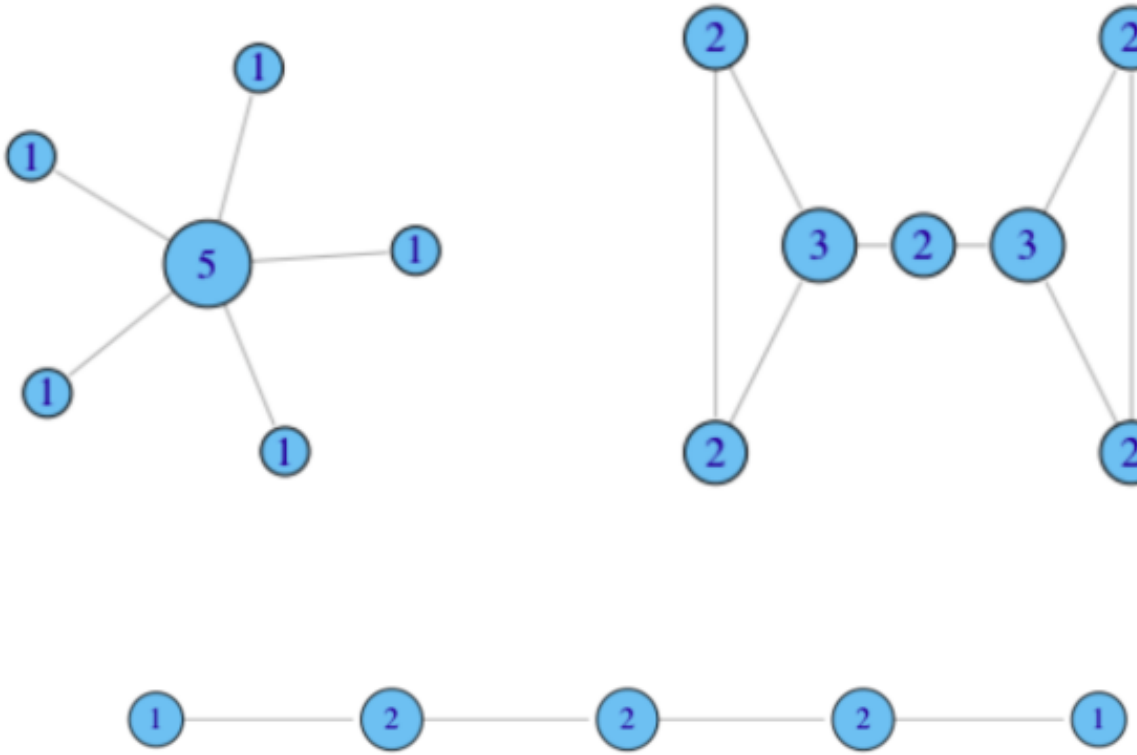
high in-centralization:
one node buying from
many others



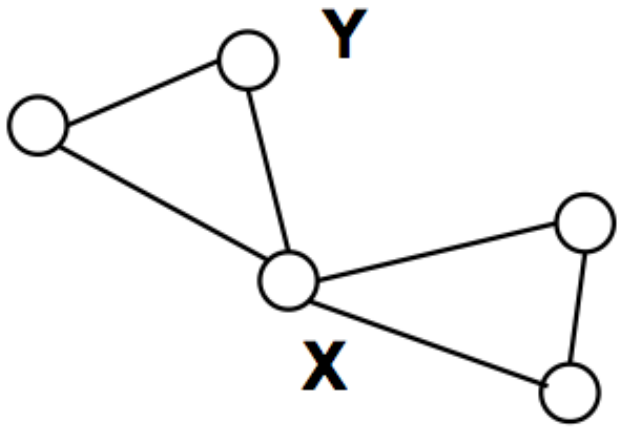
low in-centralization:
buying is more evenly
distributed

Degree often fails to capture centrality

In what ways does the degree fail to capture centrality in the following examples?



Degree does not capture brokerage!



X is in contact with nodes Y is totally unaware of.

Closeness

The closeness of node x is defined as:

$$C(x) = \frac{1}{\sum_y d(x,y)}.$$

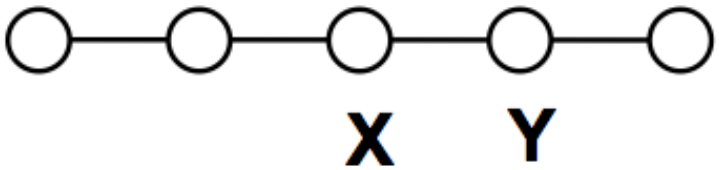
Its normalized form is:

$$C(x) = \frac{N}{\sum_y d(x,y)},$$

where N is the number of nodes in the graph.

Betweenness captures brokerage

Intuition: How many pairs of individuals would have to go through a specific node in order to reach one another in the minimum number of hops?



Betweenness

Betweenness is defined as:

$$C_B(i) = \sum_{j<k} \frac{g_{jk}(i)}{g_{jk}}$$

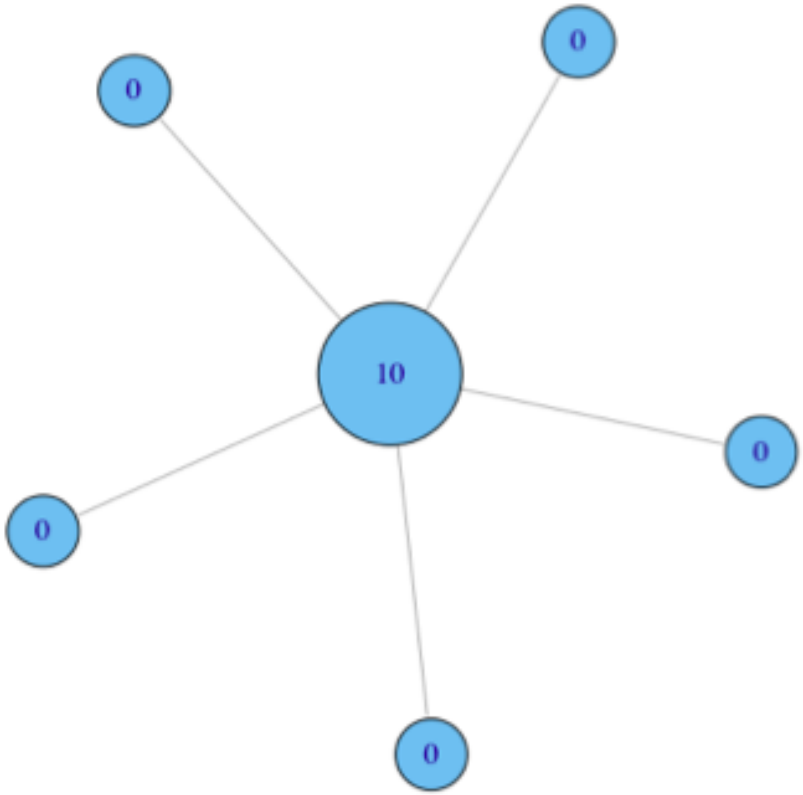
where g_{jk} is the number of shortest paths connecting j and k and $g_{jk}(i)$ is the number of the above paths that go through i.

Betweenness is usually normalized by the number of pairs of vertices excluding the vertex itself:

$$C'_B(i) = \frac{C_B(i)}{\frac{(n-1)(n-2)}{2}}$$

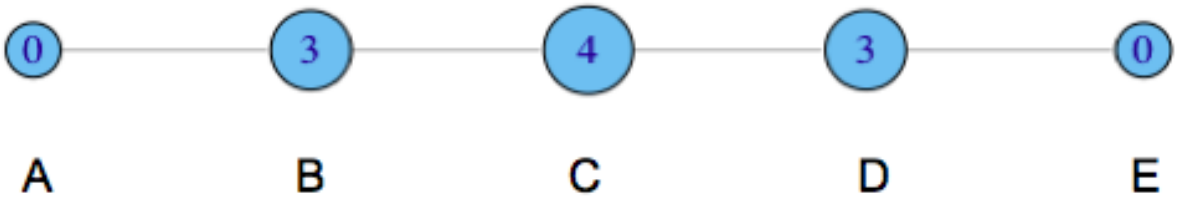
Betweenness on simple networks

Non normalized version



Betweenness on simple networks

Non normalized version

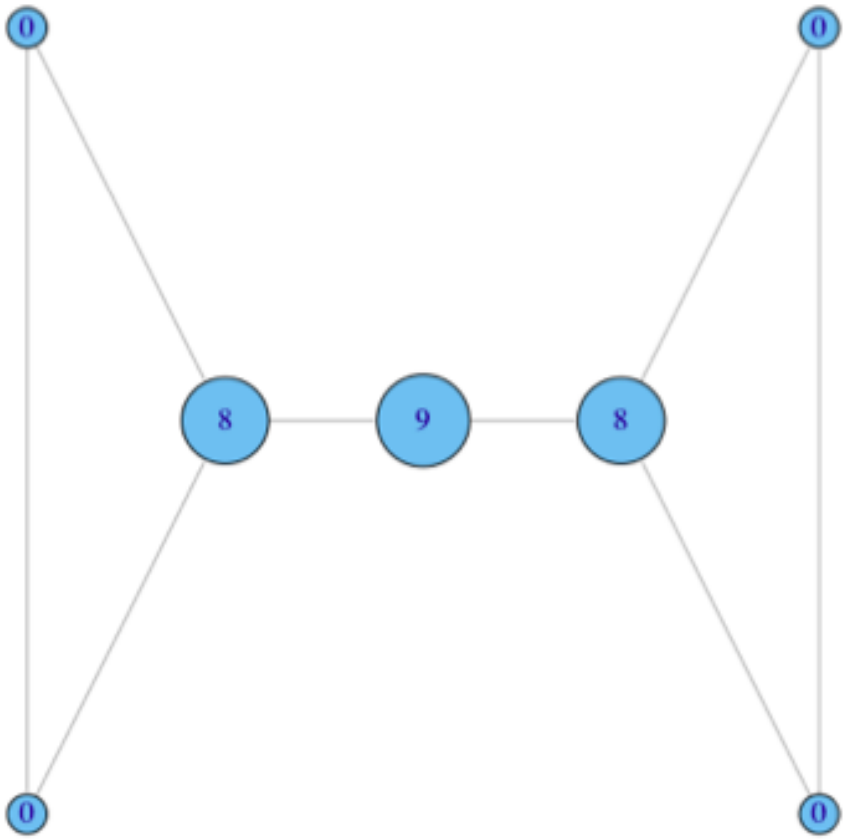


- A lies between no two other vertices
- B lies between A and 3 other vertices: C, D, and E
- C lies between 4 pairs of vertices (A,D),(A,E),(B,D),(B,E)

- note that there are no alternate paths for these pairs to take, so C gets full credit

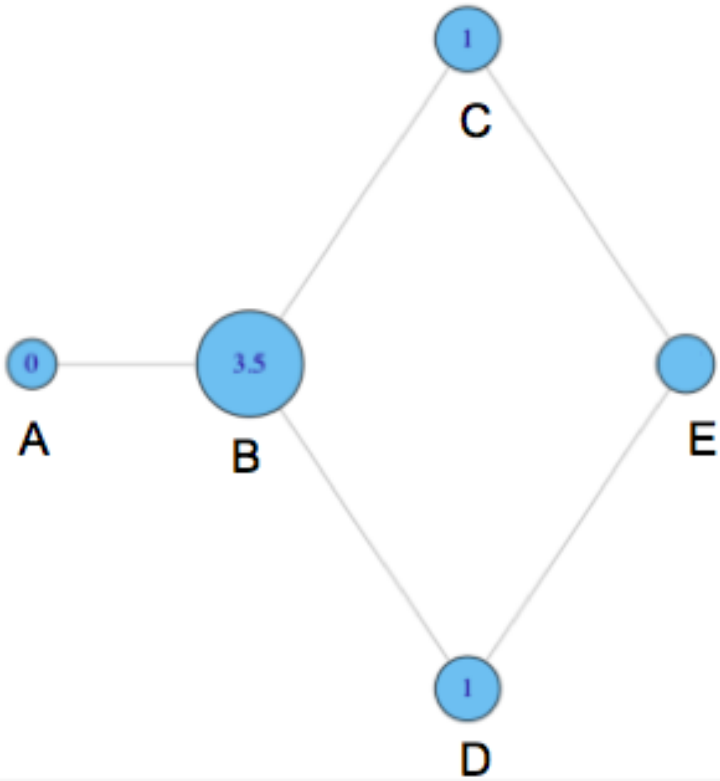
Betweenness on simple networks

Non normalized version



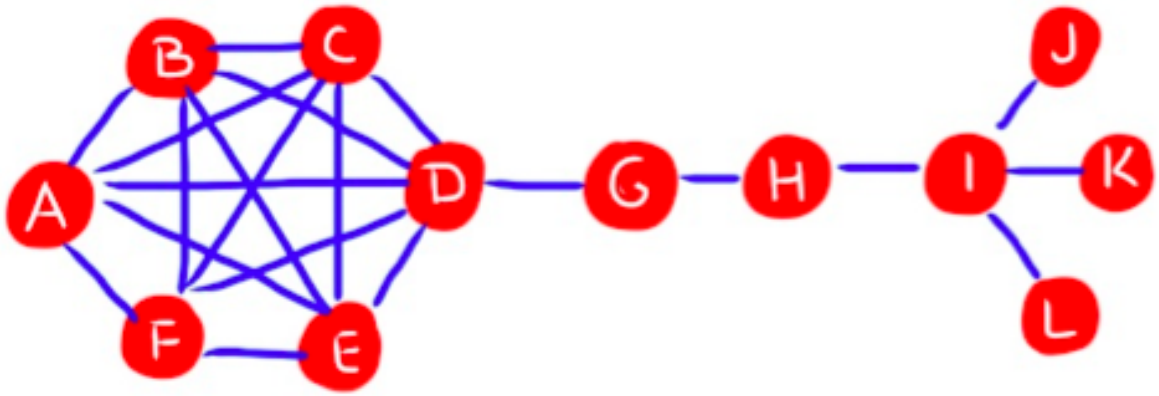
Betweenness on simple networks

Non normalized version



- why do C and D each have betweenness 1?
- They are both on shortest paths for pairs (A,E), and (B,E), and so must share credit:
  - ½+½ = 1

Betweenness on simple networks



- Find a node that has high betweenness but low degree
- Find a node that has low betweenness but high degree

Girvan-Newman algorithm for community detection

It extends the betweenness definition to the case of edges, defining the "edge betweenness" of an edge as the number of shortest paths between pairs of nodes that run along it.

Algorithm:

1. Calculate the betweenness of all existing edges in the network
2. Remove the edge with the highest betweenness
3. Recalculate the betweenness of all edges affected by the removal
4. Repeat steps 2 and 3 until no edges remain

Betweenness and Modularity cont.

Remarks on the Girvan-Newman algorithm for community detection:

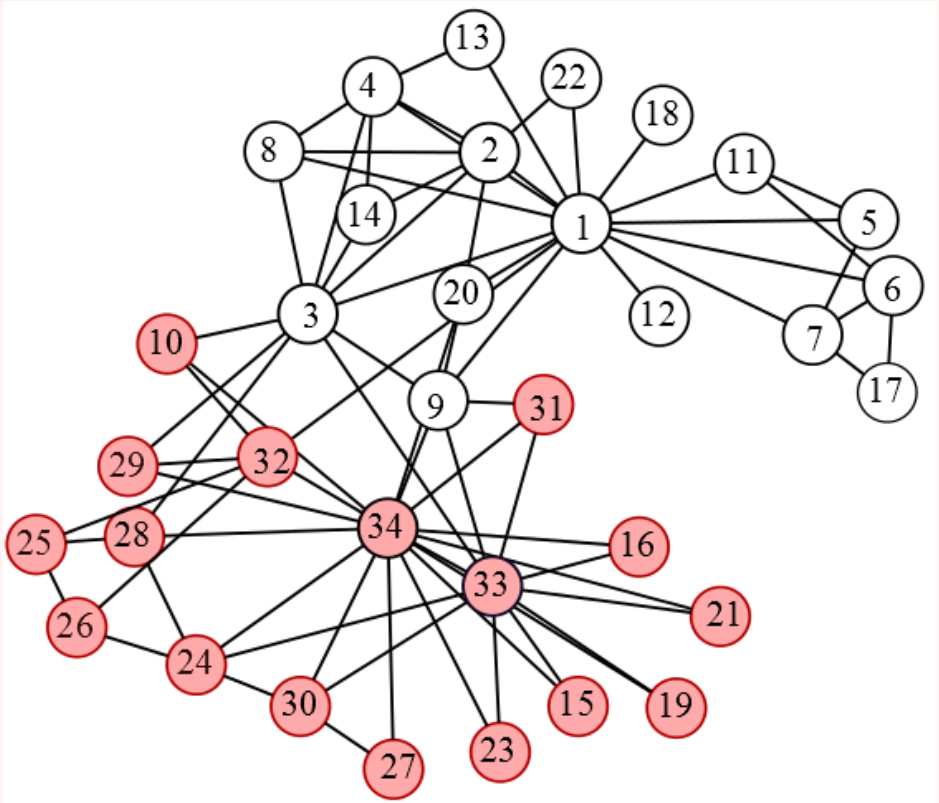
- This algorithm eventually decomposes the graph into single nodes. We can get the communities by stopping the algorithm at any previous step we like. The optimal stop is determined using the *modularity* value.
- Instead of trying to identify edges that are *central* to communities, it focuses on edges that are most likely *between* communities.

Community Detection with gephi

Zachary Karate club – Description

- Open file karate.gephi
- 34 members of a karate club, 78 pairwise links between members who interacted outside the club
- Conflict split of the club into two groups
- Zachary's study (using a maximum-flow minimum cut algorithm) predicted correctly the groups all but one members of the club joined
- Using modularity algorithm in gephi with resolution 2 you can find (predict) the two classes!

Zachary Karate club – Result



Zachary Karate club – Conclusion

Based solely on the network structure, we can predict the members that will leave the club.

LAB SESSION: Intro to SNAP

SNAP (Stanford Network Analysis Platform)

A system for large networks' analysis and manipulation

SNAP use

- We will use SNAP operations in python scripts:
  Snap.py: Python interface for SNAP
- In your python file, start with: `from snap import *`
- A detailed tutorial: `https://snap.stanford.edu/snappy/doc/tutorial/index-tut.html`

Assignment on SNAP

2 Problems:

1. Develop Python functions that receive a graph as an input and examine whether it contains an Euler path or an Euler circuit, and complete a related test-case.
2. Generate random graphs, apply node centrality measures and community detection algorithms on them and report results regarding their efficiency.

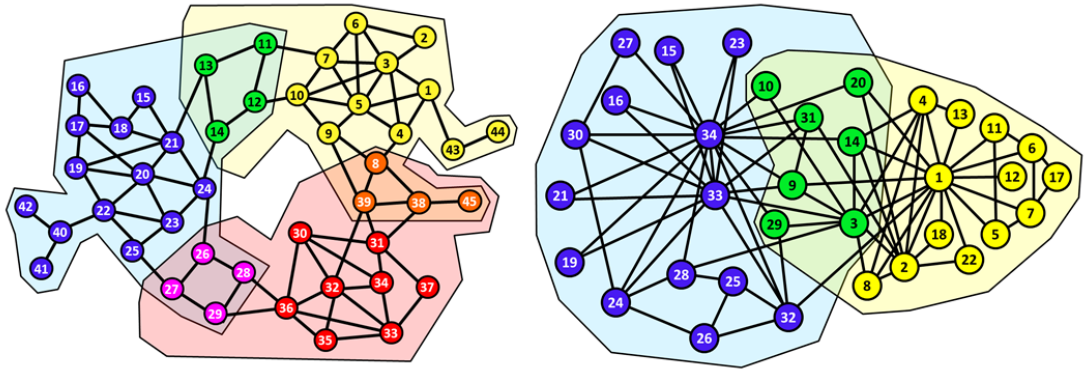
Overlapping Community Detection

Communities may be distinct:

- Members of **John A** and **Mr. Hi** karate clubs
- Fans of Olympiakos and Panathinaikos

However, it is often the case that communities *overlap*:

- Social circles of individuals (family, friends, co-workers in a social network)
- Proteins belong to several protein complexes simultaneously (protein-protein interaction network)



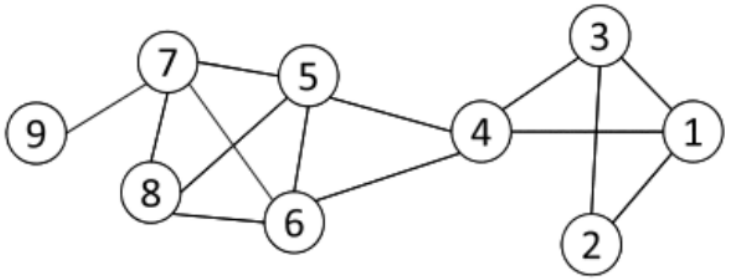
Clique Percolation Method (Palla et al. Nature, 2005)

A popular technique for overlapping community detection is the Clique Percolation Method (CPM)

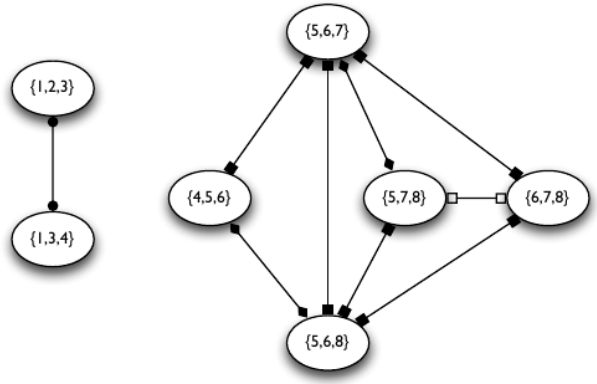
- The internal edges of a community are likely to form cliques due to their high density.
- It is unlikely that inter-community edges form cliques.
- If it were possible for a clique to move on a graph, in some way, it would probably get trapped inside its original community, as it could not cross the bottleneck formed by the inter-community edges.

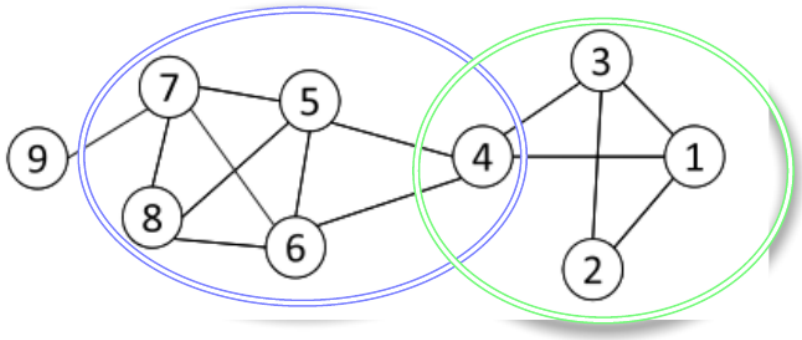
Clique Percolation Method (cont'd)

- Given a parameter k:
  - Find all the cliques of size k
  - Construct a clique graph in which two cliques are adjacent if they share k-1 vertices
  - A community is defined as the maximal union of k-cliques that can be reached from each other through a series of adjacent k-cliques.



cliques of size 3= {1,2,3}, {1,3,4},{4,5,6},{5,6,7}, {5,6,8},{5,7,8},{6,7,8}





Link Communities (Ahn et al. Nature, 2010)

- A vertex can belong to several communities (an individual belongs to many social groups)
- A link is usually related to only one community (a link between two individuals exists because of relation, friendship, etc.)

Cluster the edges instead of the nodes!

- After obtaining each cluster the communities can be formed by replacing an edge with the adjacent vertices.
- A link is usually related to only one community (a link between two individuals exists because of relation, friendship, etc.)

Live demo:

`http://scaledinnovation.com/analytics/communities/comlinks.html`

Unit 2: Recap

We discussed the following aspects:

- Centrality measures, Betweenness
- Community detection
- Overlapping community detection
- LAB SESSION: SNAP