Communities

Alberto Paccanaro

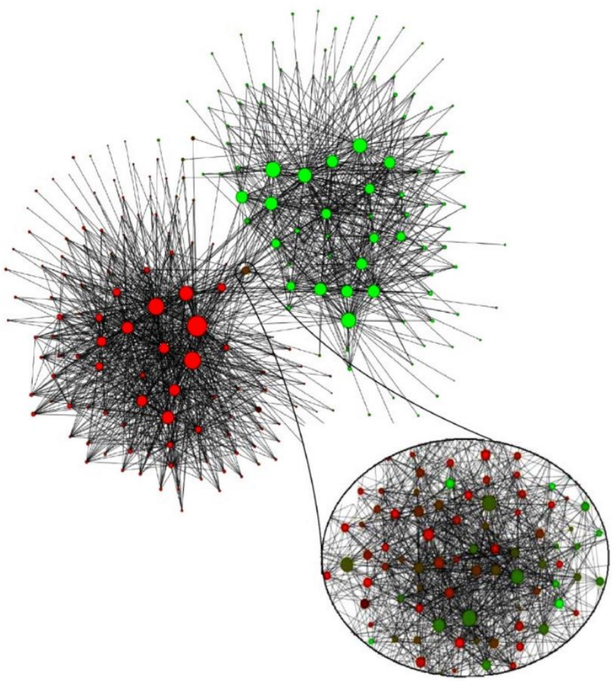
EMAp – FGV

www.paccanarolab.org

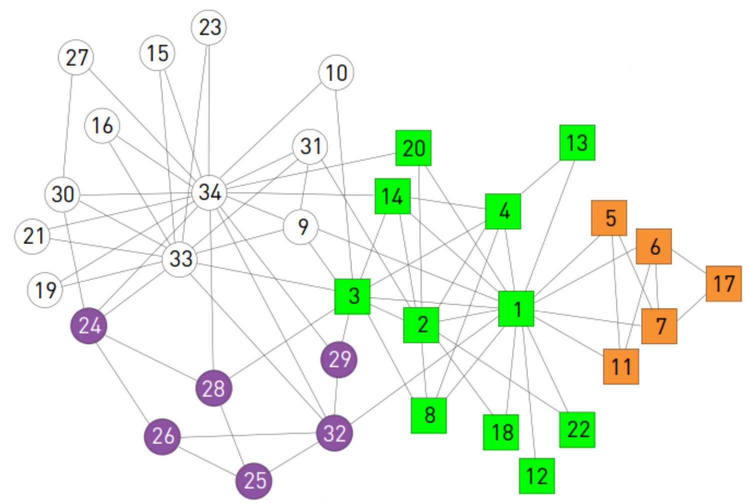
Some material and images are from (or adapted from):
A. Barabási, and M. Pósfai. Network science, Cambridge University Press, 2016

Communities

A group of nodes that have a higher likelihood of connecting to each other than to nodes



Belgian mobile phone company data



The 34 members of Zachary's Karate Club

Working hypothesis

- Existence of communities rooted in *who-connects-to-whom*.
- They cannot be explained based on the degree distribution alone.
- To find them we must inspect a network's wiring diagram.

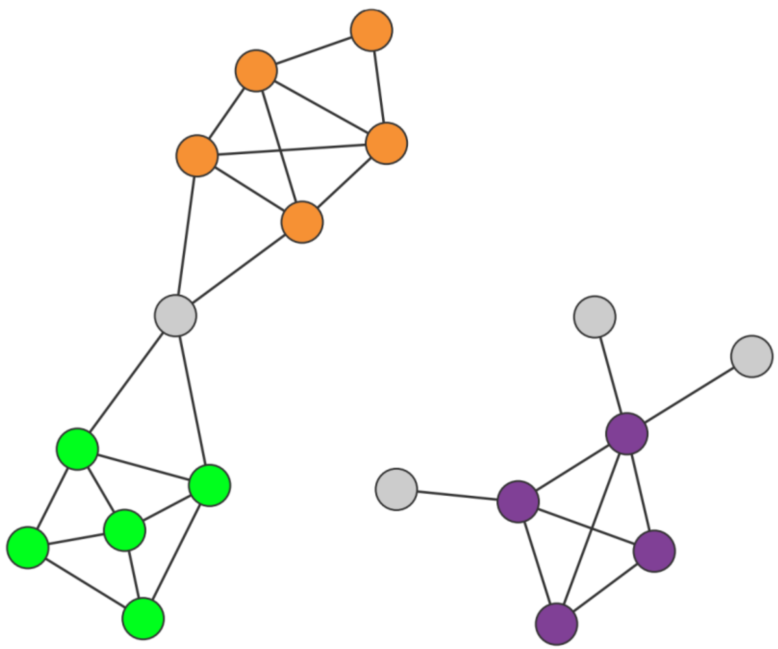
A network's community structure is uniquely encoded in its wiring diagram.

Community definitions

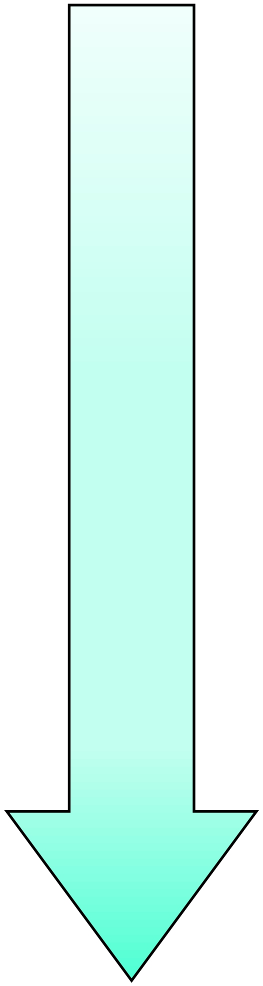
Connectedness and Density: a community is a locally <u>dense</u> <u>connected</u> subgraph in a network.

Connectedness

Density

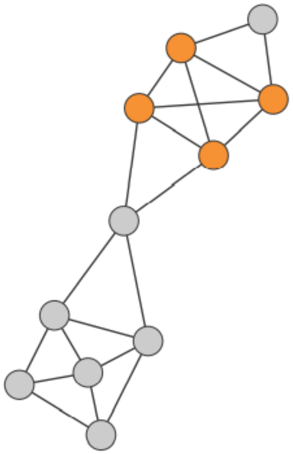


More retrictive



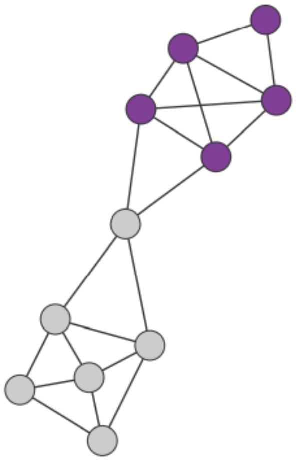
Less retrictive

Cliques: fully connected subgraph



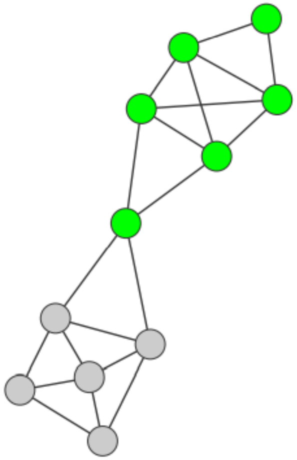
Strong Community: each node within C has more links within the community than with the rest of the graph.

$$k_i^{\mathrm{int}}(C) > k_i^{\mathrm{ext}}(C)$$



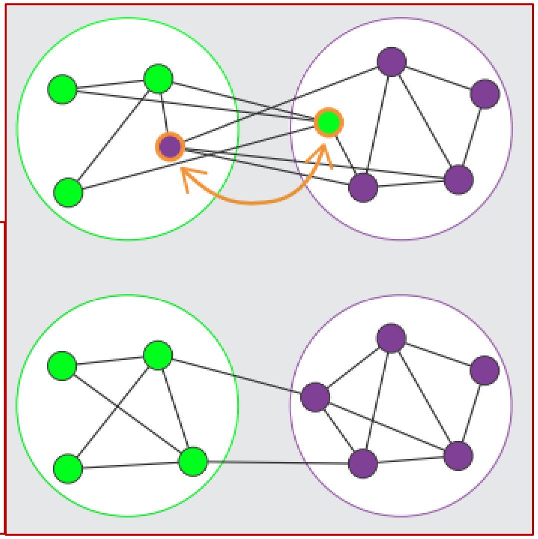
Weak Community: the total internal degree of a subgraph exceeds its total external degree

$$\sum_{i \in C} k_i^{\mathrm{int}}(C) > \sum_{i \in C} k_i^{\mathrm{ext}}(C)$$



Graph partitioning vs Community detection

Graph partitioning: divides a network into a predefined number of smaller subgraphs.



Kerningham-Lin algorithm:

- random partition
- Iterate swapping the pair that results in the largest reduction of the cut size (number of links between the nodes in the two groups)

Community detection: uncover the inherent community structure of a network.

Inspecting all the possible partitions is just not feasible…

Hierarchical clustering

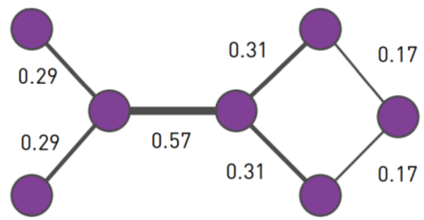
- Agglomerative hierarchical clustering

- Divisive hierarchical clustering

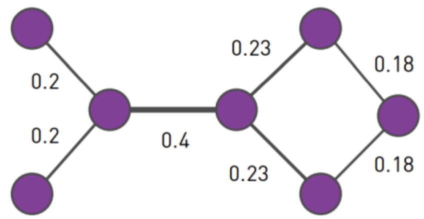
Divisive hierarchical clustering

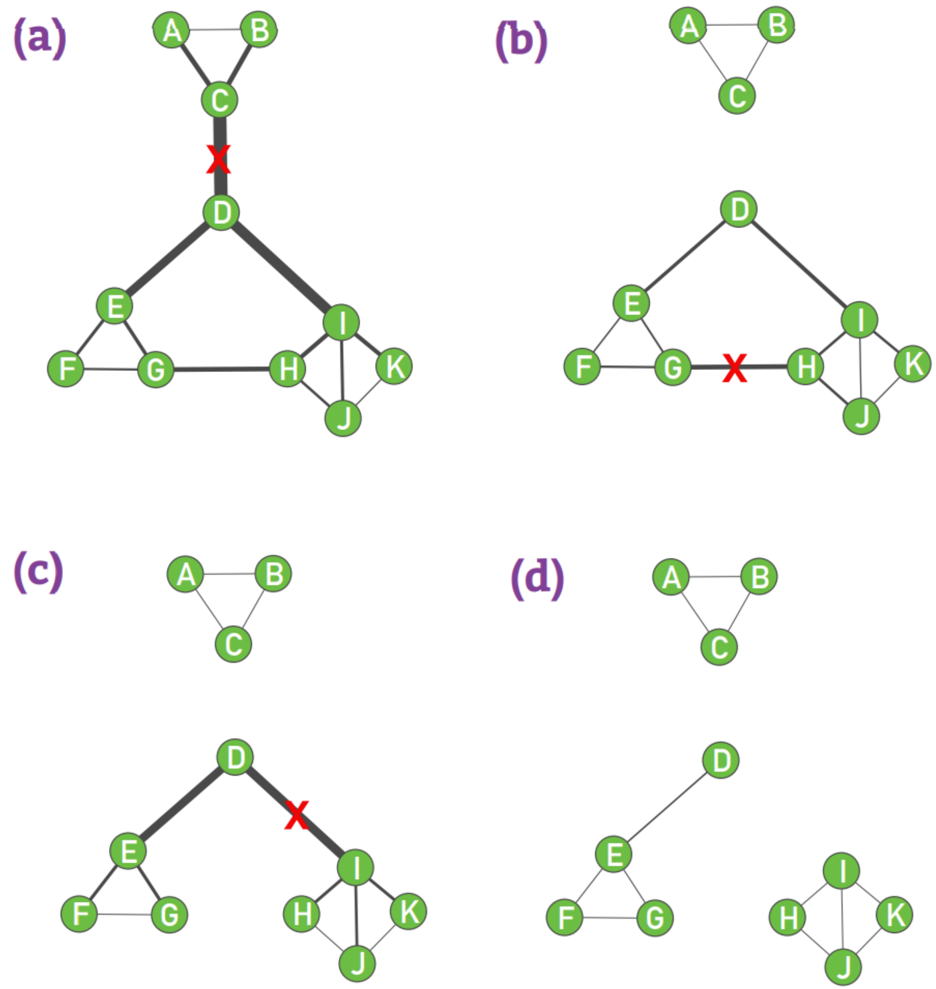
Remove the links connecting nodes that belong to different communities.

Link Betweenness: proportional to the number of shortest paths between all node pairs that run along the link *(i,j)*.



Random-Walk Betweenness: probability that the link $i \rightarrow j$ was crossed by a random walker, after averaging over all possible choices for the start/end nodes





Zachary's Karate Club

The divisive hierarchical algorithm of Girvan and Newman uses link betweenness

Modularity

Randomly wired networks lack an inherent community structure.

Idea: *compare link density of group of nodes with the link density obtained for the same nodes after randomly rewiring*

Modularity measures the quality of each partition

- allows us to decide if a particular community partition is better than some other one
- modularity optimization offers a novel approach to community detection

Difference between:

- the network's real wiring diagram (A_{ij})
- the expected number of links between i and j if the network is randomly wired (p_{ij})

$$M_c = \frac{1}{2L} \sum_{(i,j)\in C_c} (A_{ij} - p_{ij})$$

n_c communities
N_c nodes per comm.
L_c nodes per comm.
with $c = 1..\ n_c$

This is a measure for a specific community c

$$p_{ij} = \frac{k_i k_j}{2L}$$

To get the **modularity for a specific partition** for an entire graph, I can just sum it over all the communities:

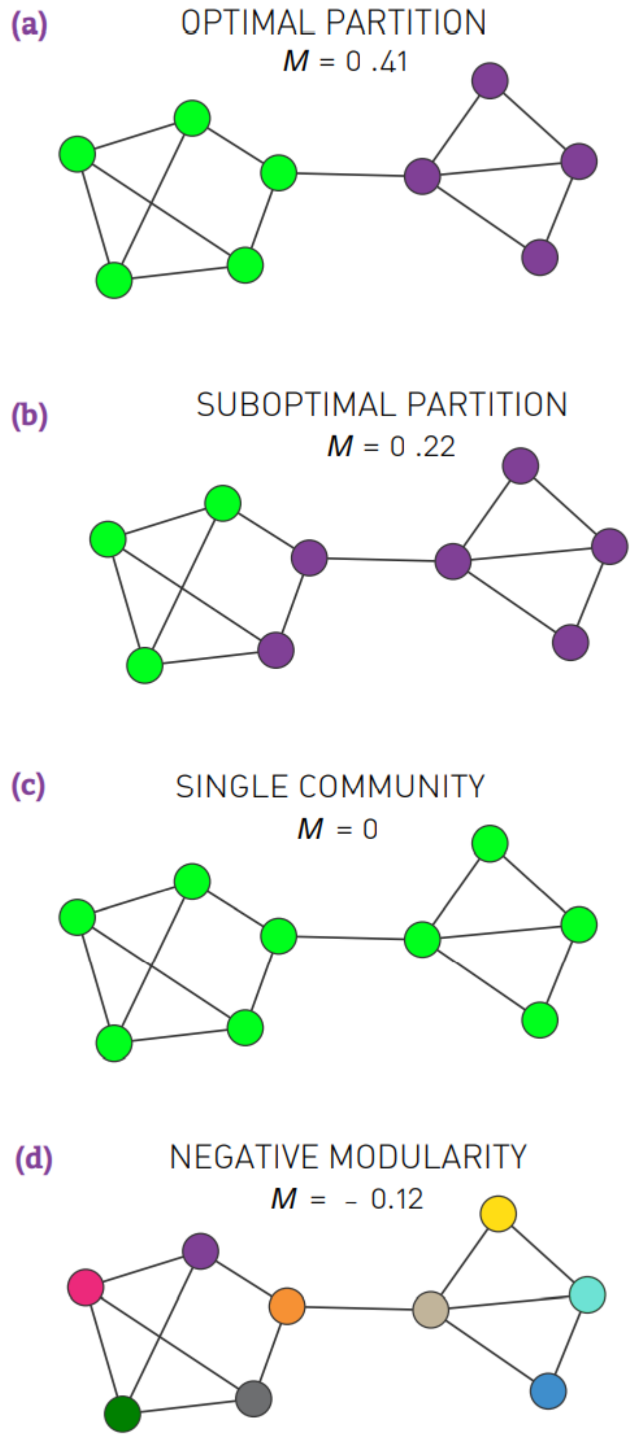
$$M = \sum_{c=1}^{n_c} M_c$$

$$M = \sum_{c=1}^{n_c} \left[\frac{L_c}{L} - \left(\frac{k_c}{2L} \right)^2 \right]$$

k_c is the total degree of the nodes in this community

Higher modularity implies better partition

➔ so, let's maximize it ☺

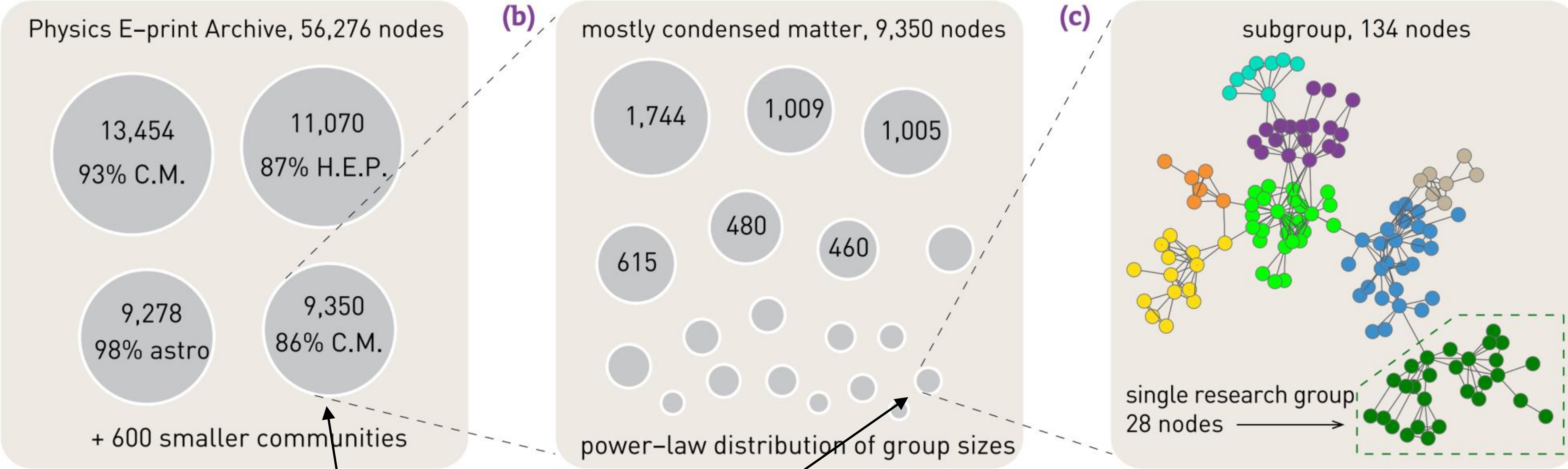


Greedy modularity maximization

iteratively joins pairs of communities if the move increases the partition's modularity

1. Start with each node in a separate community.
2. For each community pair connected by at least one link: compute the modularity difference ΔM obtained if we merge them.
3. Identify the community pair for which ΔM is the largest and merge them.
4. Goto 2, until all nodes merge into a single community, recording M for each step.
5. Select the partition for which M is maximal

Note that modularity is always calculated for the full network.



We can identify subcommunities by applying the greedy algorithm to each community, treating them as separate networks

Modularity limitation: resolution

A community, k_A total degree, *B* community, k_B total degree
l_{AB} number of links between them

Merging communities A and B into a single community, the network's modularity changes with:

$$\Delta M_{AB} = \frac{l_{AB}}{L} - \frac{k_A k_B}{2L^2}$$

Modularity maximization cannot detect communities that are smaller than the resolution limit

If $k_A k_B / 2L < 1$, then $\Delta M_{AB} > 0$ if $l_{AB} \geq 1$, hence they will be merged!

To simplify: $k_a \sim k_b = k$ and $k \leq \sqrt{2L}$ ➔ even a single link between them will force the two communities together when we maximize M.

Real networks do contain numerous small communities

Modularity Limitation: many similar maxima

Often, for many different partitions the difference in modularity is minimal.

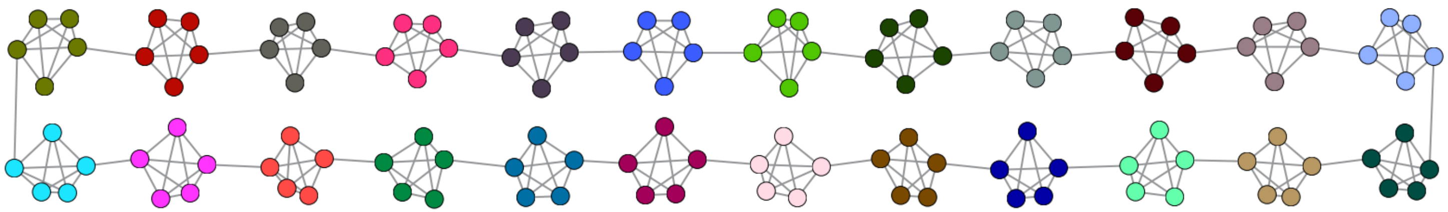
N_c subgraphs, with similar $k_c \sim 2L/n_c$ link densities
If we merge a pair of clusters:

$$\Delta M = \frac{l_{AB}}{L} - \frac{2}{n_c^2}$$

The change in modularity is tiny…

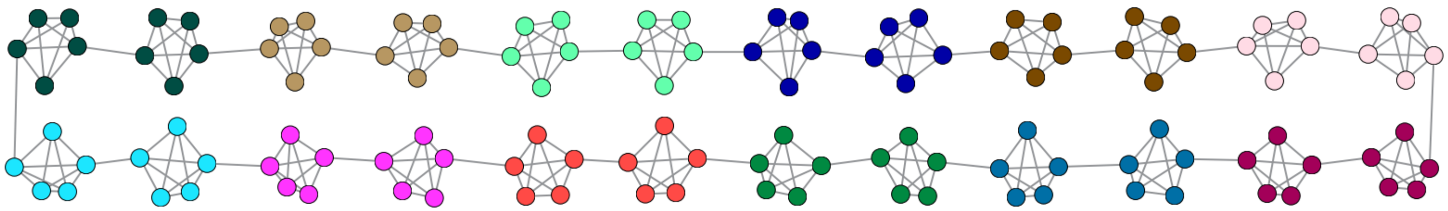
IT ONLY DEPENDS ON n_c

For a network with $n_c = 20$ communities, this change is at most $\Delta M = -0.005$!!!



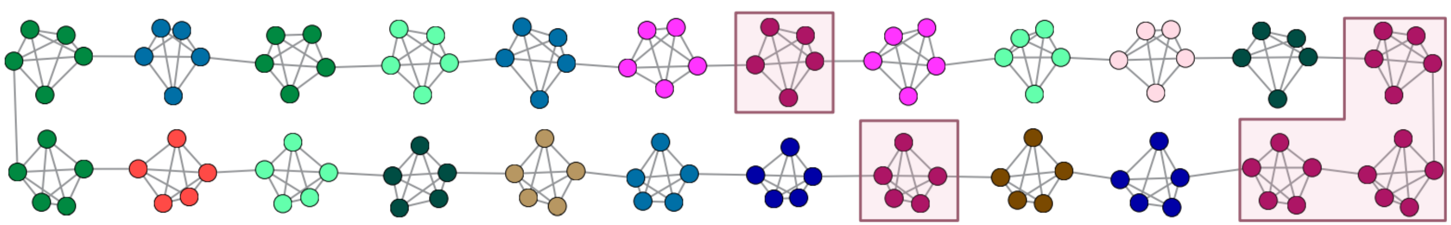
M=0.867

Intuitive Partition



M=0.871

Optimal Partition



M=0.80

Random Partition

As the number of groups increases, ΔM_{ij} goes to zero
➔ increasingly difficult to distinguish the optimal partition from the numerous suboptimal alternatives

- Modularity offers a first principle understanding of a network's community structure

- Limitations:
  - it forces together small weakly connected communities
  - networks lack a clear modularity maxima, instead many partitions with hard to distinguish modularity.
  - analytical calculations and numerical simulations indicate that even random networks contain high modularity partitions

Have a look at:

- Louvain algorithm
- Infomap algorithm (entropy-based)