Social Media & Network

Community Analysis

Social Community

[real-world] community

A group of individuals with common *economic*, *social*, or *political* interests or characteristics, often living in *relative proximity*.

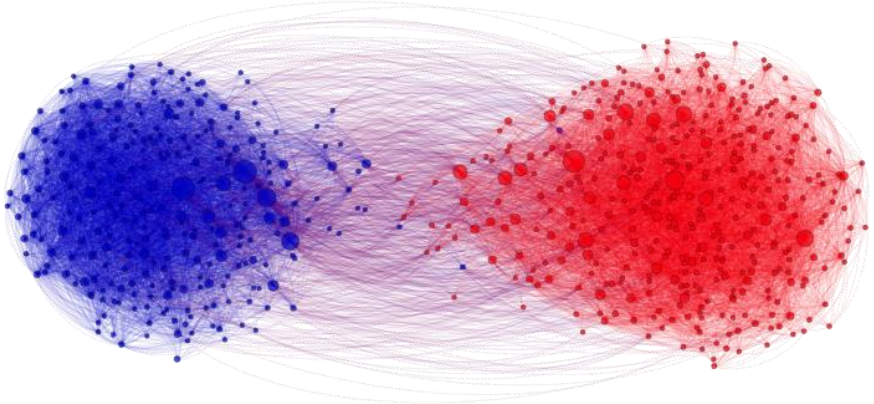
Why analyze communities?

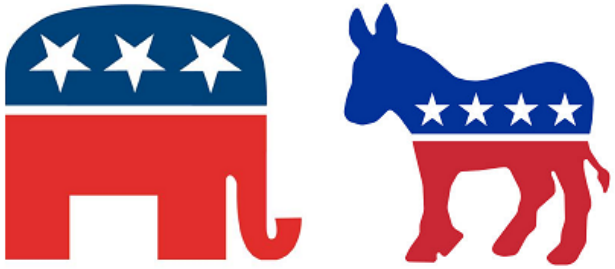
Analyzing communities helps better understand users

- Users form groups based on their interests

Groups provide a clear global view of user interactions

- E.g., find polarization





Some behaviors are only observable in a group setting and not on an individual level

- Some republican can **agree** with some democrats, but their parties can **disagree**

Social Media Communities

- **Formation:**
  - When like-minded users on social media form a link and start interacting with each other

- **More Formal Formation**:
  1. A set of at least two nodes sharing some interest, and
  2. Interactions with respect to that interest.

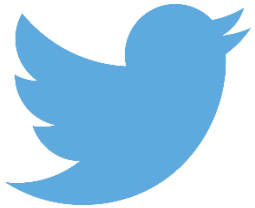
- Social Media Communities
  - **Explicit** (*emic*): formed by user subscriptions
  - **Implicit** (*etic*): implicitly formed by social interactions
    - **Example:** individuals calling Canada from the United States
    - Phone operator considers them one community for promotional offers

- Other community names: *group*, *cluster*, *cohesive subgroup*, or *module*

Examples of Explicit Social Media Communities

Facebook has groups and communities. Users can join and

- post messages and images,
- can comment on other messages,
- can like posts, and
- can view activities of other users



In Twitter, communities form as lists.

– Users join lists to receive information in the form of tweets

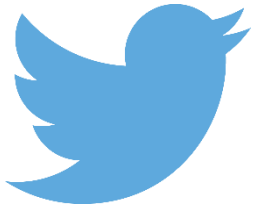
LinkedIn provides *Groups* and *Associations*.

– Users can join professional groups where they can post and share information related to the group

Examples of Implicit Social Media Communities

Facebook, from friendship links, infer

- family and friends communities
- workmates
- professional



In Twitter

– Analyse follower networks to find common interest communities

In LinkedIn

– Analyse association graph to find different professional groups

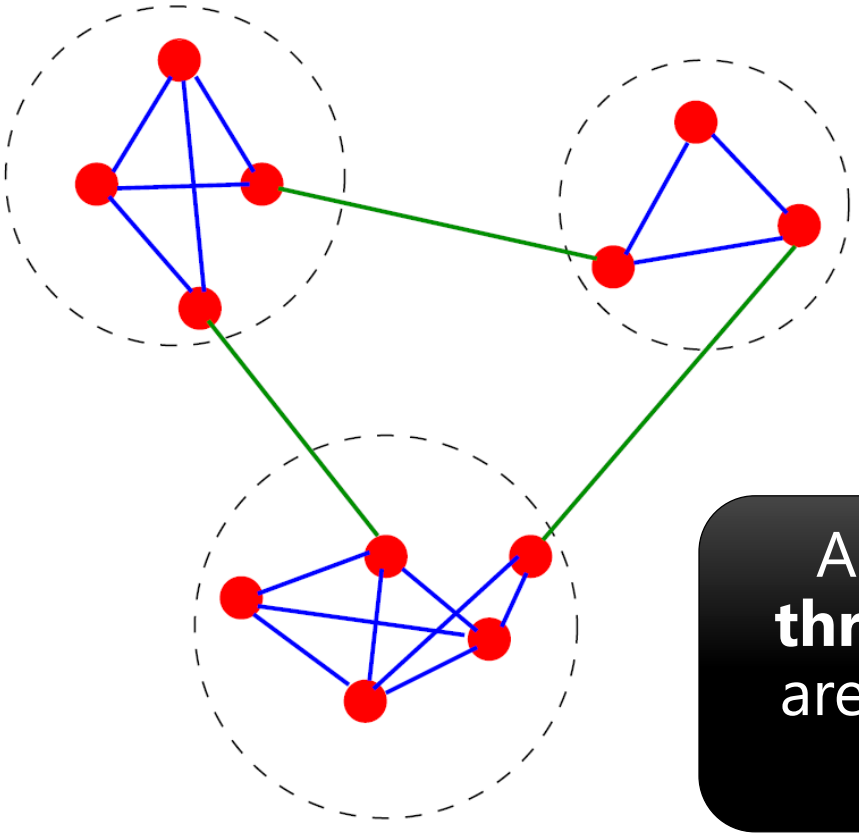
What is Community Analysis?

- **Community detection**
  - Discovering implicit communities
- **Community evaluation**
  - Evaluating Detected Communities

Community Detection

What is a Community in Graphs?

- A group of nodes that are strongly connected to each other and weakly connected to other communities/groups in the graph



A simple graph in which **three** implicit communities are found, enclosed by the dashed circles

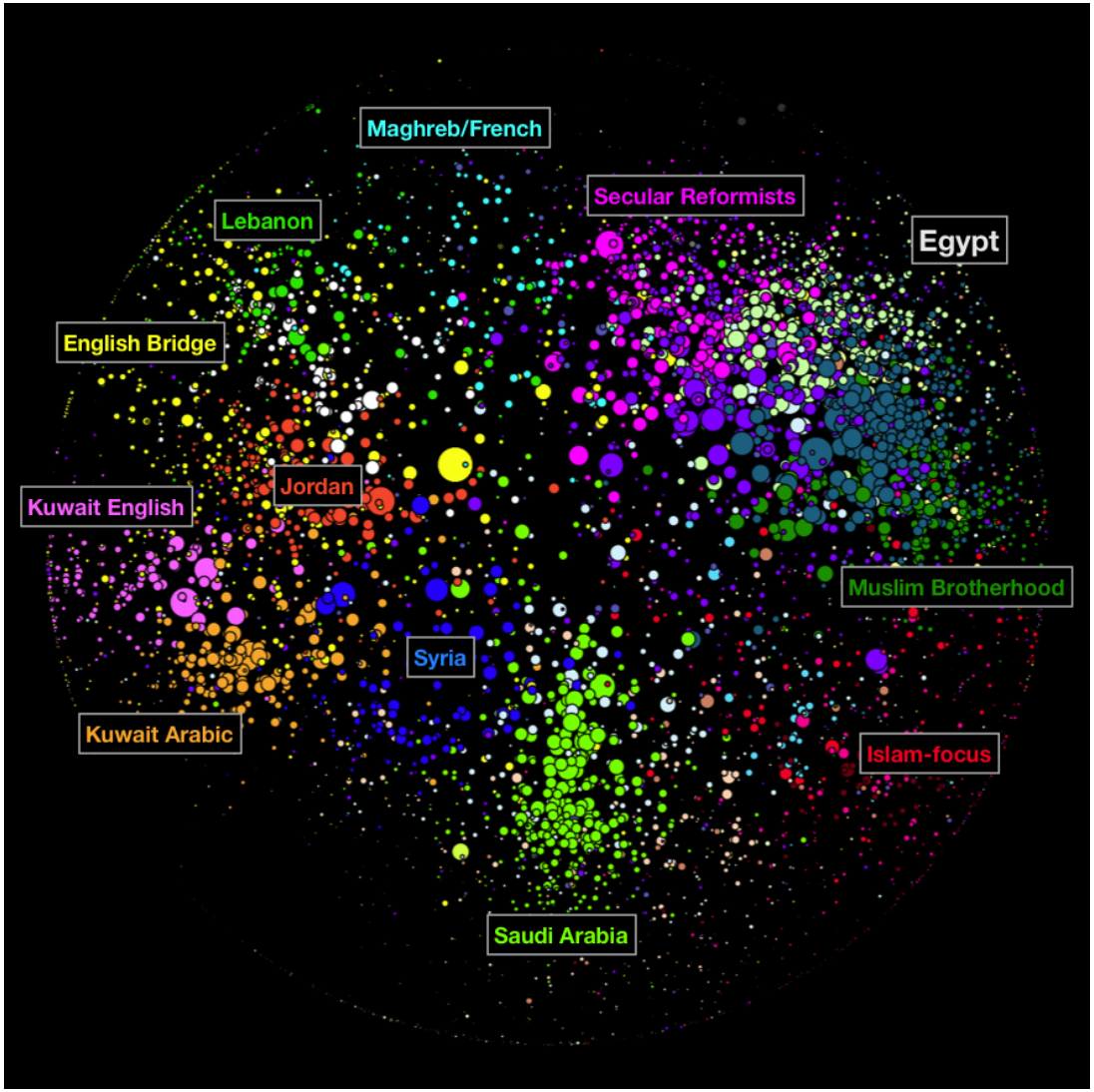
Why Community Detection?

Network Summarization

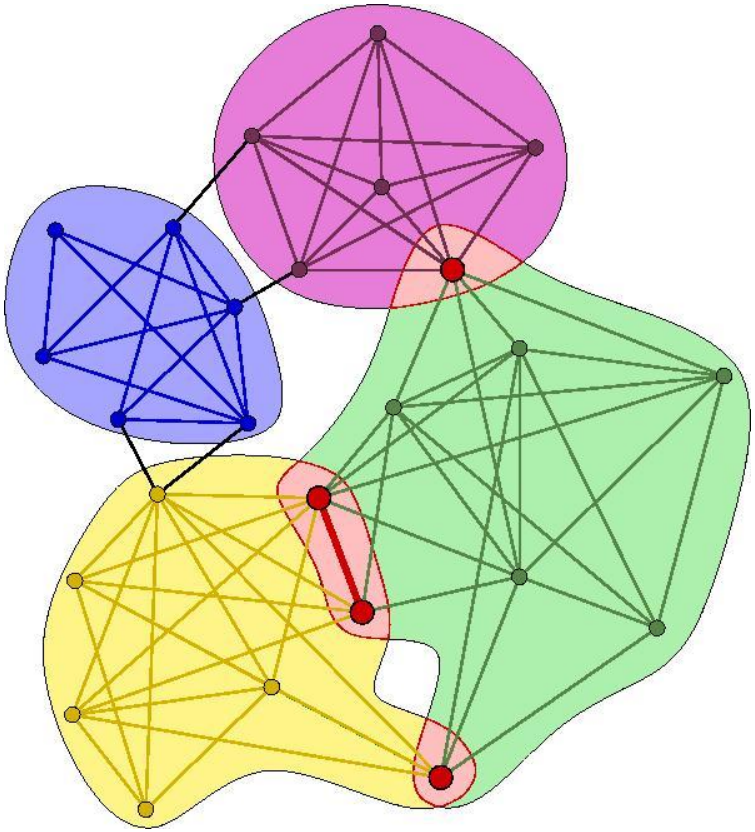
- A community can be considered as a summary of the whole network
- Easier to visualize and understand

Preserve Privacy

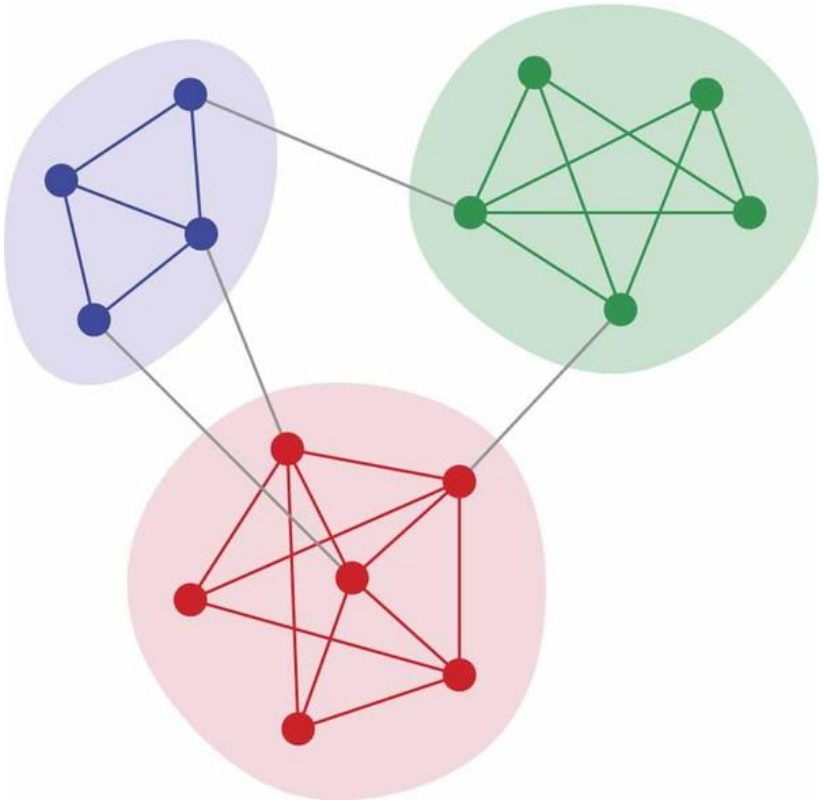
- [Sometimes] a community can reveal some properties without releasing the individuals' privacy information.



Overlapping vs. Disjoint Communities



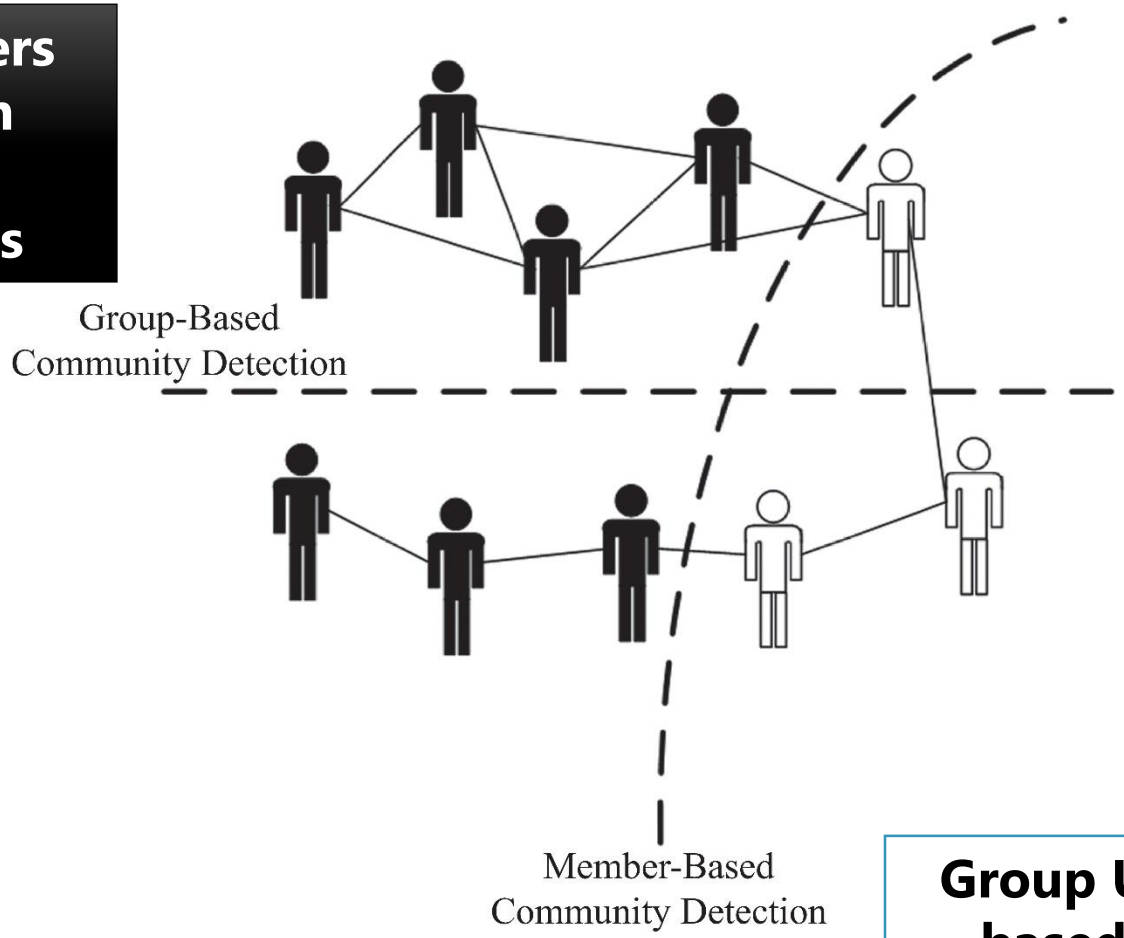
Overlapping Communities



Disjoint Communities

Community Detection Algorithms

Group Users based on Group attributes



Group Users based on Member attributes

Member-Based Community Detection

Member-Based Community Detection

- Look at nodes structural characteristics; and
- Identify nodes with similar characteristics and consider them a community

Node Characteristics

A. Degree

- Nodes with same (or similar) degrees are in one community
- Example: cliques

B. Reachability

- Nodes that are close (small shortest paths) are in one community
- Example: k-cliques

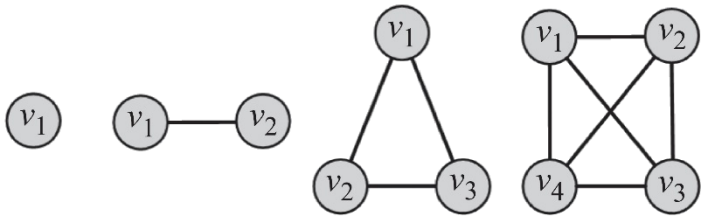
C. Similarity

- Two nodes are similar if they share the same set of neighbours

A. Node Degree

Most common subgraph searched for:

- **Clique**: a complete subgraph in which all nodes inside the subgraph are adjacent to each other



Find communities by searching for :

1. **All maximal cliques**: cliques that are not subgraphs of a larger clique; i.e., cannot be further expanded

To overcome this, we can

I. Brute Force

II. Use cliques as the core for larger communities

This is a NP-hard problem

I. Brute-Force Method

Can find all the maximal cliques in the graph

For each vertex v_x, we find the maximal clique that contains node v_x

1. Start with v_x, it forms the initial candidate clique
2. From there, try growing the clique by considering the neighbours
3. This might result in multiple cliques
4. Try growing each of them until cannot grow them and still remain a clique
5. These are the cliques that contains v_x, select the maximal size, and this will be the maximal clique containing v_x

Repeat for each vertex, and the union of all candidate maximal cliques are the set of maximal cliques of this graph.

Impractical for large networks:

- For a complete graph of only 100 nodes, the algorithm will generate at least $2^{99} - 1$ different cliques starting from any node in the graph

I. Brute-Force Method

Algorithm 1 Brute-Force Clique Identification

Require: Adjacency Matrix A, Vertex v_x

1: **return** Maximal Clique C containing v_x
2: CliqueStack = $\{\{v_x\}\}$, Processed = $\{\}$;
3: **while** CliqueStack not empty **do**
4: C=pop(CliqueStack); push(Processed,C);
5: v_{last} = Last node added to C;
6: $N(v_{last}) = \{v_i | A_{v_{last},v_i} = 1\}$.
7: **for all** $v_{temp} \in N(v_{last})$ **do**
8: **if** $C \bigcup \{v_{temp}\}$ is a clique **then**
9: push(CliqueStack, $C \bigcup \{v_{temp}\}$);
10: **end if**
11: **end for**
12: **end while**
13: Return the largest clique from Processed

Enhancing the Brute-Force Performance

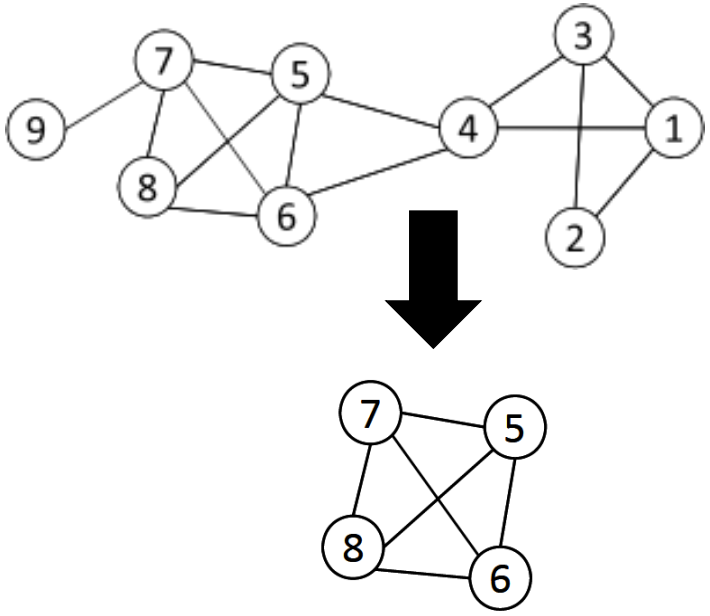
[Systematic] Pruning can help:

- When searching for cliques of size k or larger
- If the clique is found, each node should have a degree equal to or more than $k-1$
- We can first prune all nodes (and edges connected to them) with degrees less than $k-1$
  - More nodes will have degrees less than $k-1$
  - Prune them recursively
- For large k, many nodes are pruned as social media networks follow a power-law degree distribution

Maximum Clique: Pruning…

Example. to find a clique ≥ 4, remove all nodes with degree $\leq (4-1)-1=2$

- Remove nodes 2 and 9
- Remove nodes 1 and 3
- Remove node 4



Even with pruning, cliques are less desirable

- Cliques are rare
- A clique of 1000 nodes, has 999x1000/2 edges
- A single edge removal destroys the clique
- That is less than 0.0002% of the edges!

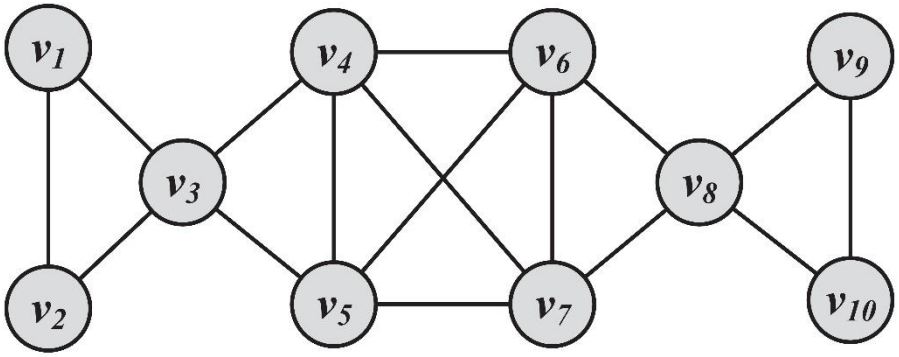
II. Using Cliques as a seed of a Community

Clique Percolation Method (CPM)

- Uses cliques as seeds to find larger communities
- CPM finds overlapping communities

- **Input**
  - A parameter k, and a network
- **Procedure**
  - Find out all cliques of size k in the given network
  - Construct a clique graph.
    - Two cliques are adjacent if they share $k-1$ nodes
  - Each connected components in the clique graph form a community

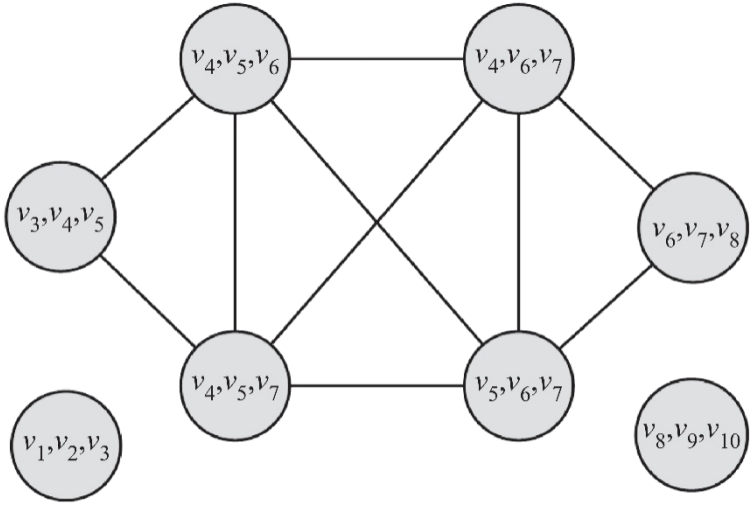
Clique Percolation Method: Example



(a) Graph

Cliques of size 3:

$\{v_1, v_2, v_3\}$, $\{v_3, v_4, v_5\}$,
$\{v_4, v_5, v_6\}$, $\{v_4, v_5, v_7\}$,
$\{v_4, v_6, v_7\}$, $\{v_5, v_6, v_7\}$,
$\{v_6, v_7, v_8\}$, $\{v_8, v_9, v_{10}\}$



(b) CPM Clique Graph

Communities:

$\{v_1, v_2, v_3\}$,
$\{v_8, v_9, v_{10}\}$,
$\{v_3, v_4, v_5, v_6, v_7, v_8\}$

Group-Based Community Detection

Group-Based Community Detection

Group-based community detection: finding communities that have certain **group properties**

Group Properties:

I. **Balanced:** Spectral clustering
II. **Robust:** k-connected graphs
III. **Modular:** Modularity Maximization
IV. **Dense:** Quasi-cliques
V. **Hierarchical:** Hierarchical clustering

III. Modular Communities

Consider a graph $G(V, E)$, where the degrees are known beforehand however edges are not

- Consider two vertices v_i and v_j with degrees d_i and d_j.

What is an expected number of edges between v_i and v_j?

- For any edge going out of v_i randomly the probability of this edge getting connected to vertex v_j is

$$\frac{d_j}{\sum_i d_i} = \frac{d_j}{2m}$$

- Probability of an edge between two vertices v_i and v_j

III. Modular Communities

- **Idea**: We assume that real-world networks should be far from random and possibly have communities in them.
  - Recall that a community is set of nodes with many edges between them
  - Therefore, if *we find a community that is very unlikely* because of the degree distributions of its nodes, then it is more likely that the community isn't formed by chance and is a valid/true one.

- Modularity defines this "unlikeliness" as a distance and modularity maximization tries to maximize this distance

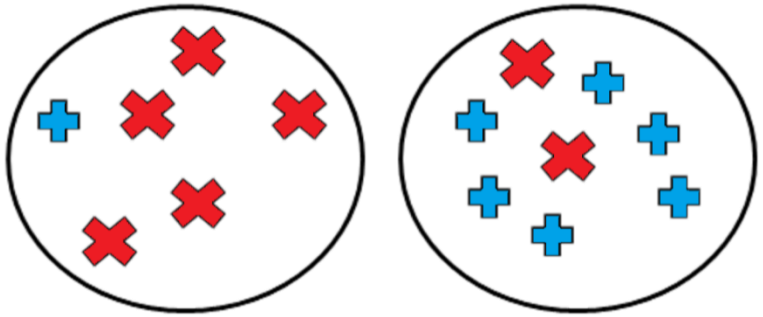
$$\sum_{v_i \text{ and } v_j \text{ in community}} A_{i,j} - (expected \# of \ edges \ between \ v_i \ and \ v_j)$$

Community Evaluation

Evaluating the Communities

We are given objects of two different kinds (+, ×)

- **The perfect community:** all objects inside the community are of the same type



- **Evaluation with ground truth**
- **Evaluation without ground truth**

Evaluation with Ground Truth

- When ground truth is available
  - We have partial knowledge of what communities should look like
  - We are given the correct community (clustering) assignments

- **Measures**
  - Precision and Recall, or F-Measure
  - Purity
  - Normalized Mutual Information (NMI)

Precision and Recall

$$Precision = \frac{Relevant\ and\ retrieved}{Retrieved}$$

$$Recall = \frac{Relevant\ and\ retrieved}{Relevant}$$

$$P = \frac{TP}{TP + FP}$$

$$R = \frac{TP}{TP + FN}$$

True Positive (TP) :

- When similar members are assigned to the same communities
- A **correct** decision.

False Negative (FN) :

- When similar members are assigned to different communities
- An **incorrect** decision

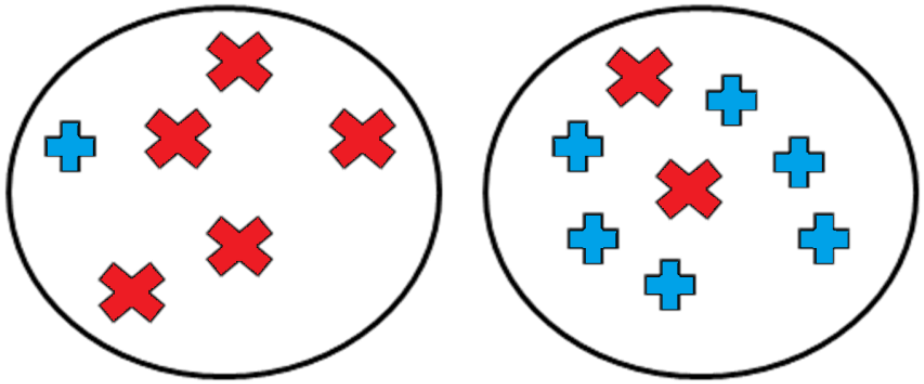
True Negative (TN) :

- When dissimilar members are assigned to different communities
- A **correct** decision

False Positive (FP) :

- When dissimilar members are assigned to the same communities
- An **incorrect** decision

Precision and Recall: Example



TP = $\binom{5}{2} + \binom{6}{2} + \binom{2}{2} = 26$

FP = (5x1) + (6x2) = 17

FN = (5x2) + (6x1) = 16

TN = (6x5) + (2x1) = 32

$$P = \frac{26}{26+17} = 0.60$$

$$R = \frac{26}{26+16} = 0.61$$

F-Measure

Either P or R measures one aspect of the performance,

- To integrate them into one measure, we can use the harmonic mean of precision and recall

$$F = 2 \cdot \frac{P \cdot R}{P + R}$$

For the example earlier,

$$F = 2 \times \frac{0.6 \times 0.61}{0.6 + 0.61} = 0.60$$

Purity

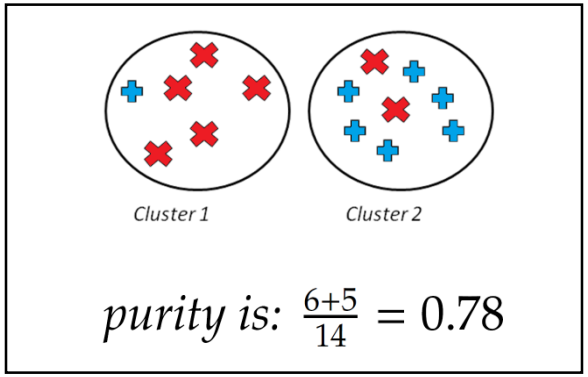
We can assume the majority of a community represents the community

- We use the label of the majority against the label of each member to evaluate the communities

Purity. The fraction of instances that have labels equal to the community's majority label

$$Purity = \frac{1}{N}\sum_{i=1}^{k}\max_{j}|C_i \cap L_j|$$

- k: the number of communities
- N: total number of nodes,
- L_j: the set of instances with label j in all communities
- C_i: the set of members in community i



Evaluation without Ground Truth

(a) U.S . Constitution

(b) Sports

- **Evaluation with Semantics**
  - A simple way of analyzing detected communities is to analyze other attributes (posts, profile information, content generated, etc.) of community members to see if there is a coherency among community members
  - The coherency is often checked via human subjects.
    - Or through labor markets: Amazon Mechanical Turk
  - To help analyze these communities, one can use word frequencies. By generating a list of frequent keywords for each community, human subjects determine whether these keywords represent a coherent topic.

- **Evaluation Using Clustering Quality Measures**
  - Use clustering quality measures (SSE)
  - Use more than two community detection algorithms and compare the results and pick the algorithm with better quality measure

Summary

- Community in social media
- Community detection
  - Node based (local): cliques, clique percolation
  - Group based (global): modularity, Lovain algorithm
- Community evaluation