Introduction to Network Analysis 2020/21 (x)

# Stochastic block models, $k$-cores decomposition

You are given two small social networks with known sociological partitioning of nodes.

- Zachary karate club network (2 groups)
- Davis southern women network (3 groups)

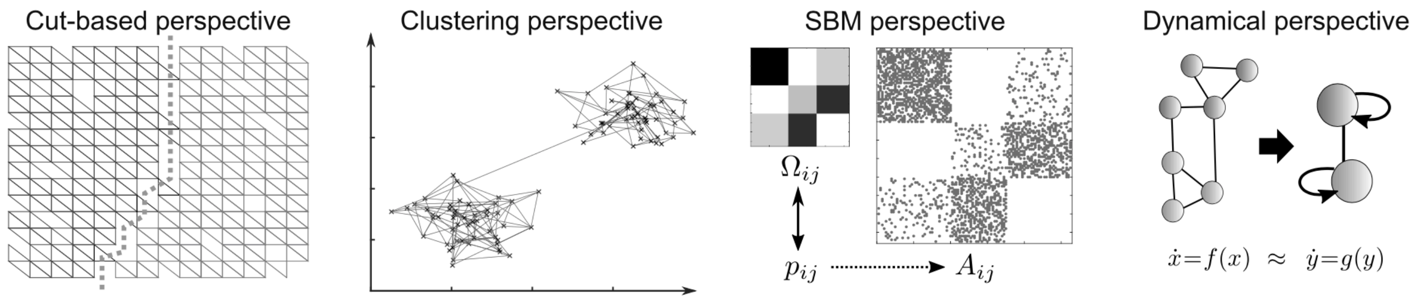
Later, you will be studying also four larger networks with labels associated with nodes.

- JUNG class dependency network (packages & classes)
- Java class dependency network (packages & classes)
- WikiLeaks cable reference network (identifiers & embassies)
- iMDB actors collaboration network (first names & surnames)

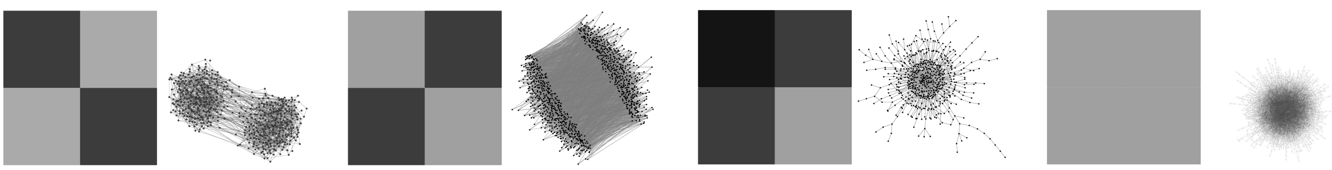
All networks are in Pajek format, whereas edge list and LNA formats are also available.

## I. Stochastic block models vs community detection

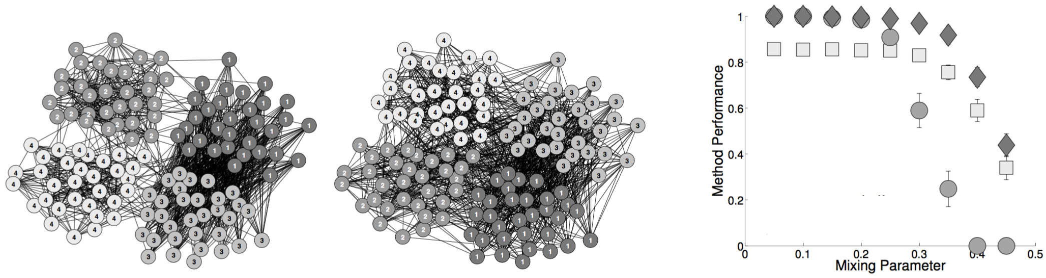
Browse CDlib, NetworkX, graph-tool or other library for implementations of community detection algorithm called Infomap and (degree-corrected) stochastic block models based on minimum description length. You will be comparing these two algorithms within the exercises below.




1. **(code)** Apply the algorithms to small social networks above and test whether the revealed groups coincide with sociological partitioning of these networks. You should apply the algorithms to each network multiple times and compare partitions using some standard measure. Since these networks are very small, you can also visualize the results of different algorithms. What type of structure do different algorithms reveal?

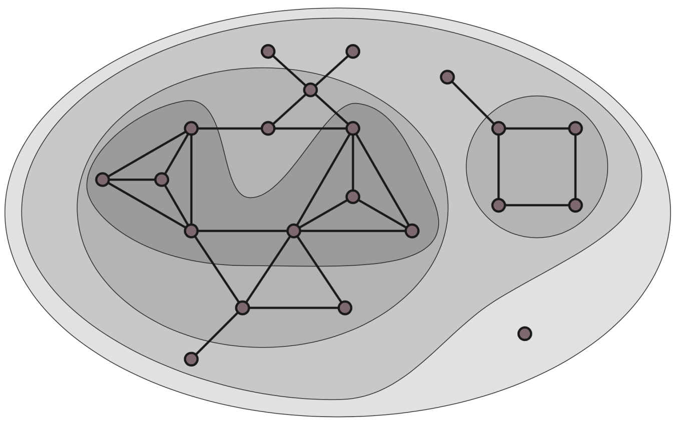


2. **(code)** Apply the algorithms to synthetic networks with planted community and disassortative structure, and test whether the revealed clusters coincide with the planted group structure. You should apply the algorithms to networks with varying mixing parameter $\mu$ and compare partitions using some standard measure. For which values of $\mu$ do the algorithms reveal the planted partition?




## II. Network $k$-cores decomposition

Consider the following algorithm for computing network $k$-cores for a given $k$. Starting with the original network, iteratively remove nodes with degree less than $k$. When no such node remains, connected components of the resulting network are the $k$-cores.



1. **(code)** Implement the algorithm and compute all $k$-cores of the larger networks above. Print out the number of $k$-cores and the size of the largest one for different values of $k$. What is the maximum value of $k$ denoted $k_{max}$ for which there exists at least one $k$-core?

2. **(code)** For each larger network, print out the labels of nodes in $k_{max}$-cores and try to interpret the results.