

# M2 – NSD (Projet)

## Détection de communautés dans les graphes

Lionel Tabourier, Fabien Tarissan

Le projet sera réalisé en binôme. Il vous est demandé de présenter une méthode de détection de communautés dans les graphes, et de montrer des résultats expérimentaux obtenus au cours d'une courte présentation orale.

Vous avez trois choix possibles, par difficulté décroissante :

1. Proposer votre propre méthode de détection communautaire. (/20 + Bonus)
2. Faire votre propre implémentation d'une méthode existante. (/20)
3. Reprendre une implémentation existante d'une méthode existante. (/16)

### Consignes pour la présentation :

Les présentations auront lieu au cours de la séance du **22/11/2016**.

- Les présentations durent 15 minutes chacune (prière de respecter cette limite), elles peuvent être en français ou en anglais, chaque membre du binôme présentera une partie du travail.
- Vous utiliserez un diaporama pour la présentation, il vous est demandé de l'envoyer avant le 21/11 en format pdf, avec les codes que vous avez utilisés pour faire les expériences, et les instructions nécessaires pour les exécuter.
- Il est attendu pour la présentation :
  - une explication de l'algorithme (partie la plus importante) et une évaluation de sa complexité,
  - une présentation de résultats expérimentaux : jeux de données utilisés (et caractéristiques), valeur du score d'optimisation (par exemple modularité), caractéristiques des communautés obtenues (par exemple distribution de taille), temps de calcul, comparaison à d'autres méthodes (par exemple Louvain)
  - vous pouvez conclure sur une discussion des résultats : variantes/améliorations que vous envisagez, efficacité, ou toute réflexion que vous jugez pertinente.

Pour les binômes qui choisissent les options 2 ou 3, vous trouverez au verso une liste d'algorithmes recommandés, un article qui les décrit et une implémentation existante. Les implémentations sont présentes à titre purement indicatifs, certaines méthodes en ont d'autres et toutes n'ont pas été testées.

Vous pouvez choisir un algorithme en-dehors de cette liste (hormis celui de Louvain, présenté en cours). Chaque binôme présentera une méthode différente des autres, il vous est demandé de choisir la méthode que vous allez présenter pour le 25/10, selon le principe "premier arrivé, premier servi".

## Liste de méthodes :

- Divisive edge-betweenness algorithm :  
*Community structure in social and biological networks*, Girvan and Newman, 2002.  
Implémentation en python : [github.com/kjahan/community](https://github.com/kjahan/community)
- Random walk based algorithm :  
*Computing Communities in Large Networks Using Random Walks*, Pons and Latapy, 2005.  
Implémentation en C++ : [www-complexnetworks.lip6.fr/~latapy/PP/walktrap.html](http://www-complexnetworks.lip6.fr/~latapy/PP/walktrap.html)
- Leading eigenvector algorithm :  
*Finding community structure in networks using the eigenvectors of matrices*, Newman, 2006.  
Implémentation en R : [igraph.org/r/doc/cluster\\_leading\\_eigen.html](http://igraph.org/r/doc/cluster_leading_eigen.html)
- Simulated annealing algorithm :  
*Functional cartography of complex metabolic networks*, Guimera and Amaral, 2005.  
Implémentation en C : [seeslab.info/downloads/network-c-libraries-rgraph](http://seeslab.info/downloads/network-c-libraries-rgraph)
- Label propagation algorithm :  
*Near linear time algorithm to detect community structures in large-scale networks*, Ragahavan et al., 2007.  
Implémentation en R : [igraph.org/r/doc/cluster\\_label\\_prop.html](http://igraph.org/r/doc/cluster_label_prop.html)
- K-cliques based algorithm (overlapping communities) :  
*Uncovering the overlapping community structure of complex networks in nature and society*, Palla et al., 2005.  
Implémentation en R : [igraph.wikidot.com/community-detection-in-r](http://igraph.wikidot.com/community-detection-in-r)
- Physics inspired algorithm (overlapping communities) :  
*Detecting fuzzy community structures in complex networks with a Potts mode*, Reichardt and Bornholdt, 2004.  
Implémentation en R : [igraph.org/r/doc/cluster\\_spinglass.html](http://igraph.org/r/doc/cluster_spinglass.html)
- Map equation algorithm :  
*Maps of random walks on complex networks reveal community structure*, Rosvall and Bergstrom, 2007.  
Implémentation en R : [igraph.org/r/doc/cluster\\_infomap.html](http://igraph.org/r/doc/cluster_infomap.html)