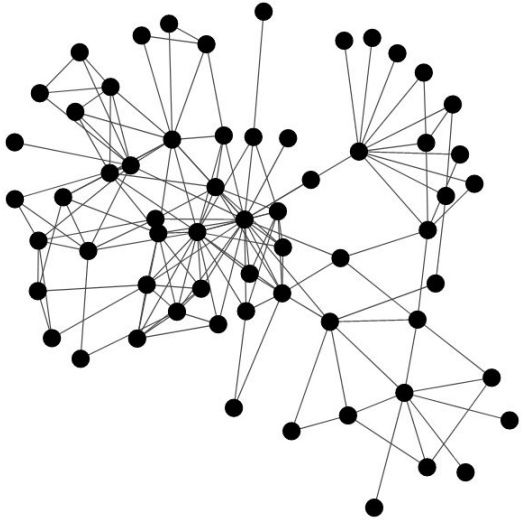
UPPSALA UNIVERSITET

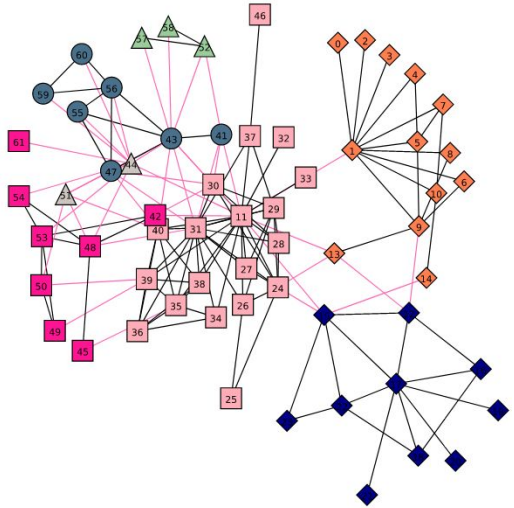
Stochastic Block Modeling in Bipartite graphs

Project Members
Deepthi Hulithala Venkataramana, Naeim Rashidfarokhi
Supervisor
Davide Vega D'Aurelio

Department of Information Technology
Uppsala university

Project presentation 2021





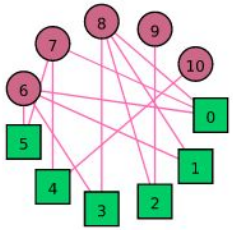
dehu1371@student.uu.se, nara5324@student.uu.se, davide.vega@it.uu.se

Community detection & Applications

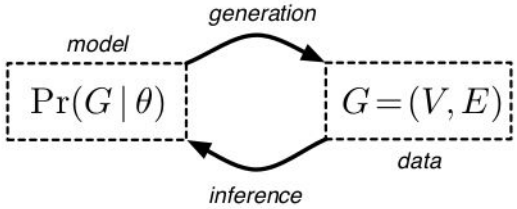
- ★ What is a network?
- ★ Why is it important?
  - Many real world interactions are in form of graph
  - Criminology, Public health, Politics, Social Network Analysis, Biology, etc

- ★ What to accomplish (Goal of the project)?
  - Revealing inherent community structures hidden in a network
  - A bigger picture of meaningful interactions among entities
  - Learn a powerful tool for this purpose named Stochastic Block Model

- ★ Bipartite network: A network whose nodes can be divided into two disjoint independent sets X and Y such that
  - Edges exist between the nodes of two sets
  - Edges do not exist within the nodes of a particular set.



Stochastic block model (Standard form) (I/III)



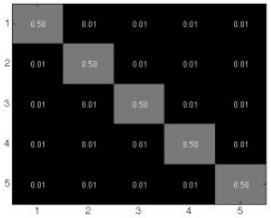
Stochastic block model and network

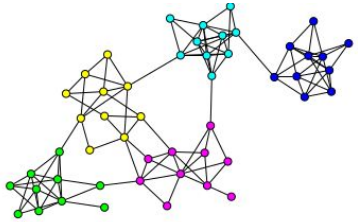
Stochastic block model(SBM) as generative model for structure detection in network

★ Generation: having a set of parameters as θ, draw an instance network G from this distribution. θ decomposes into:
- k = Number of communities in the network
- g = n x 1 vector where z gives the community index of each vertex
- ω = k x k stochastic block matrix where ω gives the probability that a vertex of community ‘i’ is connected to a vertex of community ‘j’.

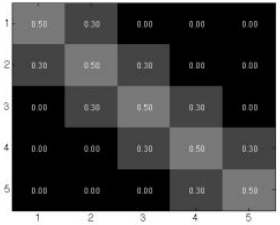
★ Inference: Having V number of vertices and a set of edge as E between them, find a set of parameters as θ which describes the network

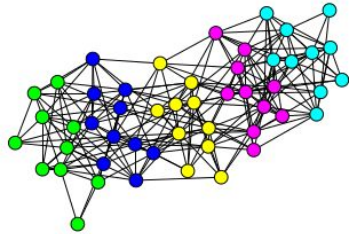
Stochastic block model (Standard form) (II/III)





assortative communities





ordered communities

Network Analysis and Modeling, lecture notes
Professor Aaron Clauset
https://www.cs.unm.edu/~aaron/blog/archives/2013/11/network_analysi.htm

Stochastic block model (Standard form) (III/III)

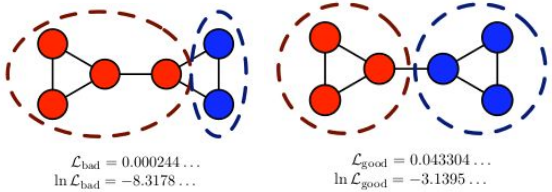
★ Maximum Likelihood Estimation (MLE)

$$P(G|\omega, g) = \prod_{u,v} P(u, v|\omega, g)$$

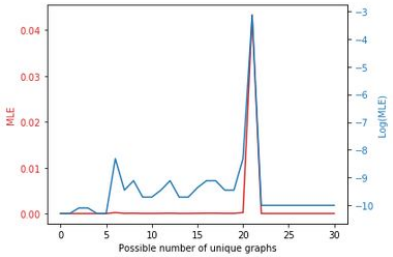
$$P(G|\omega, g) = \prod_{(u,v)\in E} P(u, v|\omega, g) \prod_{(u,v)\notin E} 1 - P(u, v|\omega, g)$$

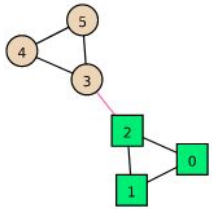
$$logP = \sum_{r,s} E_{rs}\, log\frac{E_{rs}}{N_{rs}} + (N_{rs} - E_{rs})\, log\left(\frac{N_{rs} - E_{rs}}{N_{rs}}\right) \quad or \quad logP(G|g) = \sum_{rs} \frac{m_{rs}}{2m} log\frac{\frac{m_{rs}}{2m}}{\frac{n_r n_s}{n^2}}$$

★ Communities of ‘r’ and ‘s’ while ‘*u*’ and ‘*v*’ are vertices
★ ‘E’ as existing edges or expected degrees, possible to count from the network
★ ’N’ as number of possible edges between communities
★ ‘m’ as total number of edges and ‘n’ as total number of nodes in the network
★ ‘m_{rs}‘ as number of edges between communities r and s



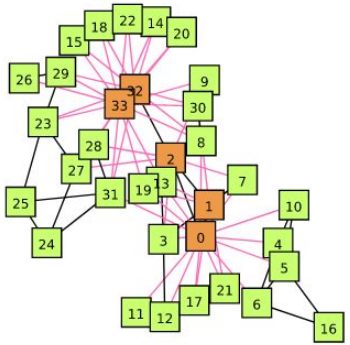
Network Analysis and Modeling, lecture notes
Professor Aaron Clauset
https://www.cs.unm.edu/~aaron/blog/archives/2013/11/network_analysis.htm





log(MLE) value:-3.1394888625872888

Drawbacks of SBM



Community detection using Standard SBM

★ If the network consists of skewed degree distribution, the standard SBM model tends to group vertices by degree.

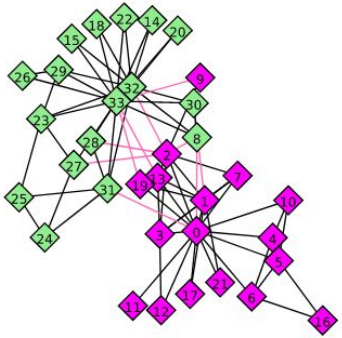
★ In the Zachary's karate club network with 2 communities, SBM detects the five highest-degree vertices as one community and all other vertices as another community, which does not correspond to the true or socially observed groups.

Degree Corrected SBM

$$logP(G|g) = \sum_{rs} \frac{m_{rs}}{2m} log \frac{\frac{m_{rs}}{2m}}{\frac{k_r}{2m}\frac{k_s}{2m}}$$

k_r = sum of degrees of the vertices in community r
m_{rs} = number of edges between communities r and s
m = total number of edges in the network

In degree corrected SBM, a parameter along with the parameter defined in SBM controlling the expected degree of the vertices is considered.

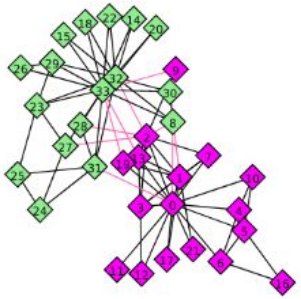


Community detection using Degree corrected SBM

Degree Corrected vs Standard SBM

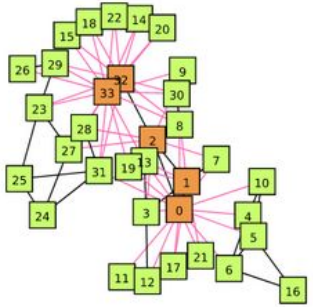
$$logP(G|g) = \sum_{rs} \frac{m_{rs}}{2m} log \frac{\frac{m_{rs}}{2m}}{\frac{k_r}{2m}\frac{k_s}{2m}}$$

$$logP(G|g) = \sum_{rs} \frac{m_{rs}}{2m} log \frac{\frac{m_{rs}}{2m}}{\frac{n_r n_s}{n^2}}$$



Community detection using Degree corrected SBM

r, s = communities
n_r = number of vertices in community r
m = total number of edges in the network
m_{rs}= number of edges between communities r and s
k_r = sum of degrees of the vertices in community r



Community detection using Standard SBM

★ In networks with substantial degree heterogeneity, the Standard SBM prefers to split networks into communities of high and low degree which can prevent from finding true community memberships.

★ The degree-corrected model correctly ignores divisions based solely on degree and hence is more sensitive to underlying structure.

Brute Force Search

★ Brute-force search as exhaustive search also known as generate and test to find Maximum Likelihood Estimation

$$ {}_nC_r = \frac{n!}{r!(n-r)!} $$

${}_nC_r$ = number of combinations
n = total number of objects in the set
r = number of choosing objects from the set

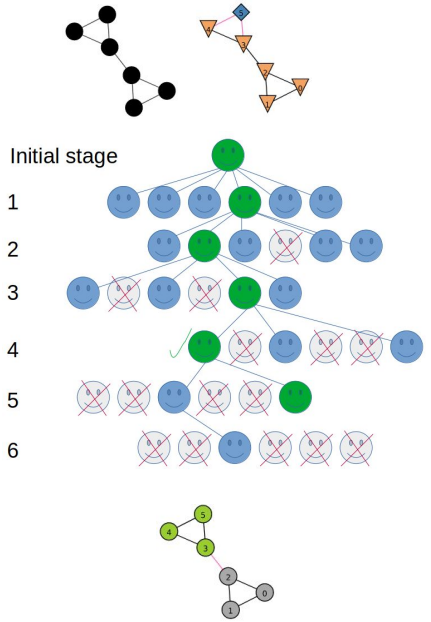
Network grouping	32 nodes and 5 communities	62 nodes and 7 communities	10000 nodes and 4 communities
Possibilities	201376	491796152	4.1642×10^{14}

Heuristic Search

- ★ Local vertex switching algorithm
- ★ A local heuristic algorithm similar to the Kernighan-Lin algorithm
- ★ Faster than one vertex Monte-Carlo

Algorithm 1: Local heuristic search

```
for itr in n iterations: do
    divide the network into K random communities;
    set all values in movement-control-dictionary to False;
    for vertex in movement-control-dictionary: do
        if vertex not moved: then
            for key,value in dict.items: do
                if key is alone in its group: then
                    pass;
                else
                    calculate new-mle;
                    if new-mle > old-mle: then
                        replace and hold network partitioning corresponded to new-mle;
                        replace and hold new-mle with old-mle;
                        replace and hold new-vertex-target-group with
                         old-vertex-target-group;
                    else
                        pass;
                    end
                end
            end
            set founded vertex as moved from final new-vertex-target-group value;
            replace original partitioning with the one corresponded to new-mle for another
             investigation;
            save new-mle result for final comparison;
        else
            pass;
        end
    end
    find accumulative MLE peak as iteration result and save it;
end
Among all iterations find the maximum peak as final result;
```



Choosing the number of communities k

Minimum description length(MDL)

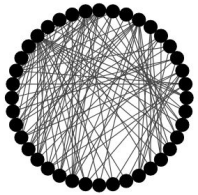
$$\Sigma_k' = Eh\left(\frac{k(k+1)}{2E}\right) - (E-N)ln\ k$$

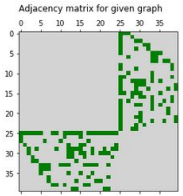
$$\mathrm{h(x)} = (1+\mathrm{x})\ \ln(1+\mathrm{x})$$

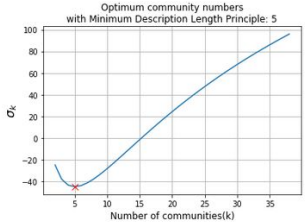
E = Total number of edges in the network
N = Total number of the nodes in the network
Σ = description length
k = number of communities

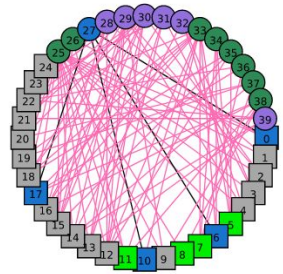
Experiments(I/II)

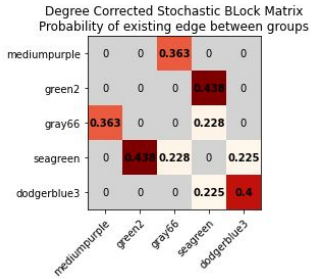
A bipartite network of the memberships of chief executive officers and the social organizations (clubs) to which they belong, 1985
40 Nodes, 95 Edges, Bipartite graph!





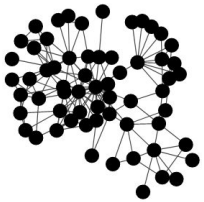


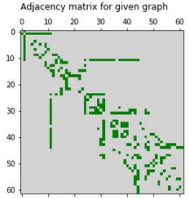


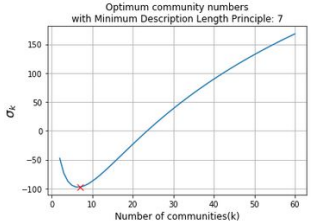


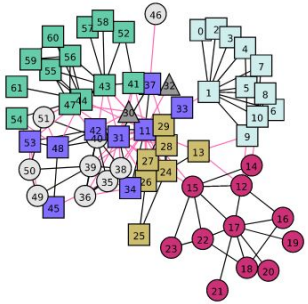
log(MLE) value:0.4353912117242914
Execution_time_local_heuristic:82.0841s after 125 iterations!

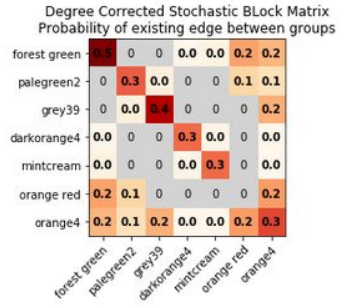
Network of individuals and their known social associations, centered around the hijackers that carried out the September 11th, 2001 terrorist attacks. Associations extracted after-the-fact from public data. 62 Nodes, 152 Edges, Unipartite graph!







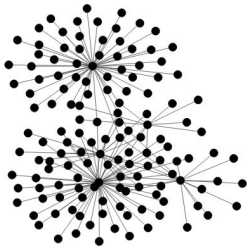


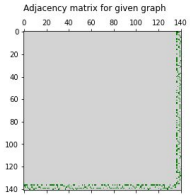


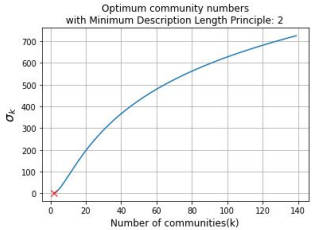
log(MLE) value:0.8700012395816545
Execution_time_local_heuristic:434.3286s after 100 iterations!

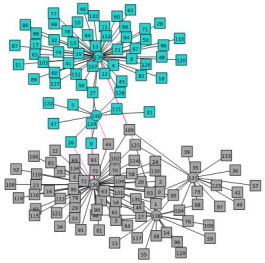
Experiments(II/II)

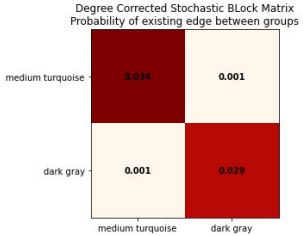
A bipartite network of the memberships of notable people and organizations, from the American Revolution (1765-1783) between users and groups on YouTube, extracted from a larger YouTube network in 2007
141 Nodes, 160 Edges, Bipartite graph!





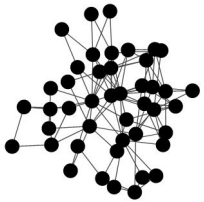


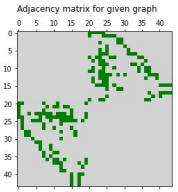


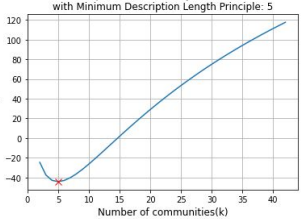


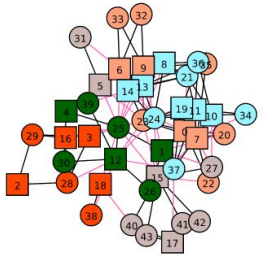
log(MLE) value:0.559082130525731
Execution_time_local_heuristic:65.7727s after 125 iterations!

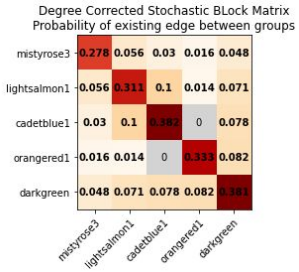
A small bipartite network of the affiliations among elite individuals and the corporate, museum, university boards, or social clubs to which they belonged, from 1962.
44 Nodes, 99 Edges, Bipartite graph!











log(MLE) value:0.5300446966991584
Execution_time_local_heuristic:39.2809s after 100 iterations!

Summary

★ We have studied and learnt about
- Blockmodeling with standard SBM
- Blockmodeling with degree corrected SBM
- Minimum Description Length concept
- Heuristic search algorithm to find highest likelihood

★ We have improved our programming skills
- Efficient use of data structures; dictionary vs lists in Python
- Efficient programing to reduce the computation cost and time
  - Local updating of existing data when only one vertex moves
- Be introduced to new packages such as igraph, itertools, etc.

★ Outcome

				Intel(R) Core(TM) i5-8250U CPU @ 1.60GHz		Intel(R) Core(TM) i5-2410M CPU @ 2.30GHz	
Network	Nodes edges	communities (k)	Number of Iterations	Execution time(sec)	MLE value	Execution time(sec)	MLE value
CEOs and social clubs	40, 95	5	125	41.4862s	0.476100	79.9764s	0.457511
9/11 terrorist attack	62, 152	7	100	180.6786s	0.848257	355.0968s	0.884689
Youtube groups and Users about American Revolution	141, 160	2	125	63.6838s	0.559082	135.6801s	0.559082
Elite individuals and organisations	44, 99	5	100	44.1877s	0.512380	76.8285s	0.526460