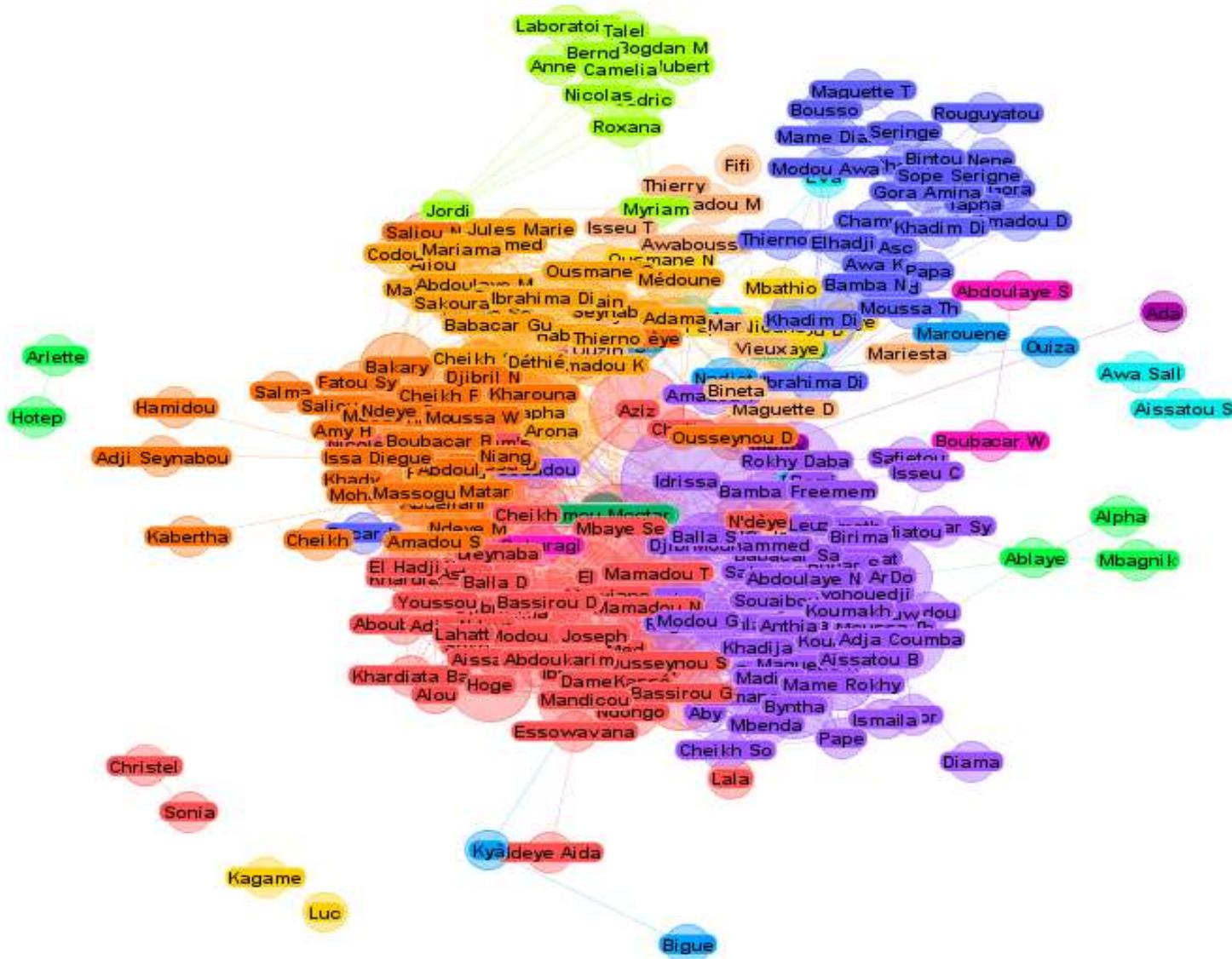


Les communautés: Détection et Evaluation

Dr Idrissa SARR

Communautés dans Facebook



Sommaire

- Pourquoi identifier des Communautés ?
- Définitions
- Approches de détection des Communautés

Pourquoi ?

- Facilite la compréhension les groupes cohérents au fil du temps
- Evaluer l'isolement des groupes
- Comprendre la formation et l'adoption des opinions



Les communautés dans la vie réelle

- 2 types de communautés (Groupes)
 - Groupes explicites : formés sur la base d'adhésion ou d'appartenance
 - Ex: Qui partage un département avec qui ?
 - Groupes implicites: formés implicitement par les interactions sociales
 - Qui échange fréquemment avec qui?
 - Les groupes peuvent changer dynamiquement puisque les interactions sociales changent?

Qu'est-ce qui fait une communauté ?

- mutualité des liens
 - Chaque nœud est en relation avec tous les autres
- fréquence des liens entre les membres (avec ou sans focus sur le temps)
 - Chaque membre du groupe a au moins k connexions avec les autres
- proximité ou accessibilité des membres du groupe
 - les individus sont séparés par au plus n sauts
- fréquence relative des liens entre les membres du sous-groupe par rapport aux non-membres
 - Unis contre le reste - peu d'échanges en dehors du groupe

Taxonomie des critères de détection

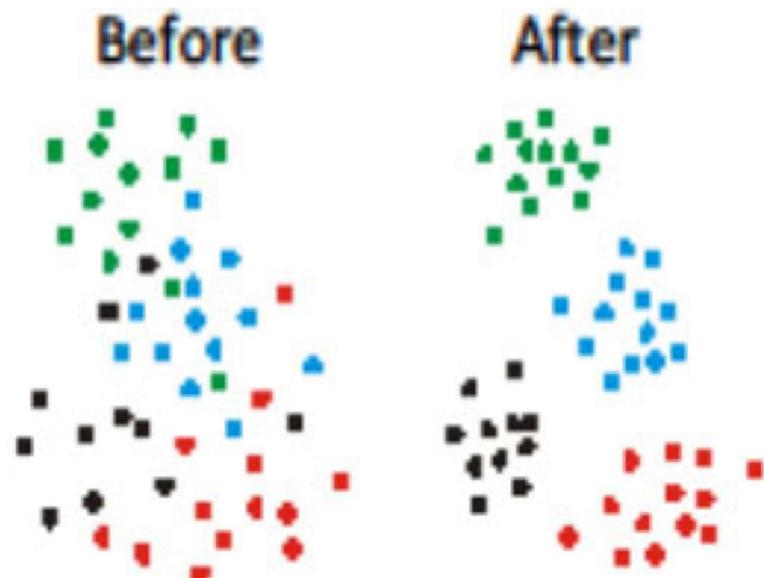
- 3 catégories (non disjointes) en fonction des objectifs
 - Communauté centrée sur les noeuds
 - Chaque noeud du groupe satisfait un nombre de propriétés
 - Communauté centrée sur le groupe
 - Le groupe dans sa globalité satisfait un nombre de propriétés (aucun accent sur les noeuds)
 - Communauté centrée sur le réseau
 - Diviser tout le réseau en plusieurs partitions

Une définition ...

- **Communauté**: “sous-ensembles d’acteurs reliés par des liens forts, directs, intenses et frequents.”
-- Wasserman and Faust, Social Network Analysis, Methods and Applications
- Par conséquent,
 - Une communauté est un ensemble d’acteurs interagissant de manière fréquente.
 - Un groupe de personne sans interactions n’est pas une communauté

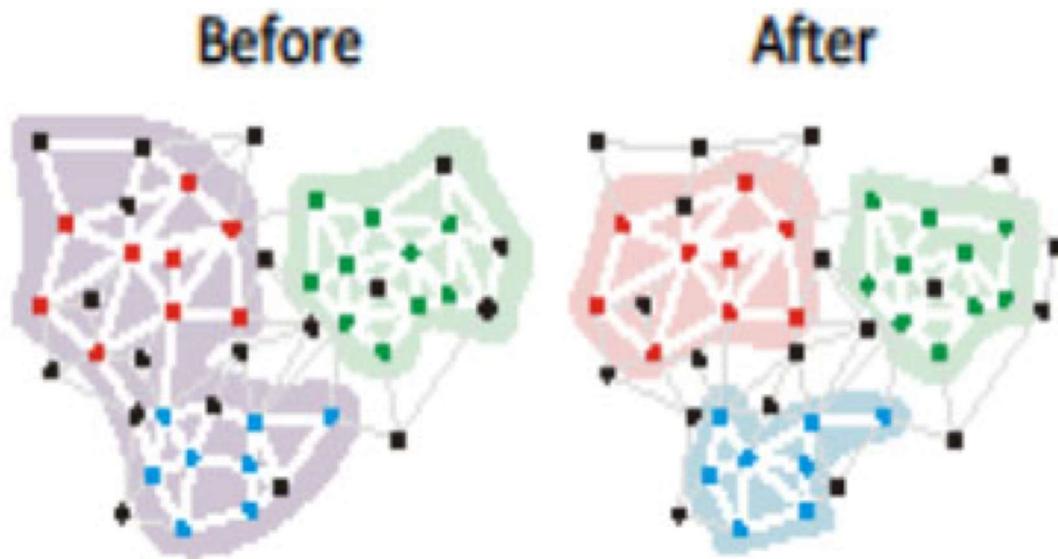
Différence entre Communauté au sens SNA vs clustering en Datamining

E.g. Crime Clustering



Increased cluster separation

Social Network Analysis



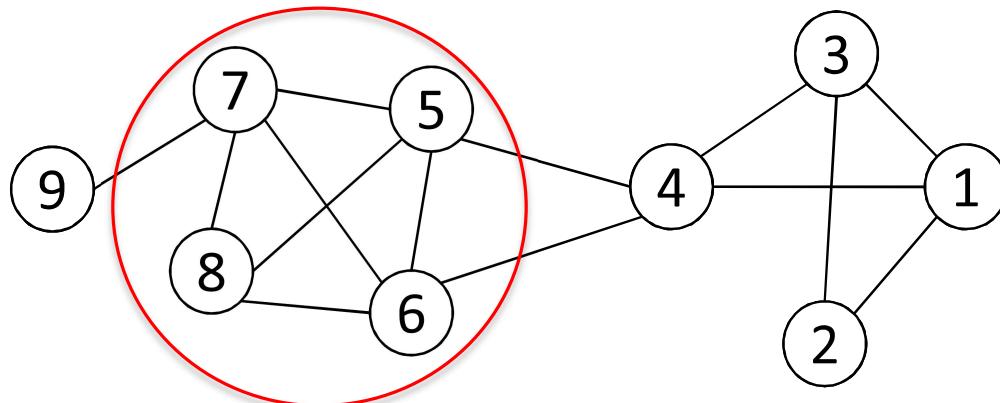
Better community finding

Détection de Communautés centrée sur les nœuds

- Les nœuds satisfont différentes propriétés
 - Mutualité complète: mutualité des liens
 - Tout le monde se connaît
 - cliques
 - proximité ou accessibilité des membres du sous-groupe
 - les individus sont séparés par au plus n sauts
 - k-clique, k-clan, k-club

Cliques

- **Clique**: un sous-graphe complet maximum dans lequel tous les nœuds sont adjacents



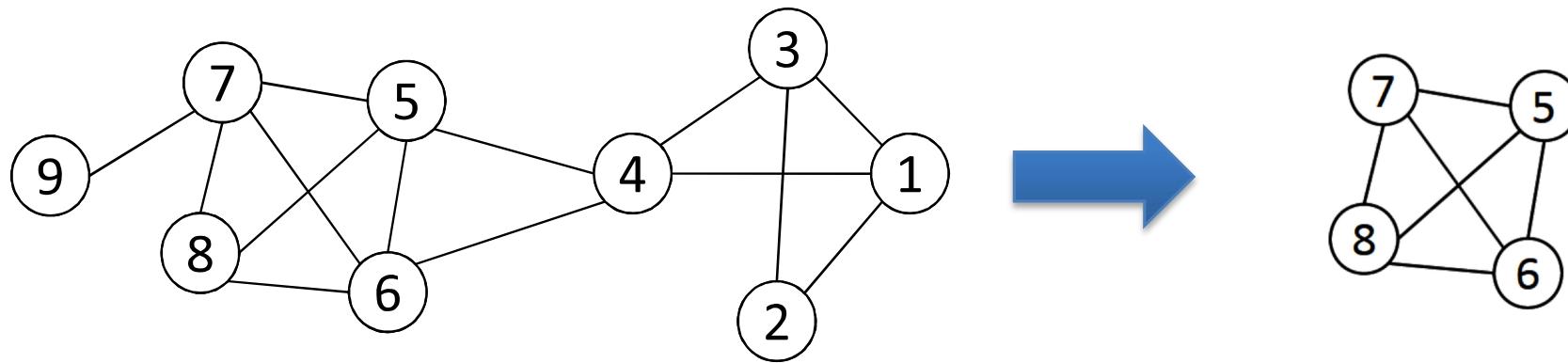
Noeuds 5, 6, 7 et 8 forment un clique

- Une mise en œuvre simple pour trouver des cliques est très coûteuse en temps
- Testons ça :
 - `Ggg <- sample_gnp(1000, 0.5) / ba <- barabasi.game(1000, power=2)`
 - `cliques(Ggg)`
 - `Largest_cliques(Ggg) / largest_cliques(as.undirected(l))`
 - `clique_num(gsy)`
- K-core : sous-graphe
 - `kc <- coreness(gsy, mode='all')`
 - `colbar <- rainbow(max(kc))`
 - `plot(gsy, vertex.size=15, vertex.color=colbar[kc]);`

Algorithme de détection de cliques maximales

- Dans une clique de taille k , chaque nœud conserve un degré $= k-1$
- Les nœuds de degré $< k-1$ ne seront pas inclus dans la clique maximale
- Algorithme
 - Supposons une clique de taille k afin de trouver une clique plus grande, supprimer tous les nœuds de degré $\leq k-1$.
 - Répétez jusqu'à ce que le réseau soit le plus petit possible
- Beaucoup de nœuds sont supprimés puisque beaucoup de médias sociaux sont de type power law

Exemple

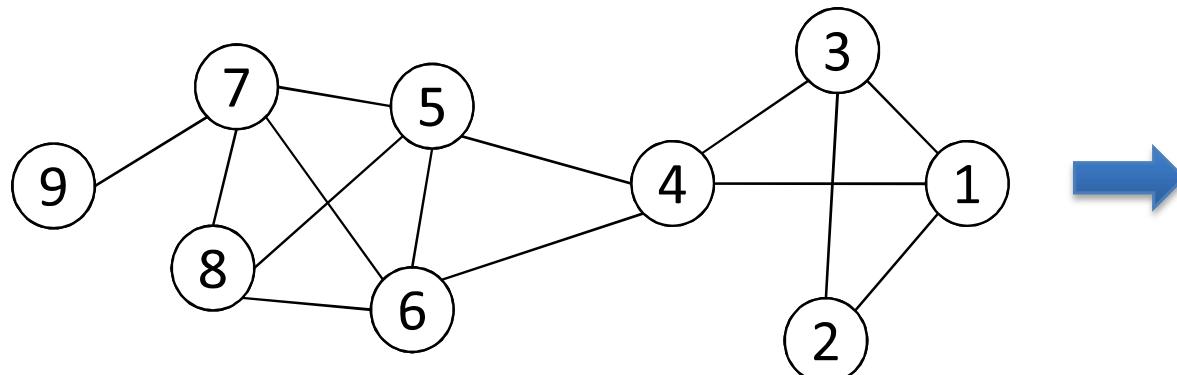


- Pour trouver une clique > 3 , supprimer tous les nœuds de degré $\leq 3 - 1 = 2$
 - Supprimer les nœuds 2 et 9
 - Supprimer les nœuds 1 et 3
 - Supprimer le noeud 4

Clique Percolation Method (CPM)

- Clique est une définition très stricte, instable
- À utiliser pour trouver des cliques comme noyau ou comme graine pour trouver de plus grandes communautés
- Méthode de détection de communautés chevauchantes.
- - **Données d'entrée**
 - k , et un réseau
 - **Procédure**
 - Trouver les cliques de taille k
 - Construire le graphe de cliques. 2 cliques sont adjacents si elles partagent $k-1$ nœuds
 - Les composants connectés du graphe forment une communauté

CPM : Exemple

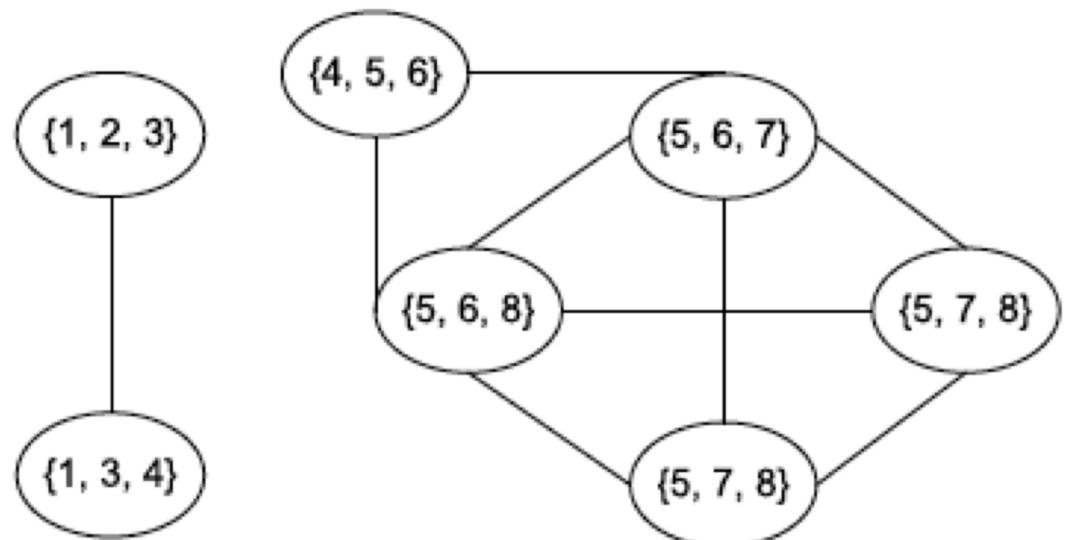


Cliques de taille 3:

$\{1, 2, 3\}, \{1, 3, 4\}, \{4, 5, 6\},$
 $\{5, 6, 7\}, \{5, 6, 8\}, \{5, 7, 8\},$
 $\{6, 7, 8\}$

Communautés:

$\{1, 2, 3, 4\}$
 $\{4, 5, 6, 7, 8\}$



Détection de communauté base sur le groupe: Density-Based Groups

- L'ensemble du groupe doit satisfaire des conditions
 - Exemple, la densité du groupe \geq à un seuil

$$\frac{|E_s|}{|V_s|(|V_s| - 1)/2} \geq \gamma$$

Proposer un algorithme pour détecter des communautés avec cette définition.

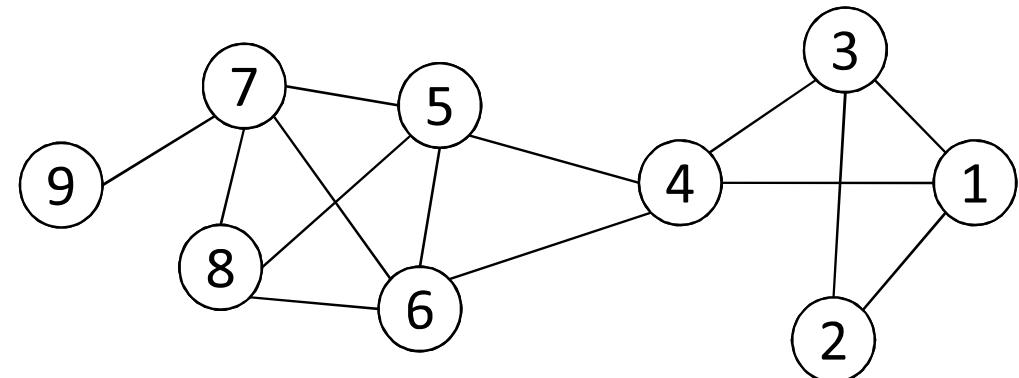
Détection centrée sur le réseau

- Le critère centré sur le réseau doit prendre en compte les connexions au sein d'un réseau global
- But : **partitionner le réseau en groups disjoints**
- Approches:
 - Similarité des liens
 - Maximisation de la modularité (fonction de qualité mesurant la qualité du partitionnement)
 - `cfg <- cluster_fast_greedy(as.undirected(gsy))`
 - `plot(cfg, as.undirected(gsy))`
 - `modularity(cfg)`

Clustering based on Vertex Similarity

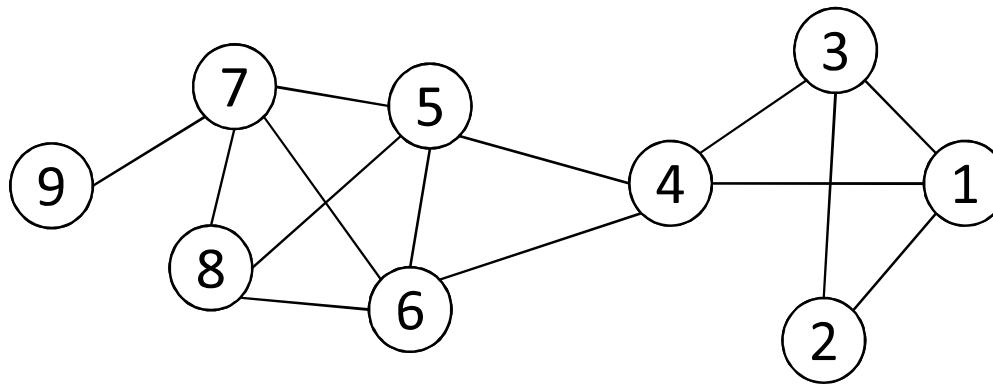
- Deux liens sont similaires s'ils partagent le même voisinage
 $\leftarrow \rightarrow$ Equivalence structurelle

Noeuds 1 et 3 sont structurellement équivalents;
Pareil pour 5 et 7.



Similarité des liens

- Similarité de Jaccard $J(A, B) = \frac{|A \cap B|}{|A \cup B|}$



$$Jaccard(4, 6) = \frac{|\{5\}|}{|\{1, 3, 4, 5, 6, 7, 8\}|} = \frac{1}{7}$$

Détection de communauté hiérarchique

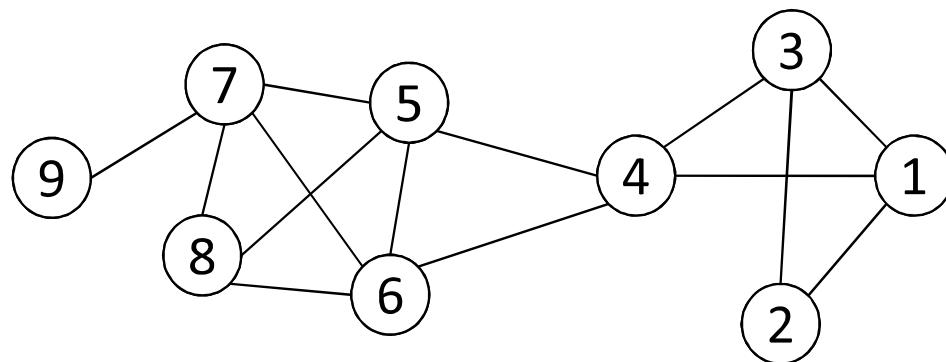
- But : construire une structure hiérarchique de communautés basée sur la topologie du réseau
- Permet une analyse à plusieurs niveaux
- 2 approches :
 - Divisive ou top-down
 - Agglomérative ou bottom-up

Approche divisive

- Partitionner les noeuds en plusieurs groupes
- Chaque groupe est divisé ensuite
- Exemple de partitionnement : supprimer récursivement les “faibles” lien

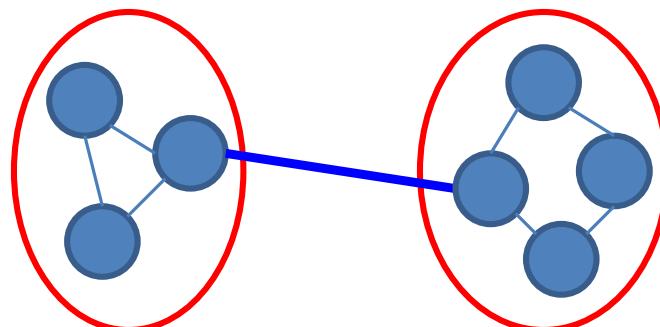
Edge Betweenness

- Le poids d'un lien peut être mesuré par **edge betweenness**

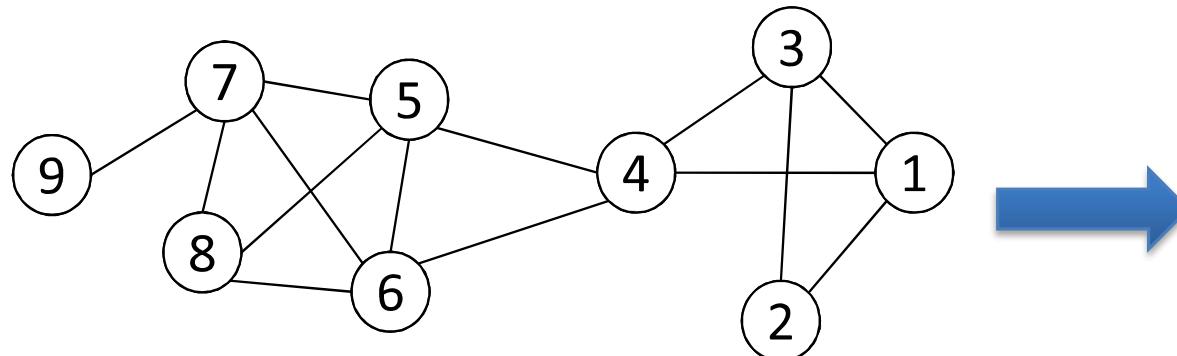


Le edge betweenness de $e(1, 2)$ est 4, puisque tous les courts chemins de 2 à {4, 5, 6, 7, 8, 9} passe par $e(1, 2)$ ou $e(2, 3)$, et $e(1,2)$ est le plus court chemin entre 1 et 2

- Les liens faibles jouent le rôle de ponts entre deux communautés.

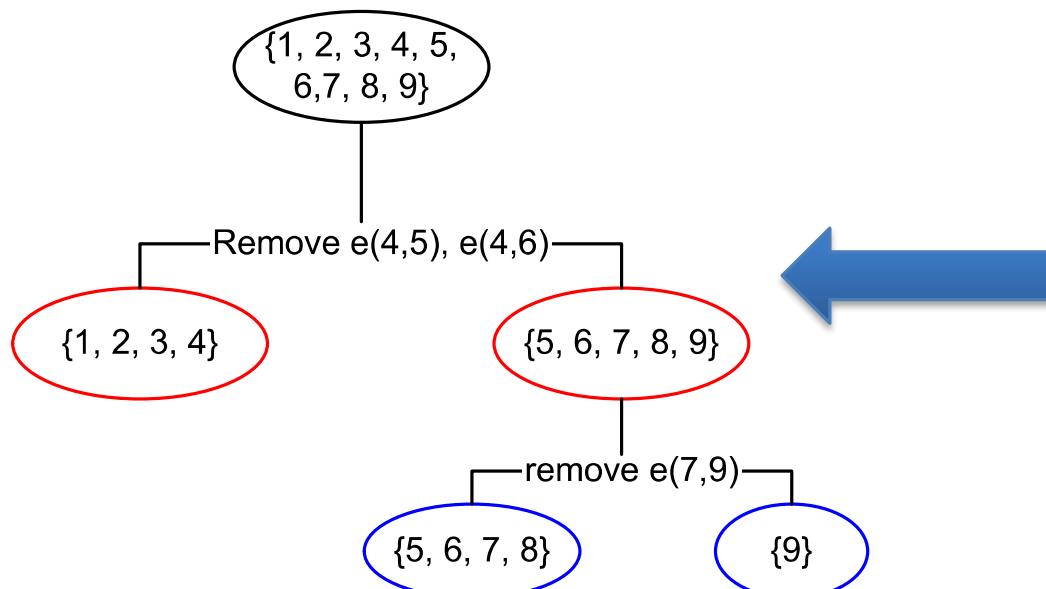


Approche divisive basée sur edge betweenness



Valeurs Initiales betweenness

		Table 3.3: Edge Betweenness								
		1	2	3	4	5	6	7	8	9
1	1	0	4	1	9	0	0	0	0	0
	2	4	0	4	0	0	0	0	0	0
3	1	4	0	9	0	0	0	0	0	0
4	9	0	9	0	10	10	0	0	0	0
5	0	0	0	10	0	1	6	3	0	0
6	0	0	0	10	1	0	6	3	0	0
7	0	0	0	0	6	6	0	2	8	0
8	0	0	0	0	3	3	2	0	0	0
9	0	0	0	0	0	0	8	0	0	0



Après suppression $e(4,5)$, le betweenness de $e(4,6)$ devient 20;

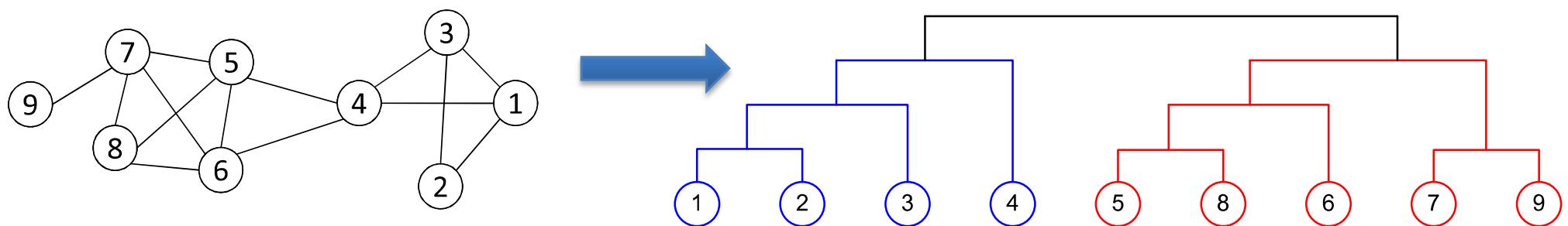
Après suppression de $e(4,6)$, le edge $e(7,9)$ a la plus grande valeur et doit être supprimé

Exemple avec Igraph

- `ceb <- cluster_edge_betweenness(gsy)`
- `dendPlot(ceb)`
- `plot(ceb, gsy)`

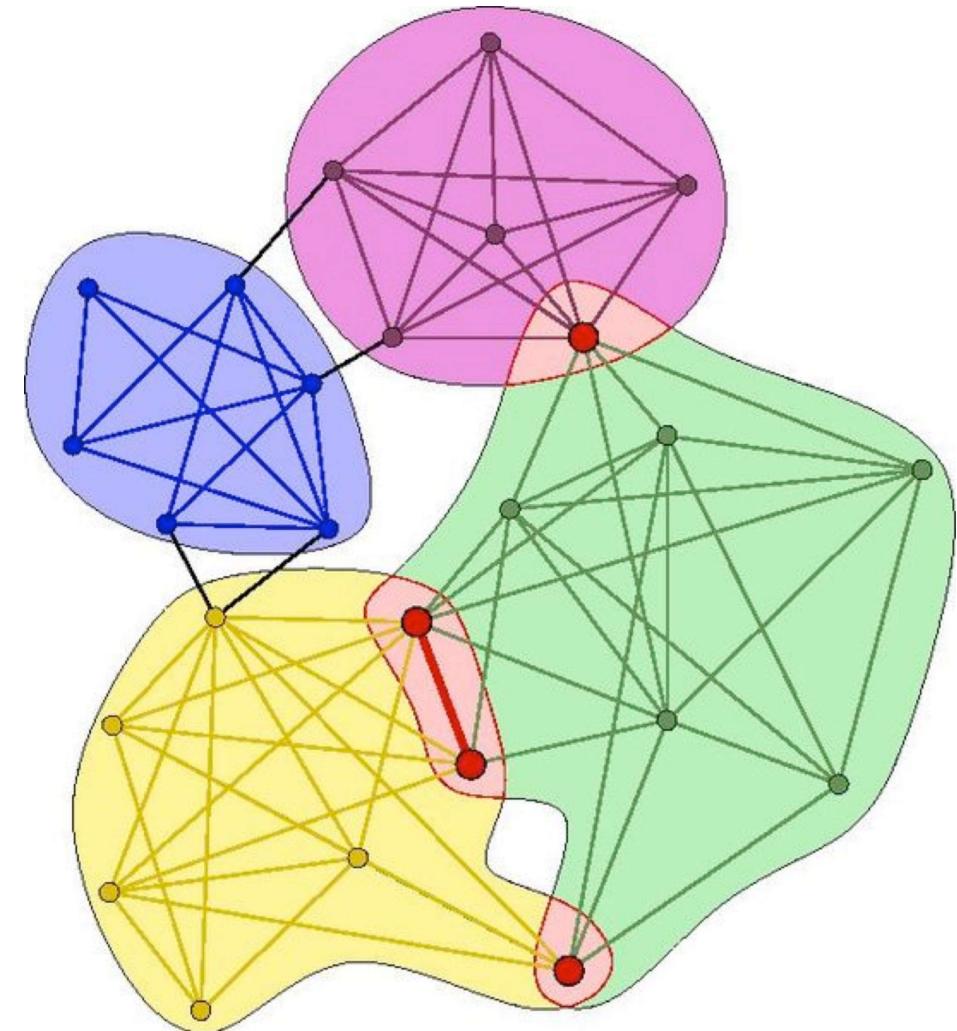
Détection agglomérative

- Initialiser chaque noeud comme une communauté
- Fusionner les communautés selon des critères
 - E.g., en fonction de la modularité ou similarité



Clique finder

- <http://cfinder.org>



Data Sets : <http://konect.unikoblenz.de/help/categories>