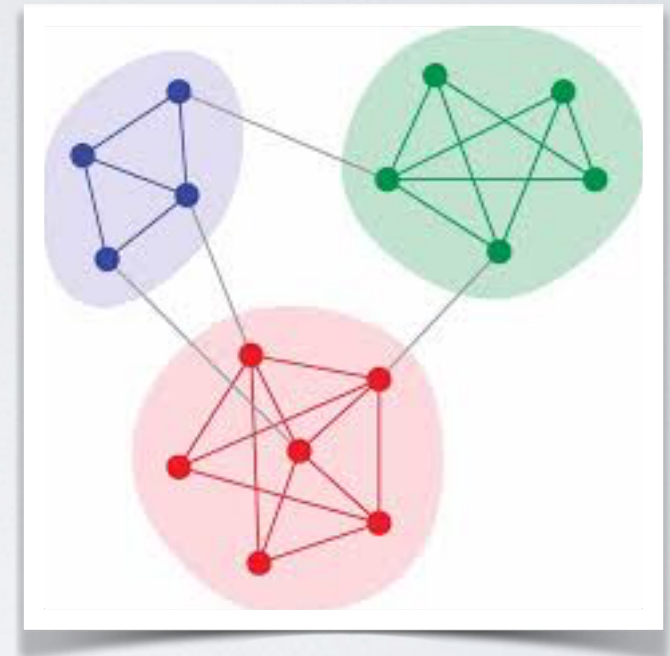


# DÉTECTION DE **COMMUNAUTÉS**

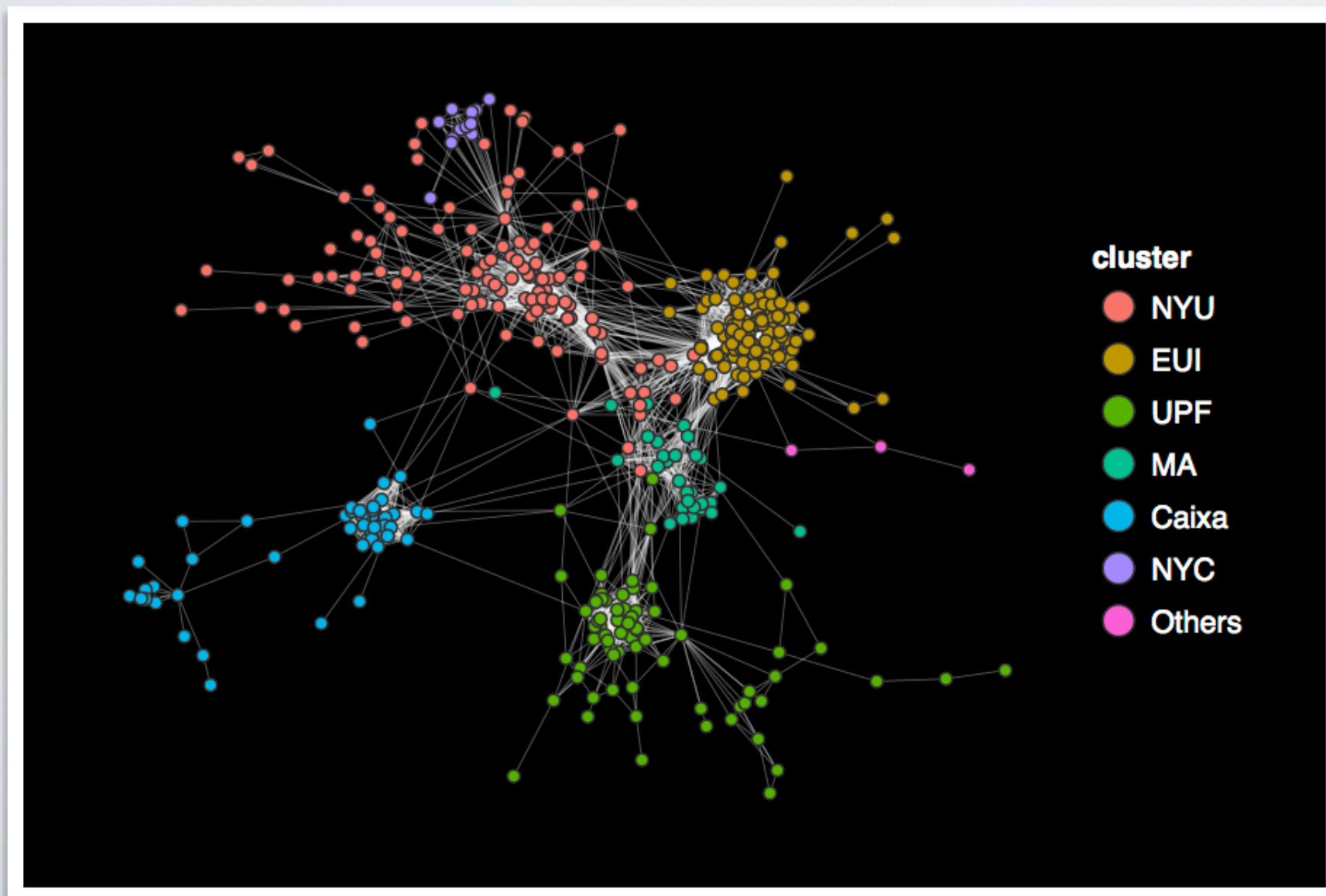
# COMMUNAUTÉS

- Détection de communautés
  - Découvrir des groupes de nœuds:
    - Fortement connectés entre eux
    - Faiblement connecté au reste du réseau
  - Pas de définition mathématique universelle
    - Plusieurs définitions imparfaites existe



# COMMUNAUTÉS DANS DES GRAPHS RÉELS

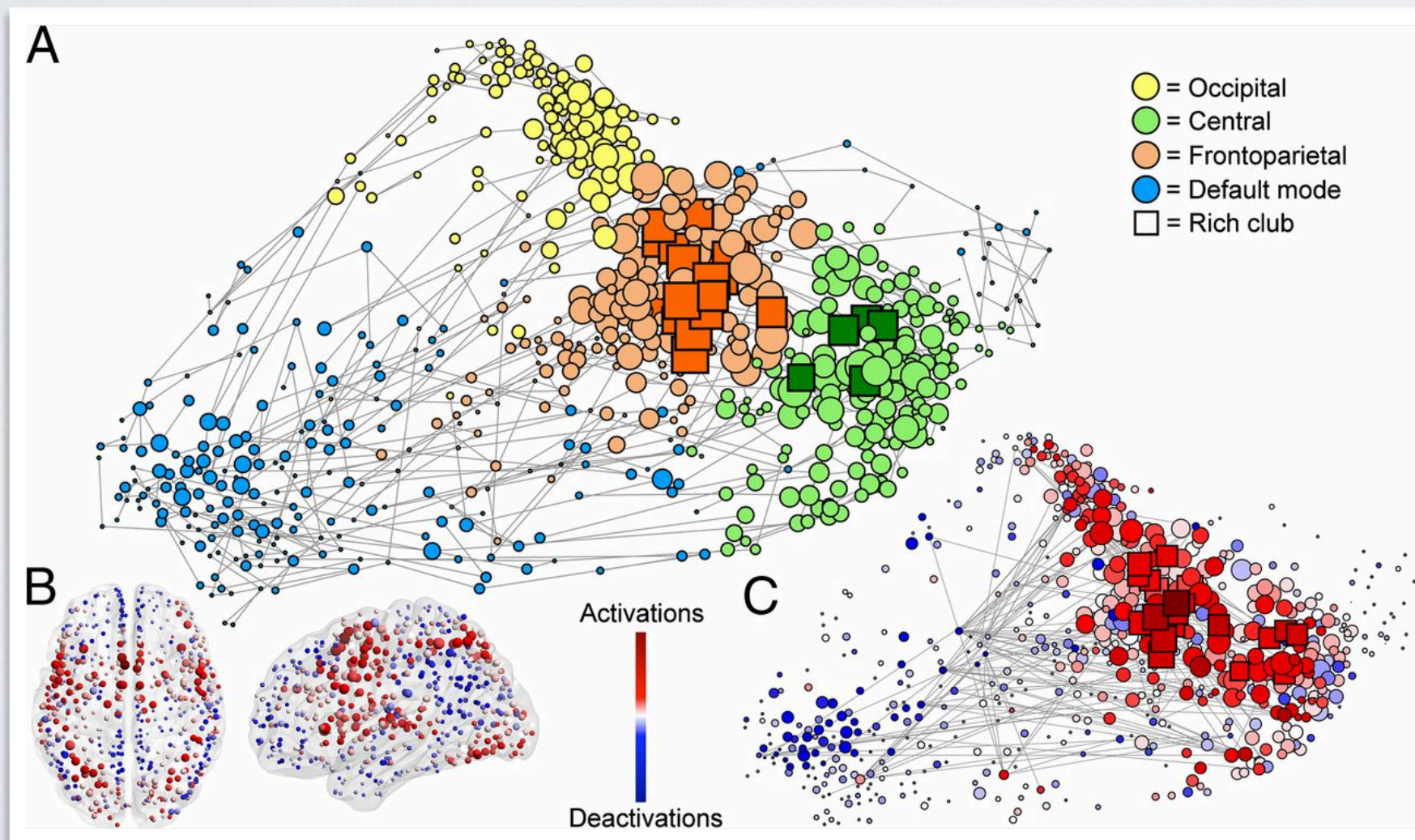
- Réseau social (Linked-in/Facebook/etc.)





# COMMUNAUTÉS DANS DES GRAPHERS RÉELS

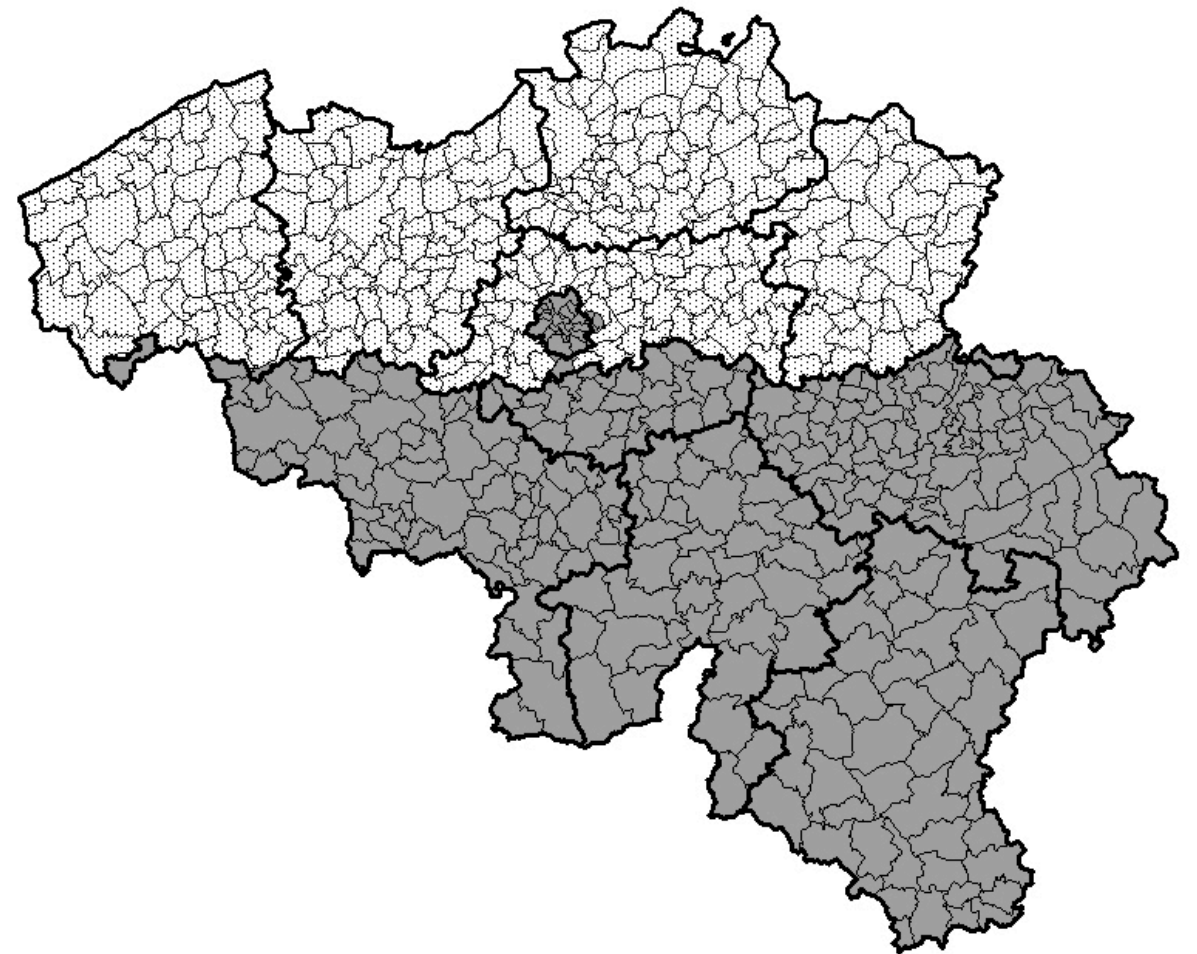
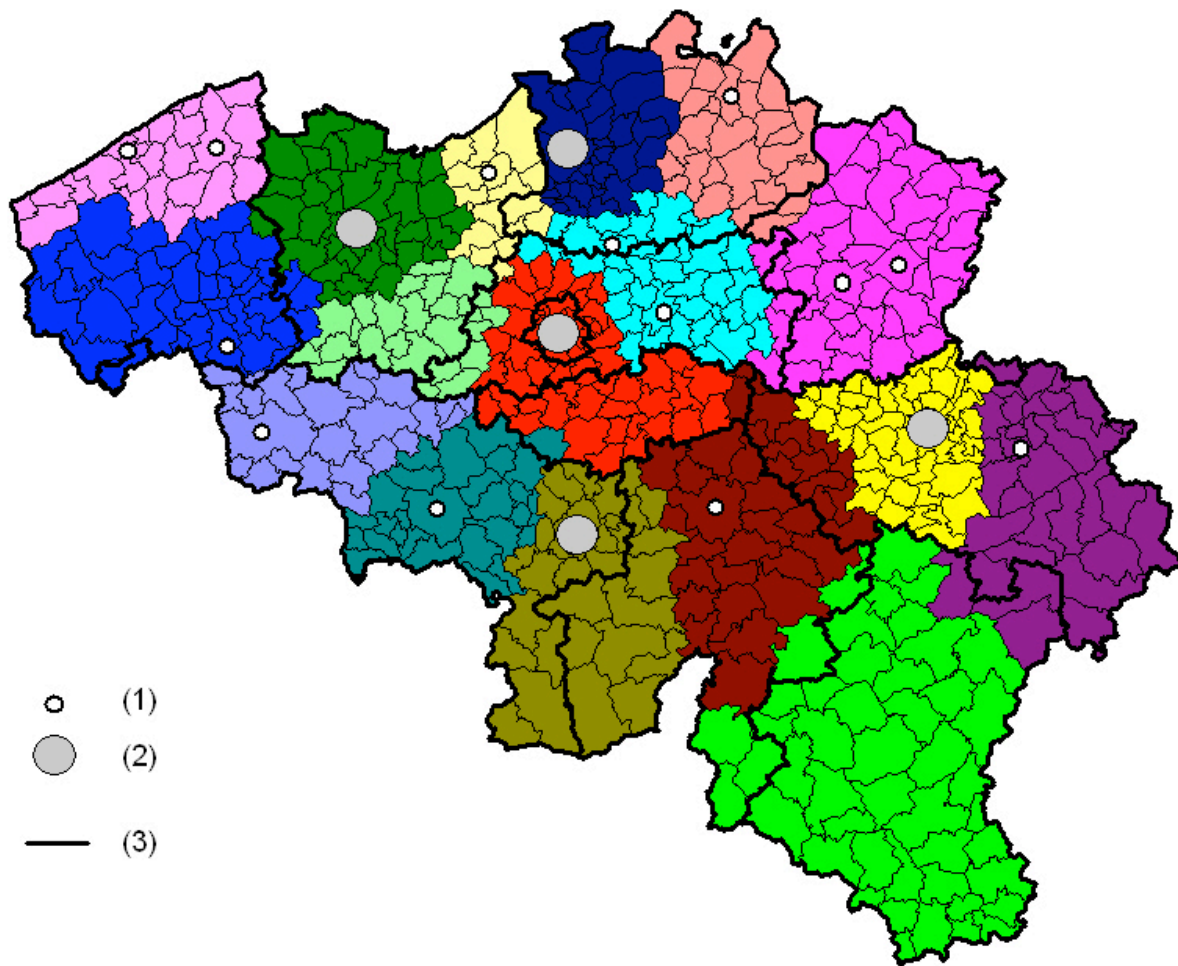
- Connexions dans le cerveau





# COMMUNAUTÉS DANS DES GRAPHS RÉELS

- Appels téléphoniques en Belgique ?



# ALGORITHME DE LOUVAIN

- Algorithme le plus connu, et celui présent dans Gephi
- Recherche à optimiser la **modularité**
  - **Modularité:** un score de “qualité” des communautés
  - L'algorithme cherche parmi toutes les partitions possibles celle de meilleur score (Approche gloutonne)
- Partitions *non recouvrantes*: chaque nœud appartient à une et une seule communauté.



# MODULARITÉ

- Défini comme la différence entre:
  - La fraction des liens **observés** à l'intérieur des communautés
  - La fraction des liens **attendus** à l'intérieur des communautés
    - Attendu si les liens étaient distribués au hasard dans un graphe où l'on conserve le nombre de nœuds, de liens, et les degrés de chaque nœud.

$$Q = \frac{1}{L} \sum_{i=1}^{|C|} (L_i - \frac{1}{2} K_i^2)$$

with  $L_i = L(H(c_i))$  the number of edges inside community  $i$  and  $K_i = \sum_{u \in c_i} k_u$  the sum of degrees of nodes in community  $i$ .

# ALGORITHME DE LOUVAIN

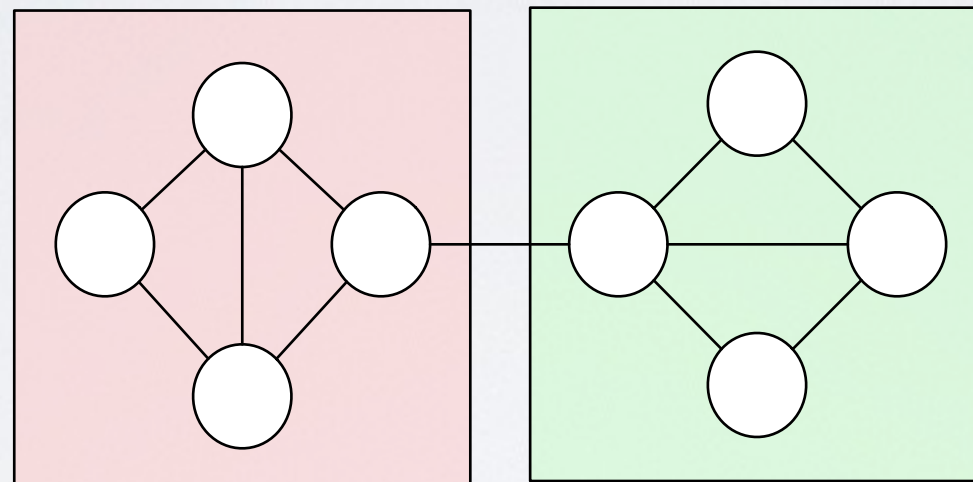
- On donne un graphe à l'algorithme, il retourne une partition (un ensemble de communautés)
- Attention, définition imparfaite:
  - Limite de modularité=>paramètre de résolution
  - Pas de garantie de trouver la "meilleure" solution
  - Algorithme stochastique: 2 exécutions peuvent renvoyer des résultats différents
- En pratique, fonctionne très bien



# MODULARITY INTUITION

$$n = 8$$

$$m = 11$$

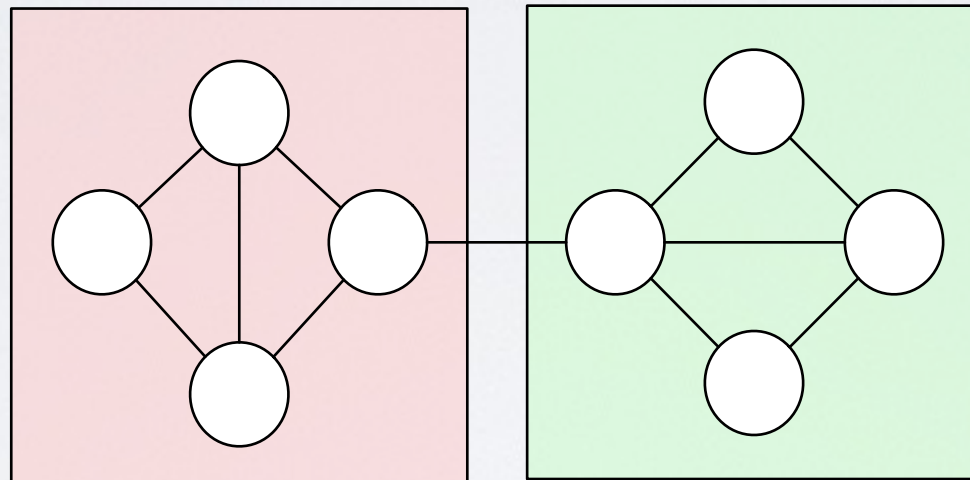


ER random graph

# MODULARITY INTUITION

$$\begin{aligned}n &= 8 \\m &= 11 \\p(u, v) &\approx 0.39\end{aligned}$$

$$d(G) = p(u, v) = \frac{11}{\frac{1}{2}8(8-1)} = \frac{11}{28} \approx 0.39$$



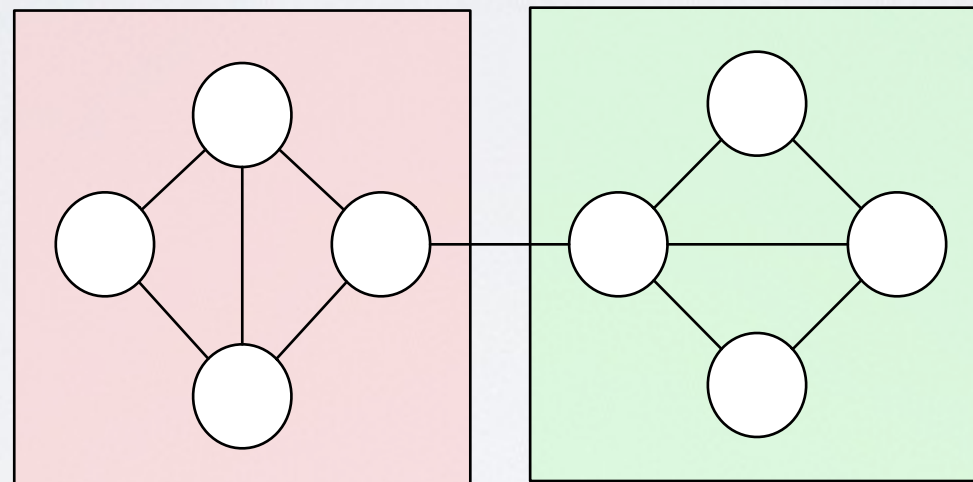


# MODULARITY INTUITION

$$n = 8$$

$$m = 11$$

$$p(u, v) \approx 0.39$$



ER random graph

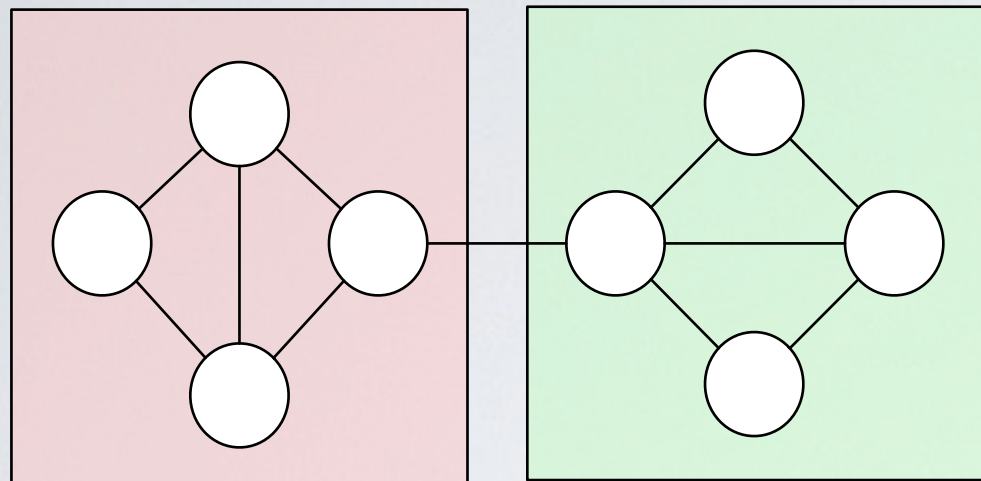
Expected edges inside red (or green)  
(#node pairs \* prob to observe an edge)

$$\frac{4(4-1)}{2} * p(u, v) = 2.34$$

---

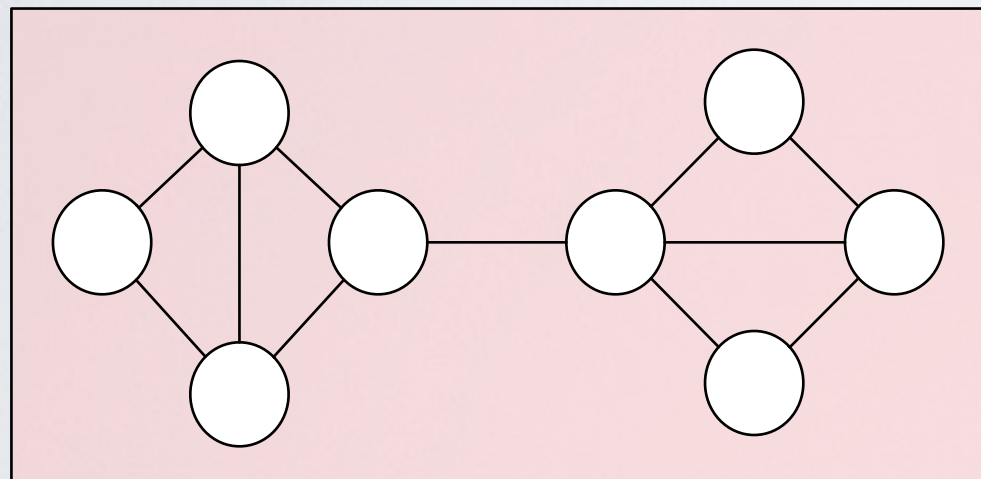
$$\text{Modularity} = \frac{2(5 - 2.34)}{m} = 0.48$$

# MODULARITY INTUITION

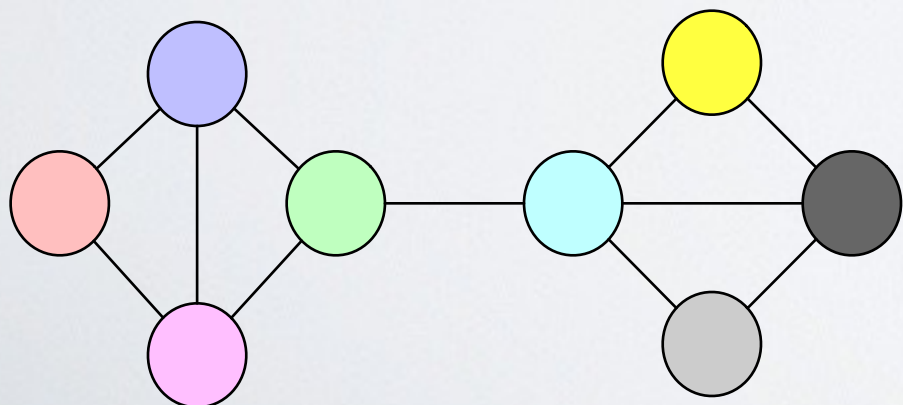


$$n = 8$$
$$m = 11$$

$$Q = 0.48$$



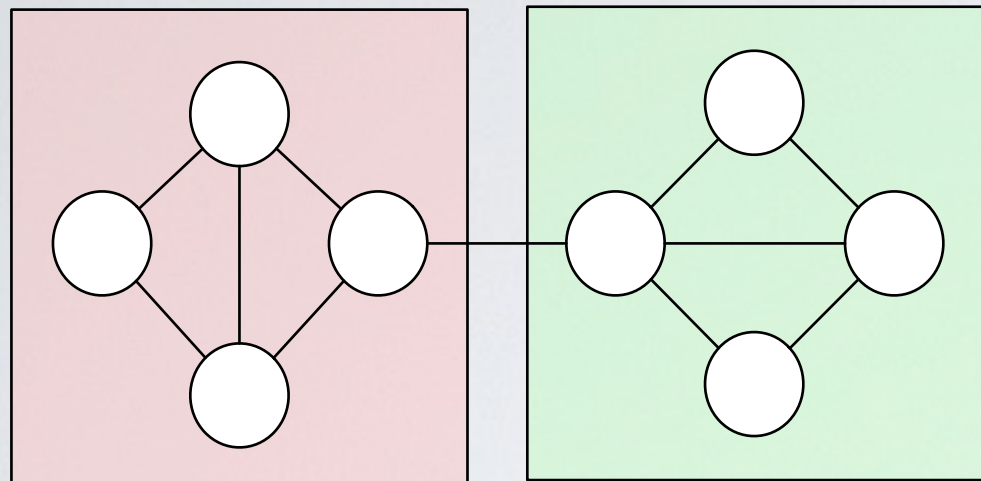
$$Q = ?$$



$$Q = ?$$

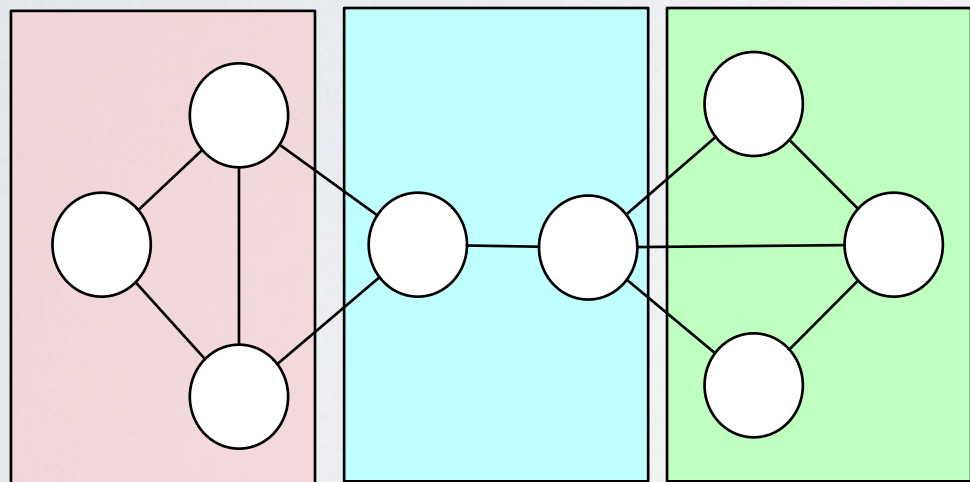


# MODULARITY INTUITION

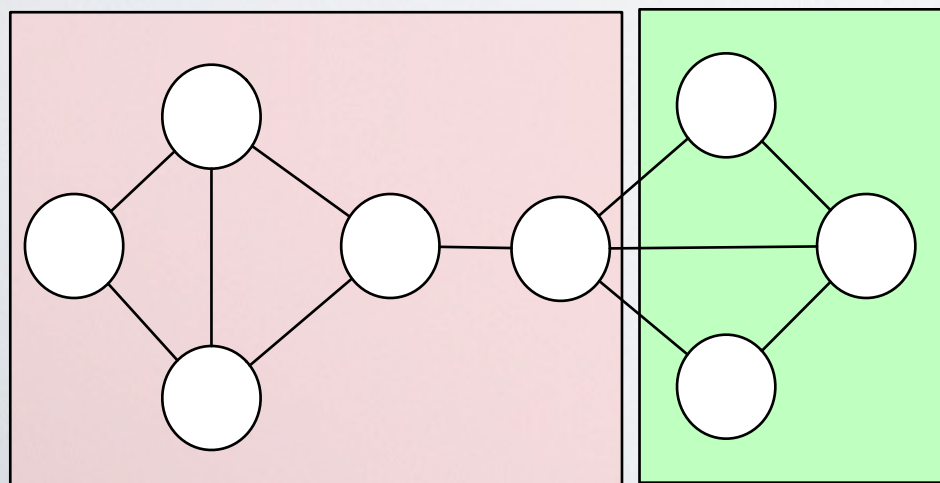


$$p=0.39$$

$$Q = (5-6p) + (5-6p) = 10-12p = 5.32$$

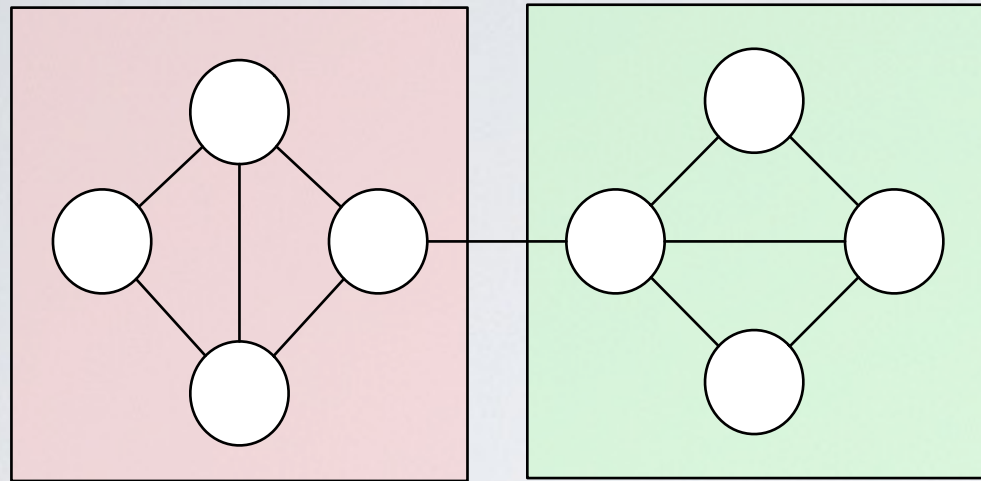


$$Q = (3-3p) + (1-p) + (2-3p) = 7-7p = 4.27$$



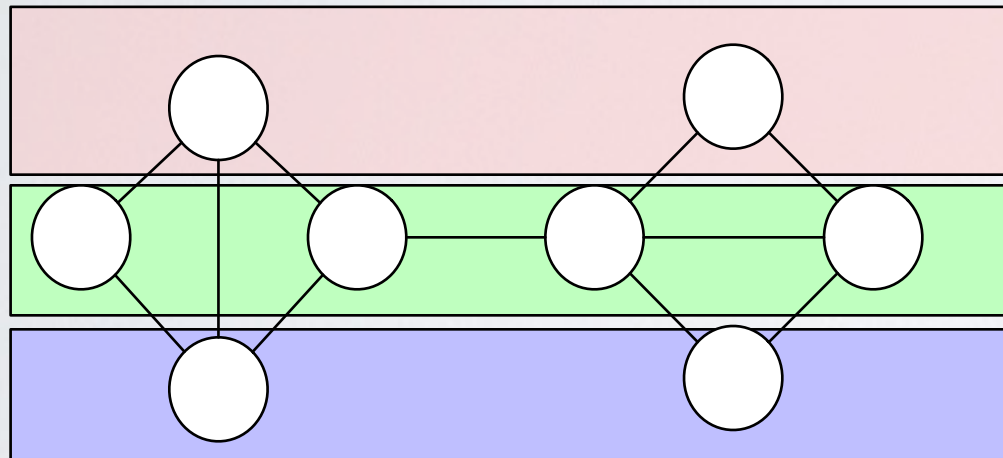
$$Q = (6-10p) + (2-3p) = 8-13p = 2.93$$

# MODULARITY INTUITION



$$p=0.39$$

$$Q = (5-6p) + (5-6p) = 10-12p = 5.32$$



$$Q = (0-p) + (2-6p) + 0-p = 2-8p = -0.34$$