Social cohesiveness

Equivalence

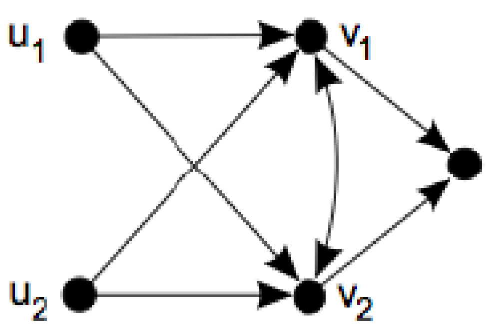
- Global, statistical properties of the networks:
  - average node degree (degree distribution)
  - average clustering
  - average path length
- Local, per vertex properties:
  - node centrality
  - page rank
- Pairwise properties:
  - node equivalence
  - node similarity
  - correlation between pairs of vertices (node values)

Structural equivalence

Definition

Structural equivalence: two vertices are structurally equivalent if their respective sets of in-neighbors and out-neighbors are the same

Interchangeable

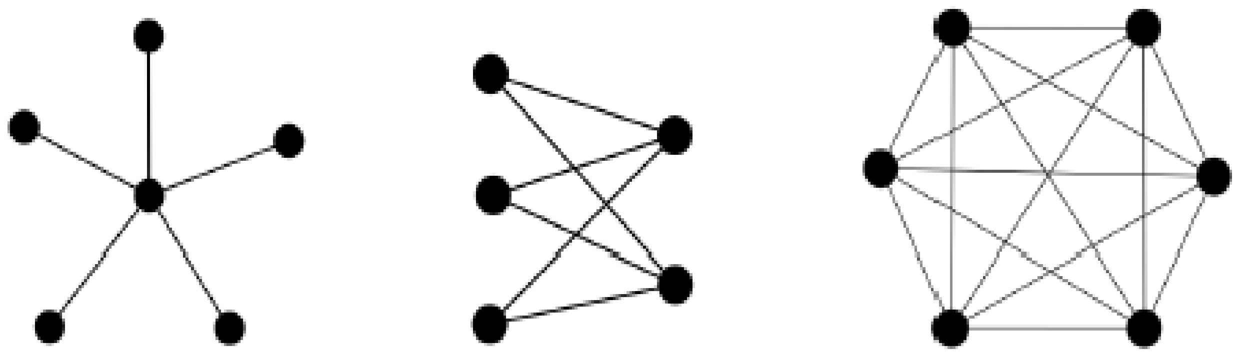


	u1	u2	v1	v2	w
u1	0	0	1	1	0
u2	0	0	1	1	0
v1	0	0	0	1	1
v2	0	0	1	0	1
w	0	0	0	0	0

rows and columns of adjacency matrix of structurally equivalent nodes are identical, "connect to the same neighbors"

Structural equivalence

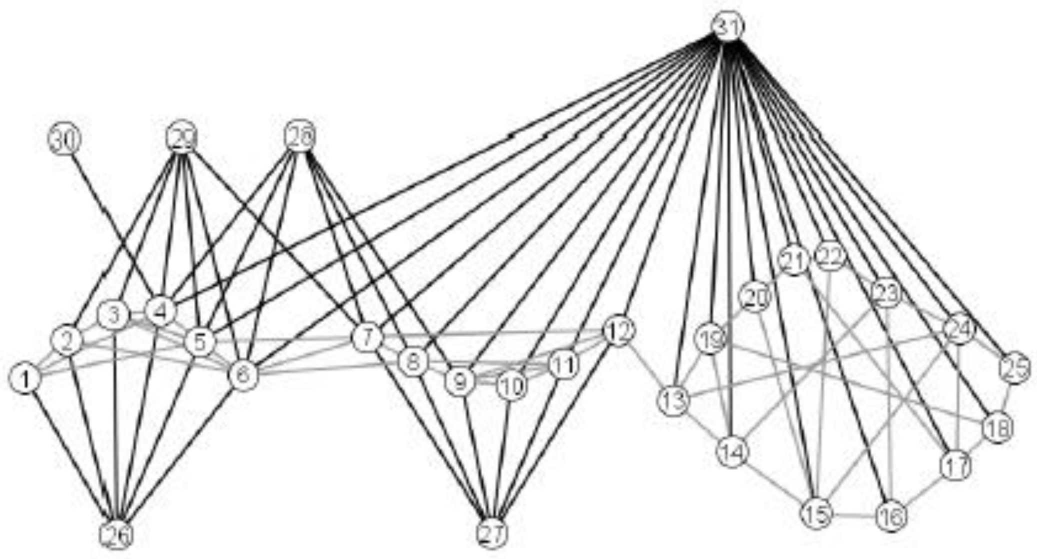
- In order for adjacent vertices to be structurally equivalent, then might have self loops.
- Sometimes called "strong structural equivalence"
- Sometimes relax requirements for self loops for adjacent nodes



Structural equivalence-similarity measures

- Jaccard similarity

$$J(v_i, v_j) = \frac{|\mathcal{N}(v_i) \cap \mathcal{N}(v_j)|}{|\mathcal{N}(v_i) \cup \mathcal{N}(v_j)|}$$



Structural equivalence-similarity measures

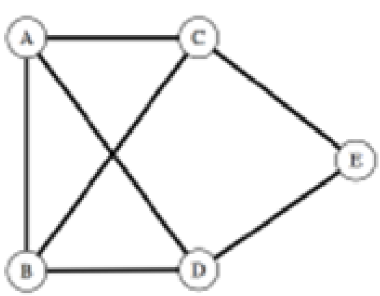
weighted

- Undirected graph
- Cosine similarity (vectors in n-dim space)

$$\sigma(v_i, v_j) = cos(\theta_{ij}) = \frac{\mathbf{v}_i \ \mathbf{v}_j^T}{|\mathbf{v}_i||\mathbf{v}_j|} = \frac{\sum_k A_{ik} A_{kj}}{\sqrt{\sum A_{ik}^2}\sqrt{\sum A_{jk}^2}}$$

- Pearson correlation coefficient:

$$r_{ij} = \frac{\sum_k (A_{ik} - \langle A_i \rangle)(A_{jk} - \langle A_j \rangle)}{\sqrt{\sum_k (A_{ik} - \langle A_i \rangle)^2}\sqrt{\sum_k (A_{jk} - \langle A_j \rangle)^2}}$$



$$\begin{matrix} 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 \end{matrix}$$

Structural equivalence-similarity measures

- Unweighted undirected graph $A_{ik} = A_{ki}$, binary matrix, only 0 and 1
- $k_i = \sum_k A_{ik} = \sum_k A_{ik}^2$ - node degree
- $n_{ij} = \sum_k A_{ik} A_{kj} = (A^2)_{ij}$ - number of shared neighbors
- $\langle A_i \rangle = \frac{1}{n} \sum_k A_{ik}$

- Cosine similarity (vectors in n-dim space)

$$\sigma(v_i, v_j) = cos(\theta_{ij}) = \frac{n_{ij}}{\sqrt{k_i k_j}}$$

Euclidian distance

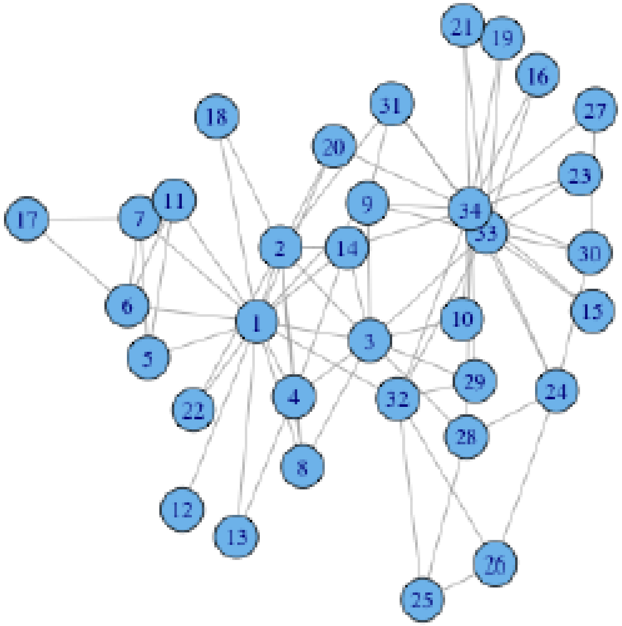
The Euclidean distance is defined as

$$d_{ij} = \sum_k (A_{ik} - A_{jk})^2$$

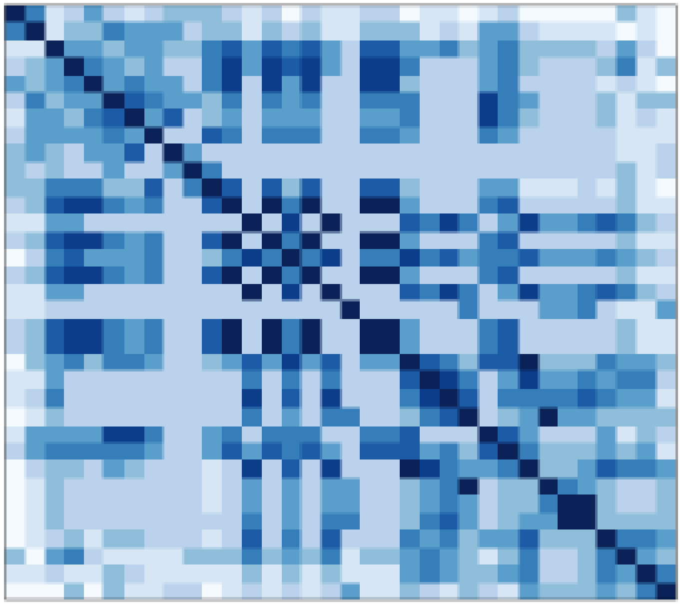
Normalize by maximum possible distance between the nodes

$$similarity = \frac{d_{ij}}{k_i + k_j}$$

Structural equivalence



Graph

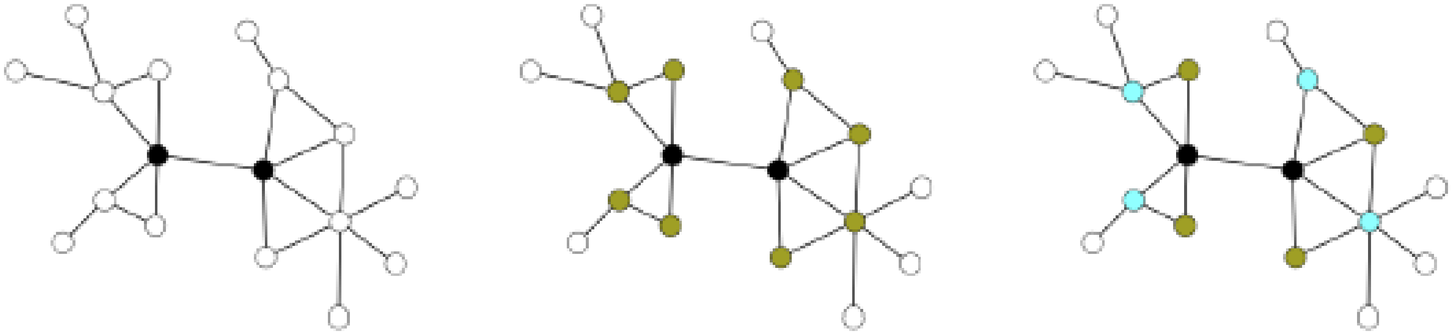


Node similarity matrix

Regular equivalence

Definition

Regular equivalence: two vertices are regularly equivalent if they are equally related to equivalent others.



- Equivalent definition in terms of role assignment (coloring): vertices that are colored the same, have the same colors of their neighborhoods

Regularly equivalent nodes are connected with regularly equivalent nodes

Regular equivalence

- Recursive definition: two vertices are regularly equivalent if they are equally related to equivalent others. Quantitative measure of regular equivalence - σ_{ij} , similarity score

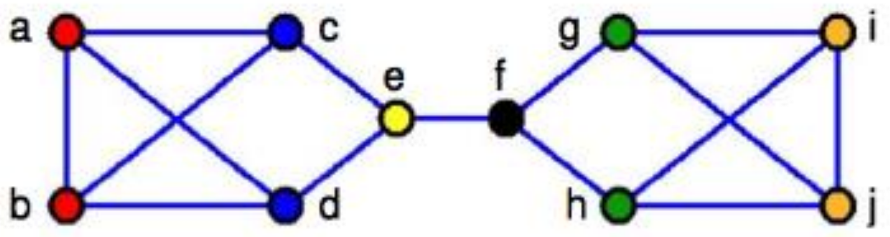
Matrix n*n $\longleftarrow$ $$\sigma_{ij} = \alpha \sum_{k,l} A_{ik} A_{jl} \sigma_{kl}$$

- should have high σ_{ii} - self similarity

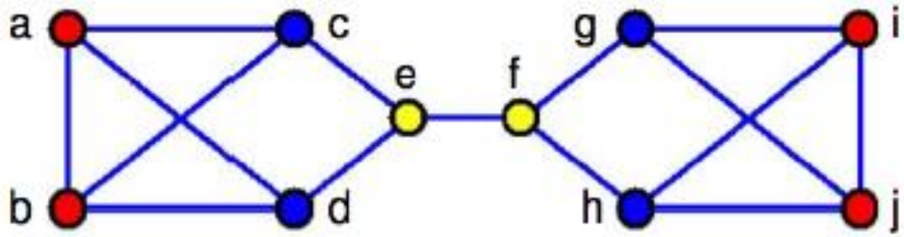
$$\sigma_{ij} = \alpha \sum_{k,l} A_{ik} A_{jl} \sigma_{kl} + \delta_{ij}$$

Regular equivalence

- structural equivalence



- regular equivalence



Regular equivalence

- Recursive definition: two vertices are regularly equivalent if they are equally related to equivalent others. Quantitative measure of regular equivalence - σ_{ij} , similarity score

Matrix n*n $\longleftarrow$ $\sigma_{ij} = \alpha \sum_{k,l} A_{ik} A_{jl} \sigma_{kl}$

- should have high σ_{ii} - self similarity

$$\sigma_{ij} = \alpha \sum_{k,l} A_{ik} A_{jl} \sigma_{kl} + \delta_{ij}$$

Regular equivalence (variation)

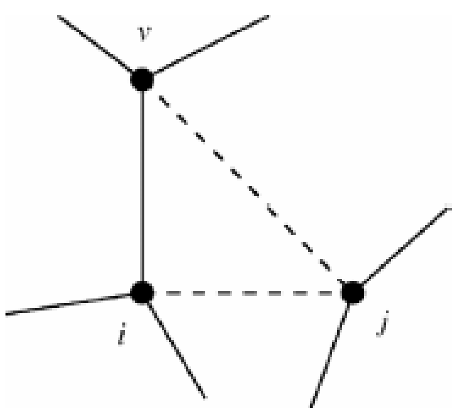
- A vertex j is similar to vertex i (dashed line) if i has a network neighbor v (solid line) that is itself similar to j

$$\sigma_{ij} = \alpha \sum_{v} A_{iv} \sigma_{vj} + \delta_{ij}$$

$$\boldsymbol{\sigma} = \alpha \mathbf{A} \boldsymbol{\sigma} + \mathbf{I}$$

- Closed form solution:

$$\boldsymbol{\sigma} = (\mathbf{I} - \alpha \mathbf{A})^{-1}$$



Automorphic Equivalence

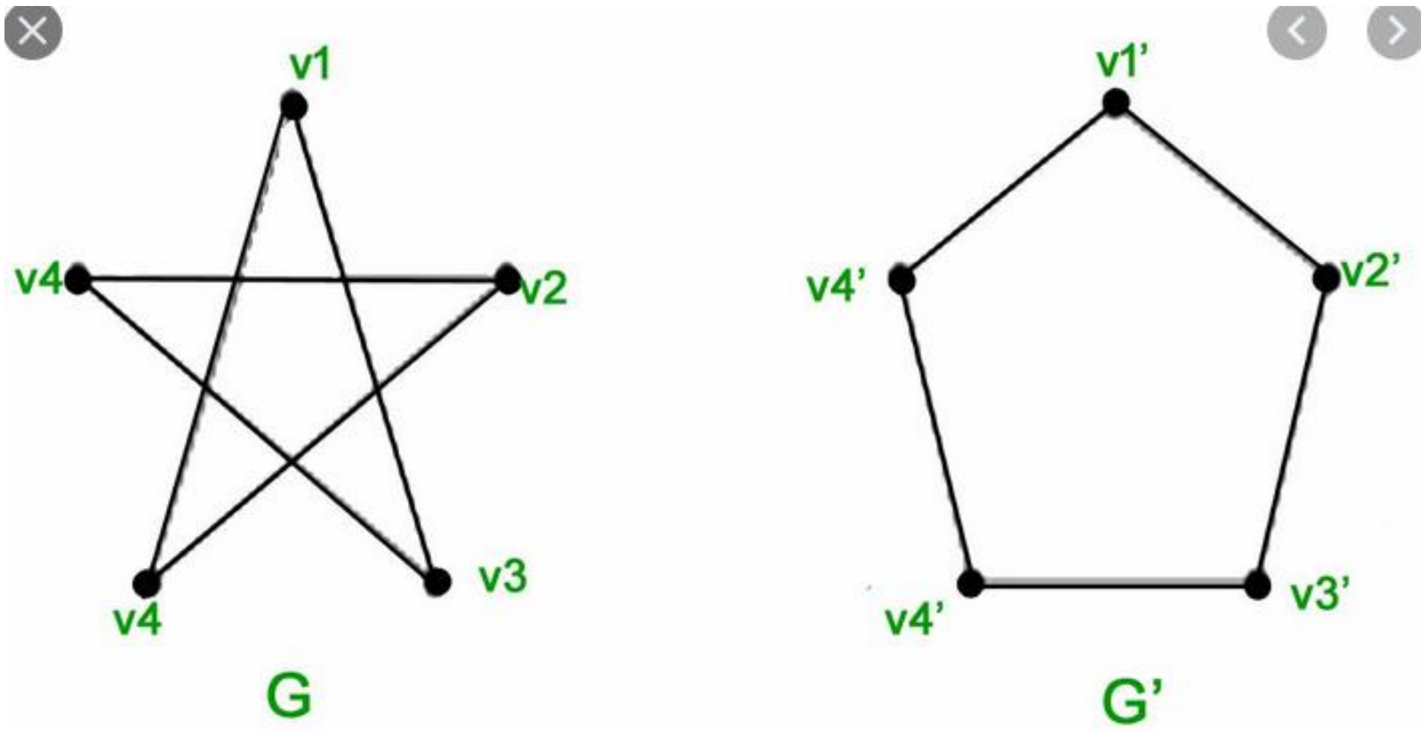
- Automorphic equivalence is not as demanding a definition of similarity as structural equivalence
- But is more demanding than regular equivalence

- Formally, two vertices u and v of a **labeled graph G** are automorphically equivalent

  If all the vertices can be re-labeled to form an isomorphic graph with the labels of u and v interchanged.

  Two automorphically equivalent vertices share exactly the same label-independent properties

Graph Isomorphism



Maintains the node adjacency property

Automorphic Equivalence

Suppose that we had 10 workers in the University Avenue McDonald's restaurant, who report to one manager. The manager, in turn, reports to a franchise owner. The franchise owner also controls the Park Street McDonald's restaurant. It too has a manager and 10 workers.

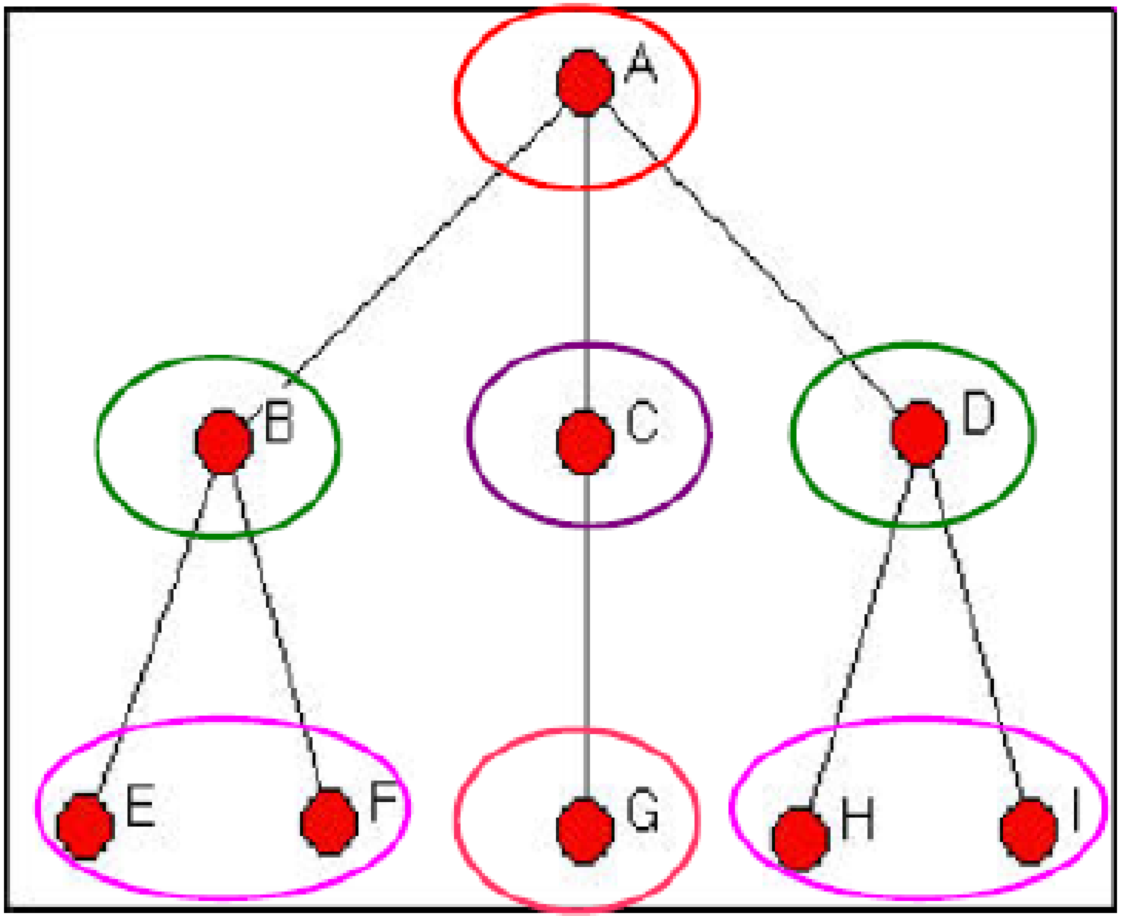
Now, if the owner decided to transfer the manager from University Avenue to the Park Street restaurant (and vice versa), the network has been disrupted.

But if the owner transfers both the managers and the workers to the other restaurant, all of the network relations remain intact.

Transferring both the workers and the managers is a permutation of the graph that leaves all of the distances among the pairs of actors exactly as it was before the transfer. In a sense, the "staff" of one restaurant is equivalent to the staff of the other, though the individual persons are not substitutable.

More intuitively, actors are automorphically equivalent if we can permute the graph in such a way that exchanging the two actors has no effect on the distances among all actors in the graph. If we want to assess whether two actors are automorphically equivalent, we first imagine exchanging their positions in the network. Then, we look and see if, by changing some other actors as well, we can create a graph in which all of the actors are the same distance that they were from one another in the original graph.

Automorphic Equivalence

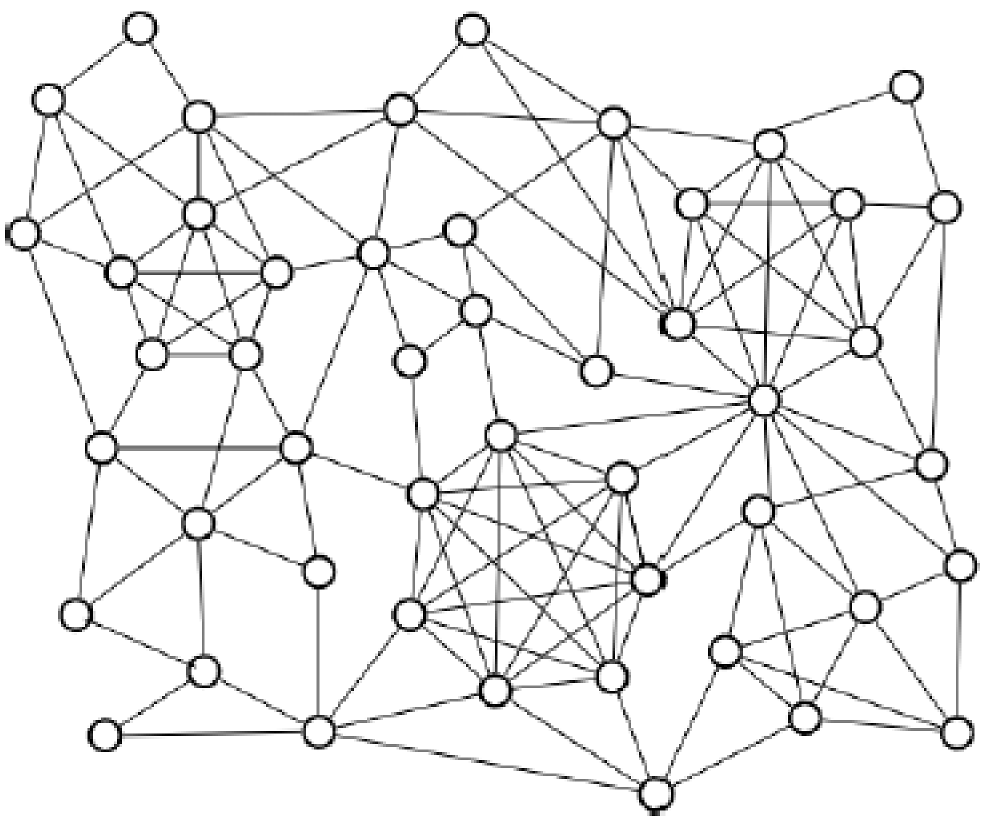


Social cohesiveness

Definition

A *clique* is a complete (fully connected) subgraph, i.e. a set of vertices where each pair of vertices is connected.

Social cohesiveness refers to the closeness of the members in the social network.

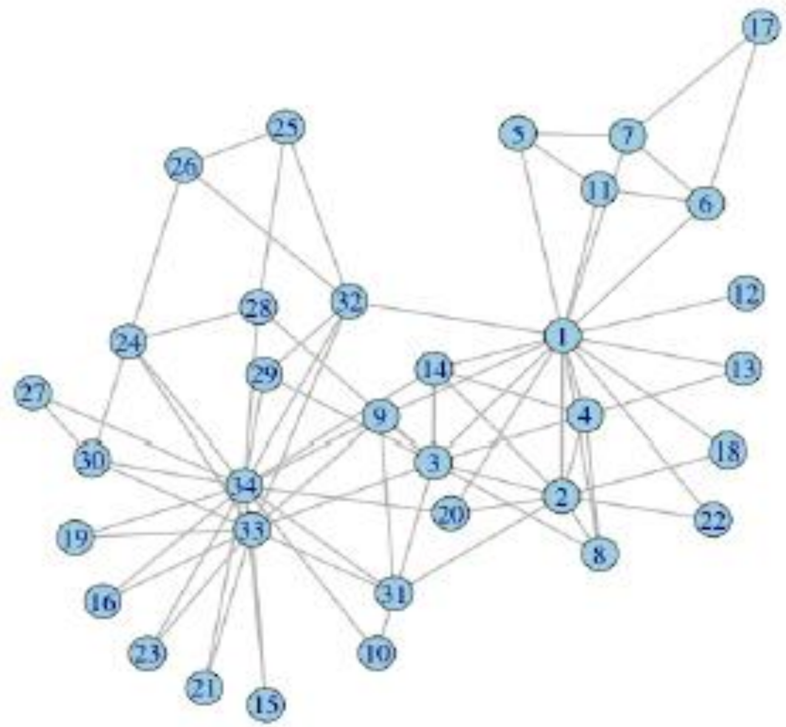


In graph theoretic terms, it refers to the "cliquishness" of a graph.

However, a complete clique is too strict to be practical and is rarely observed in social networks.

Graph cliques

Zachary Karate Club, 1977

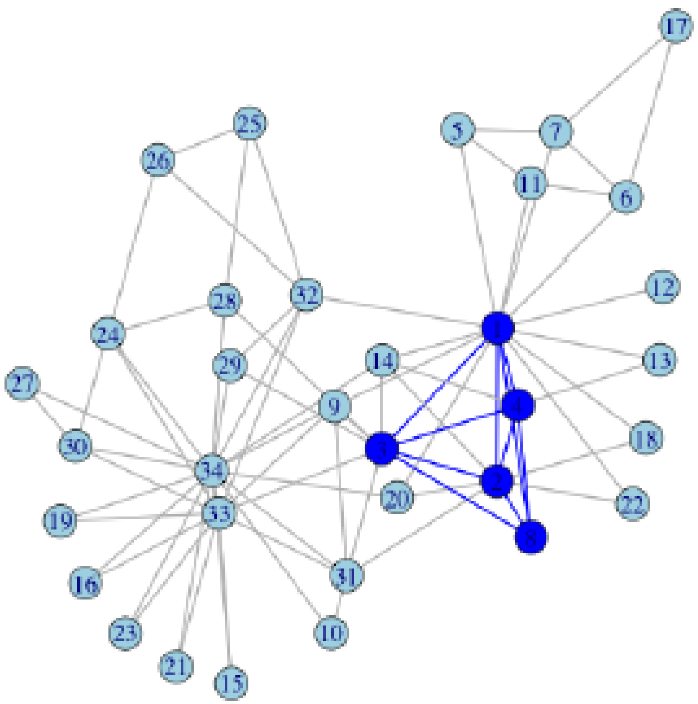


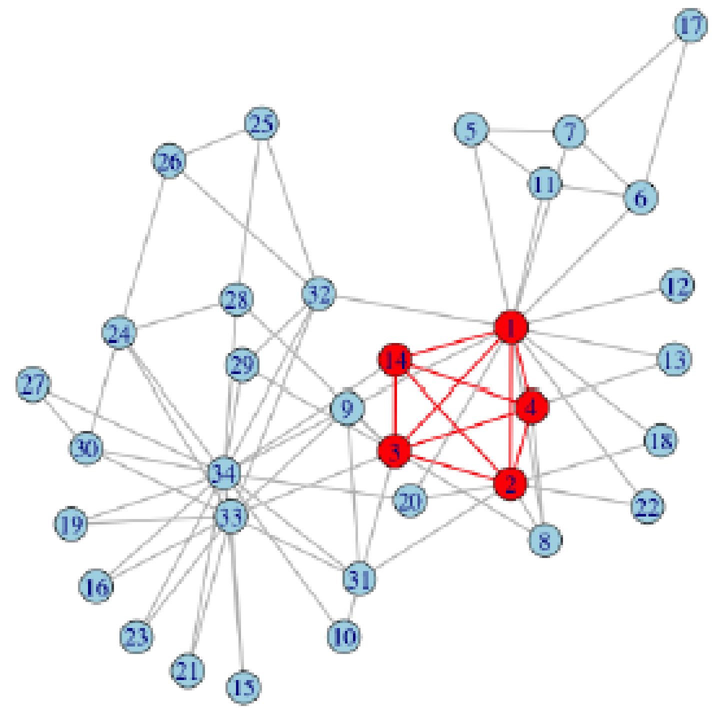
Maximal cliques:

Clique size:	3	4	5
Number of cliques:	21	2	2

Graph cliques

Zachary karate club 1,2,3,4 - cores



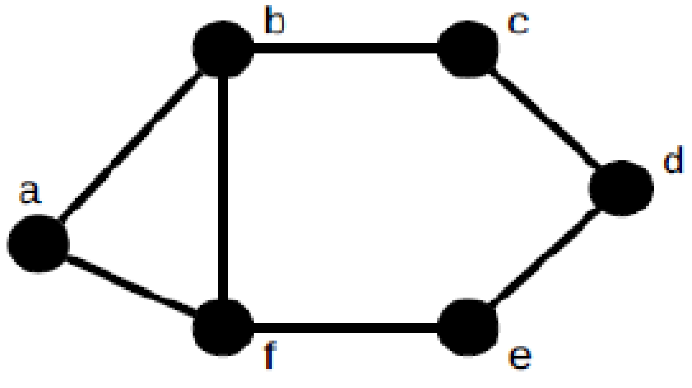


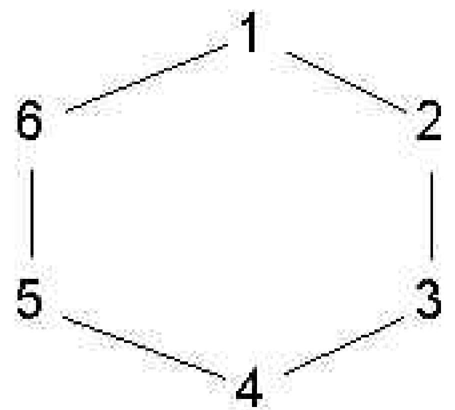
Maximum cliques

K-clique

A k-**clique** is a *maximal* set S of nodes in which the geodesic path between every pair of nodes $\{u, v\} \in S$ is less than or equal to k. As an examples, consider the network in Figure

In the graph {1,3,5} is a 2-clique, but none are connected to each other.

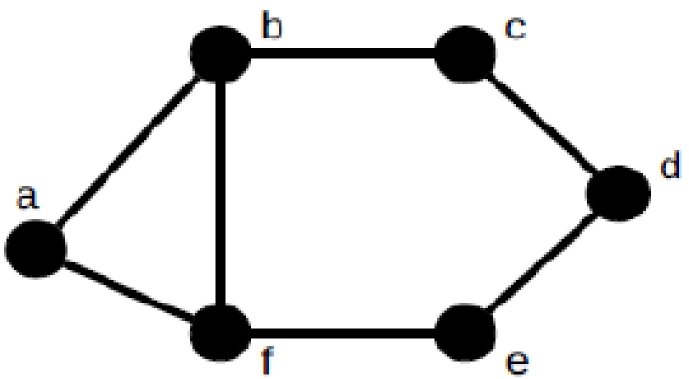




{a; b; c; f; e} forms a 2-clique, as the node d causes the distance between the nodes c and e to be 2, even though it is not a part of the 2-clique. Thus, k-cliques might not be as cohesive as they look. To resolve this issue, we consider k-clans.

K-Clan

A k-clan is a k-clique in which the subgraph induced by S has diameter less than or equal to k



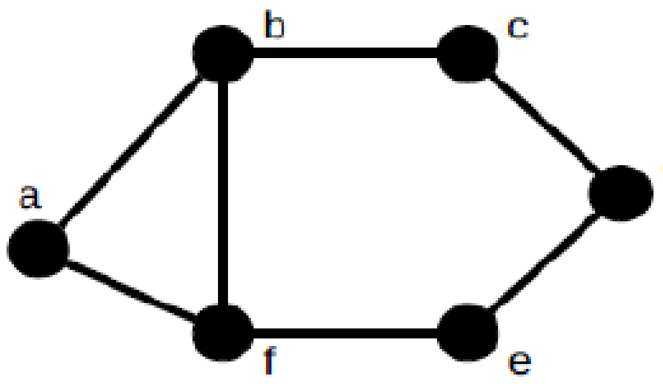
{b; e; f} also induces a subgraph that has diameter 2, but it does not form a 2-clan

Why?

{b; c; d; e; f} forms a 2-clan.

K-club

If we relax the maximality condition on k-clans, we get a k-club.



{b; e; f}, {a; b; f; e} form 2-club

2-cliques: {a,b,c,f,e}, {b,c,f,e,d};
2-clan {b,c,f,e,d};
2-clubs {a,b,f,c}, {a,b,f,e} and {b,c,f,e,d}

every n-clan is an n-club and every n-clan is an n-clique

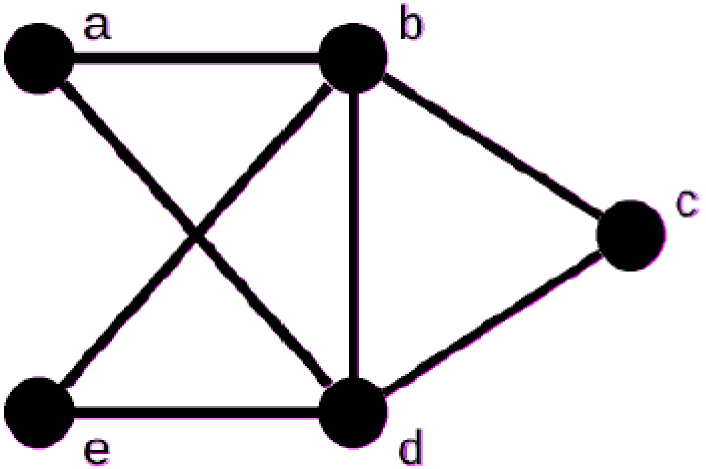
but every n-club is not an n-clan or n-clique, although it is contained in them (fail n-clique maximality condition)

K-plex

A k-plex is a maximal subset S of nodes such that every member of the set is connected to at least n - k other members, where n is the size of S.

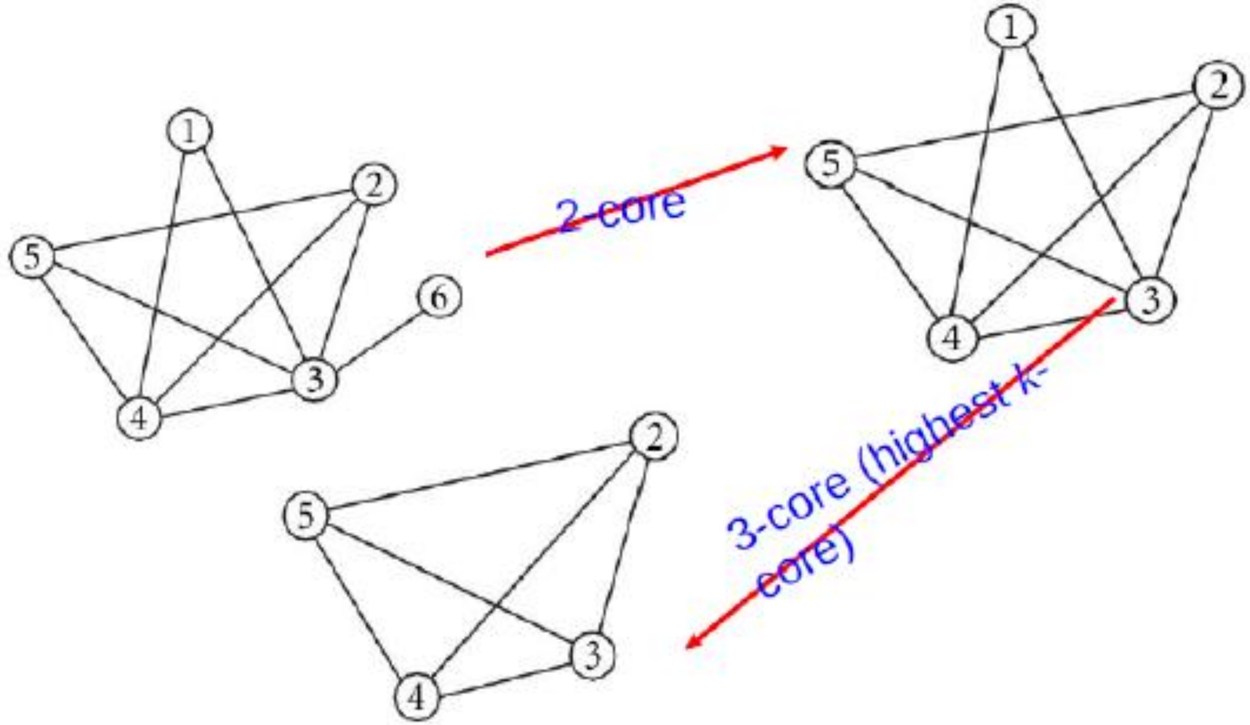
we see that {a; b; e; d} forms a 2-plex

when k = 1 ?



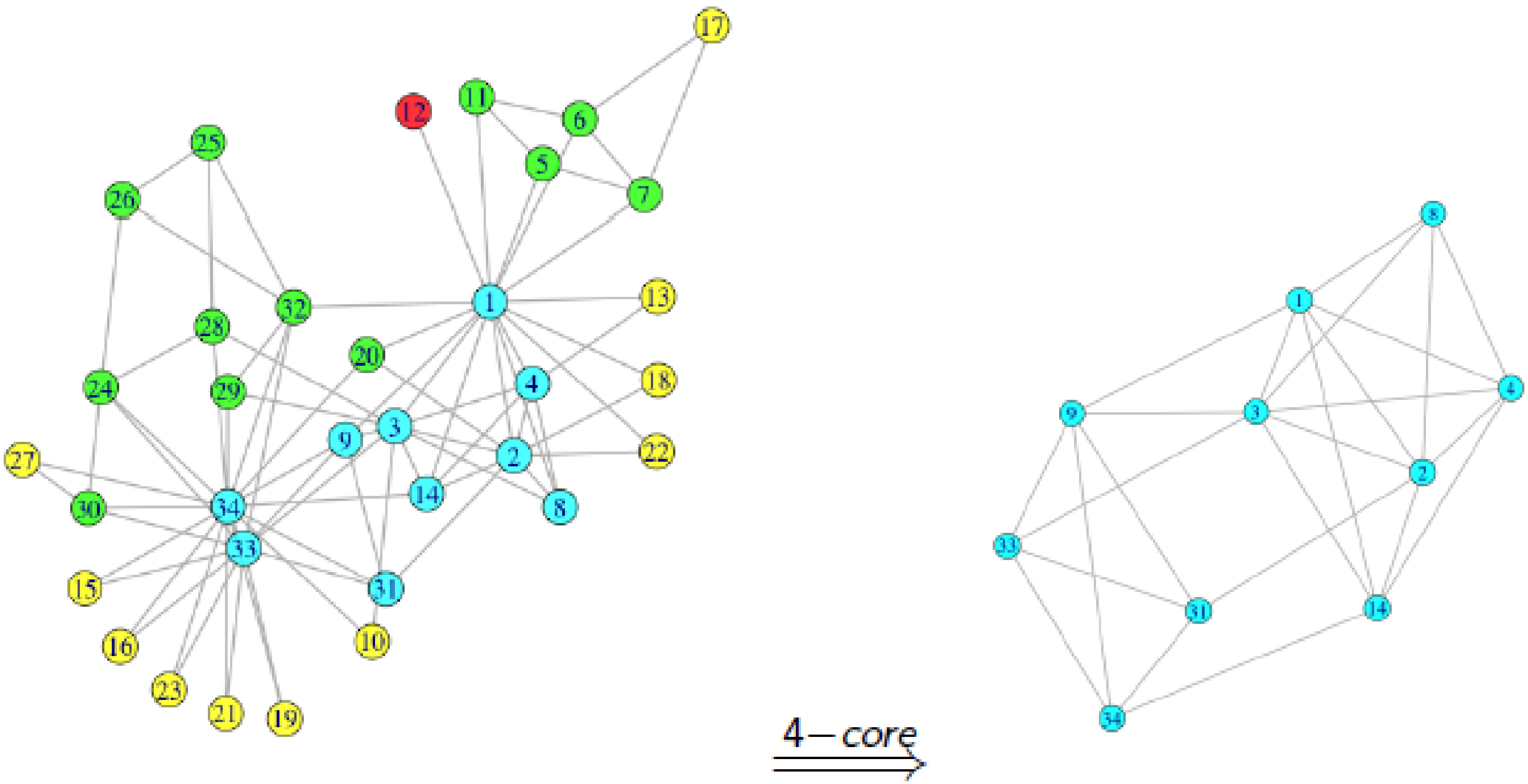
K-core

A k-core of a graph is a maximal subgraph such that each node in the subgraph has at least degree k



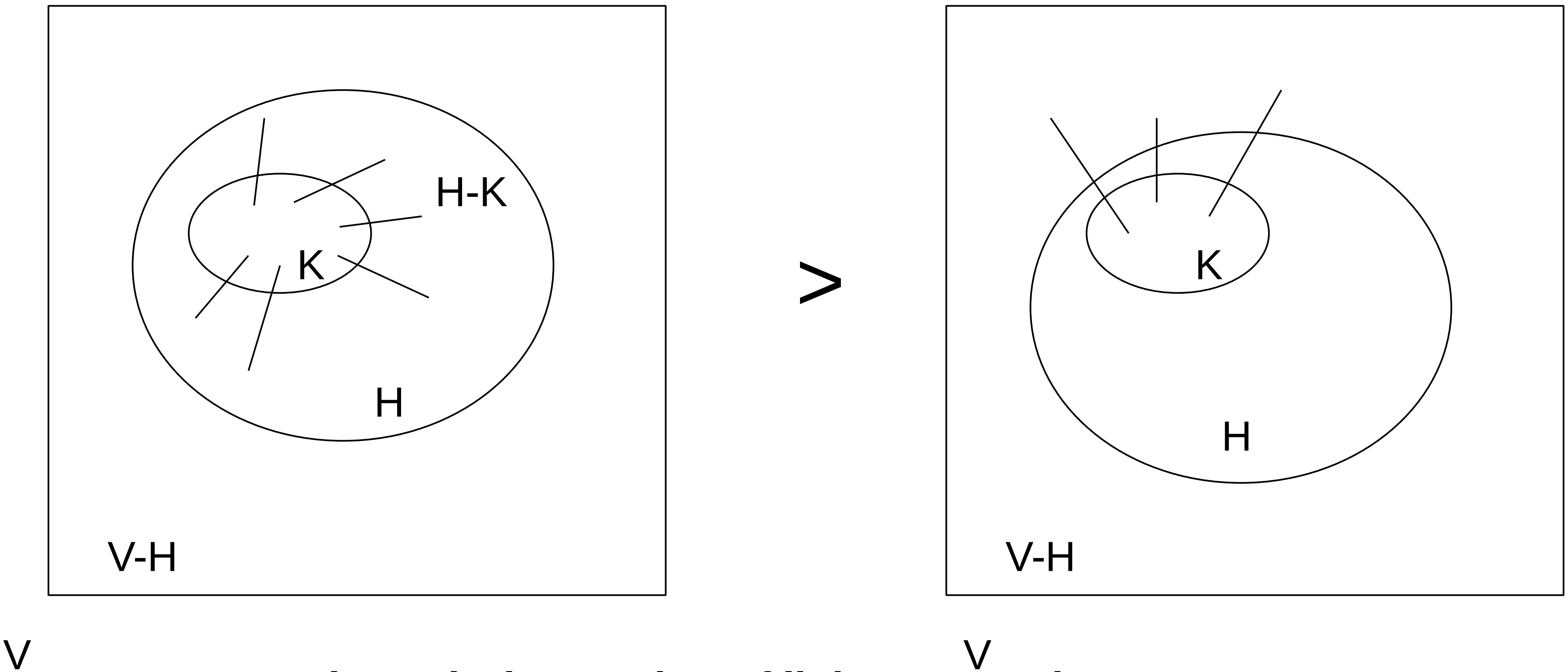
K-cores

Zachary karate club: 1,2,3,4 - cores



LS set

Proposition 1. Let $G(V, E)$ be a graph. A subset H of V is an LS set if and only if for any proper subset K of H, $\alpha(K, H-K) > \alpha(K, V-H)$.

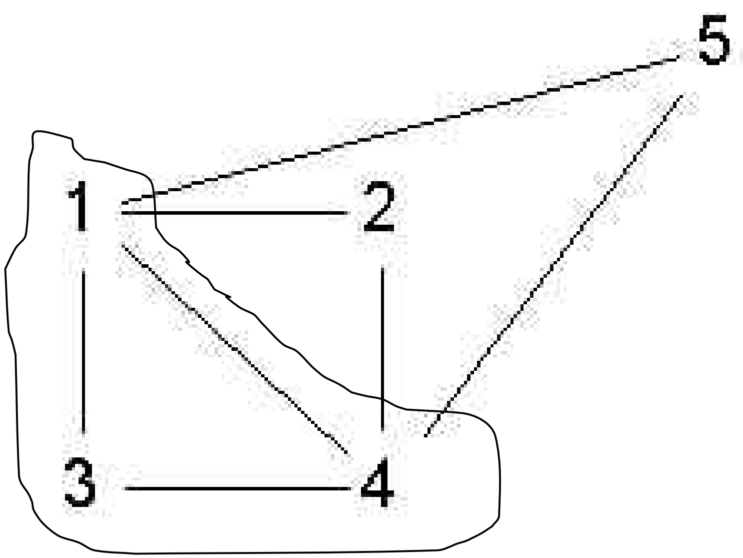


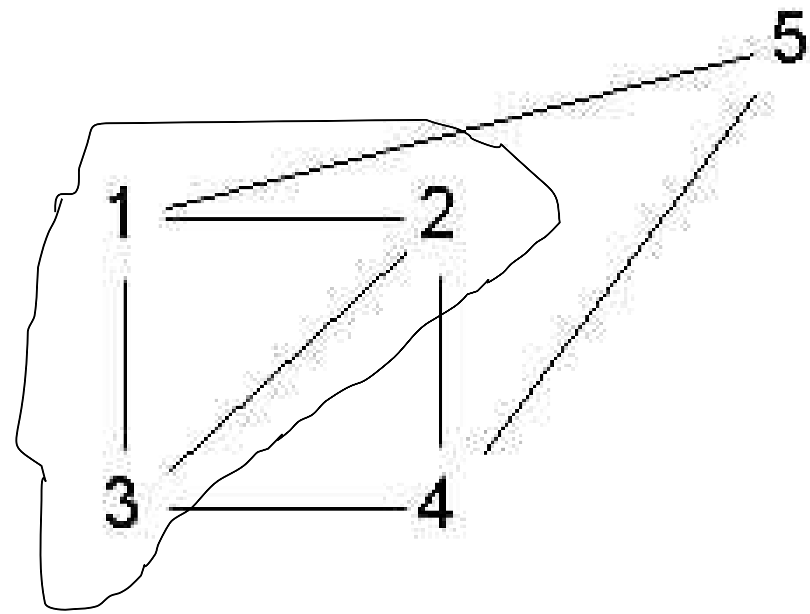
α(A,S-A)=the number of links connecting A, S-A

The essence of the idea is that an LS set may be thought of as the union of its subsets, and this union is “better” than any subset because it has fewer connections to the outside.

LS set

{1,2,3,4}





Lambda set

Definition 2. Given a graph $G(V, E)$, a *lambda set* S is a subset of V such that for all $a, b, c \in S$ and $d \in V - S$, $\lambda(a, b) > \lambda(c, d)$.

A lambda set is a maximal subset of nodes who have more edge-independent paths connecting them to each other than to outsiders.

Edge connectivity λ(a,b) of node pair a and b denote the minimum number of links that needs to be removed to disconnect a, b

For a subset S
k=λ(S)={min λ(a,b), $\forall$a,b in S}

Lambda-k set