



# Network Analysis

AN INTRODUCTION FOR HUMANISTS

Dr Katarzyna Anna Kapitan  
27 March 2025

# Modelling (Recap)



## Why and How

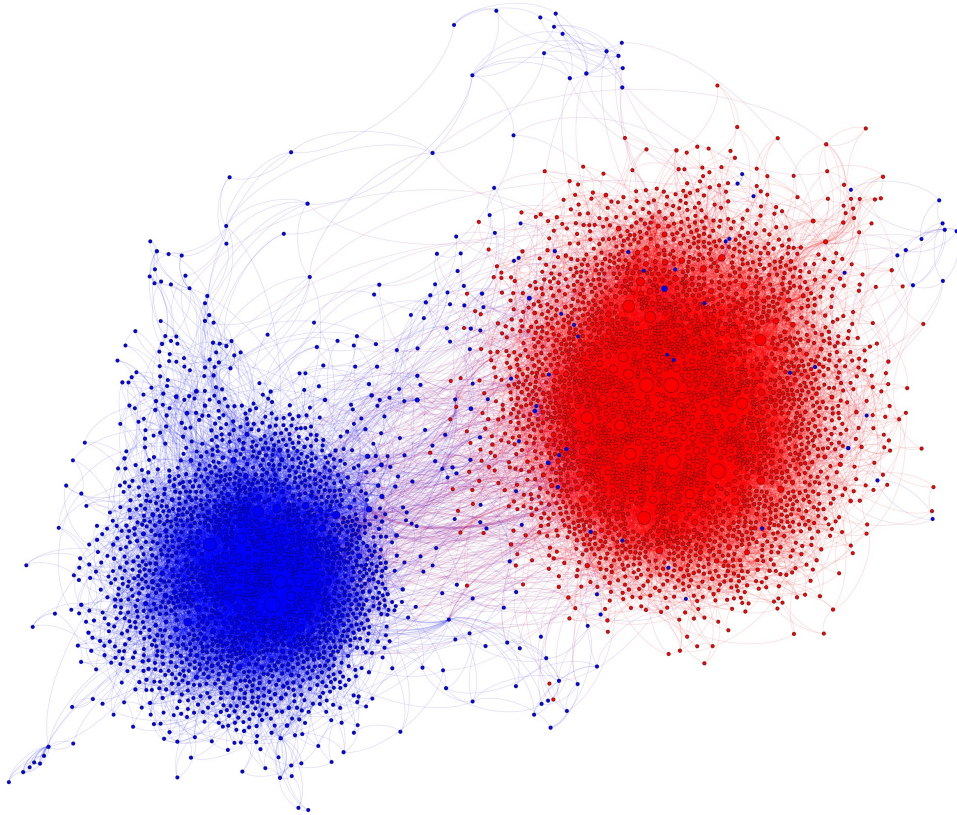


## Common models:

Erdős-Rényi & Gilbert models (no clusters)

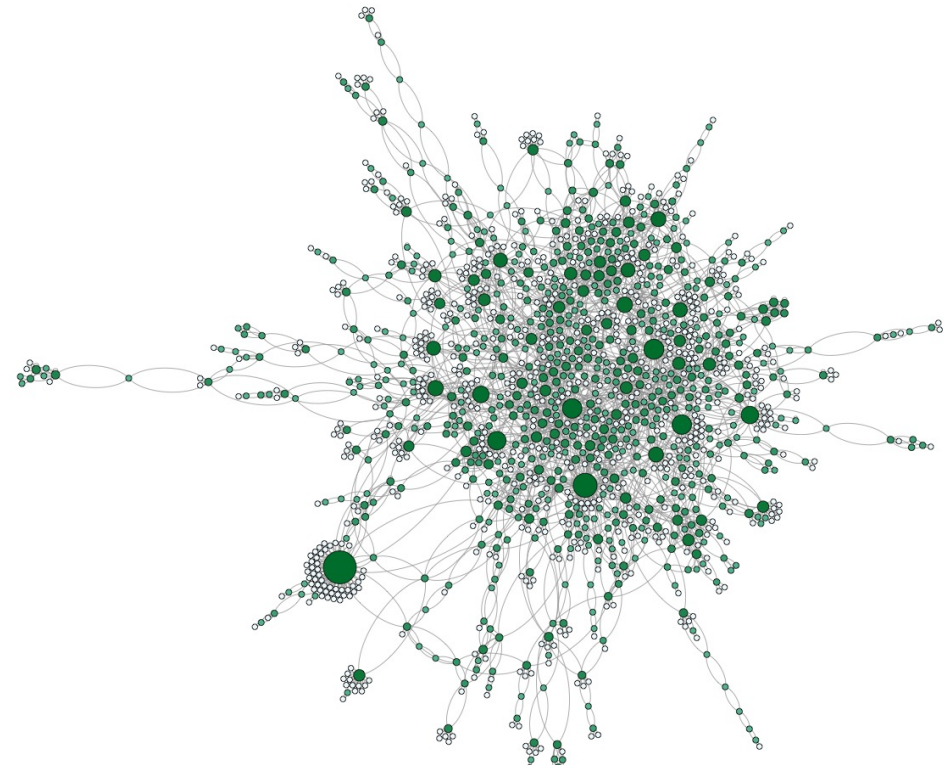
The Watts-Strogatz model (no hubs)

The Barabási-Albert Model (hubs, but  
always old)



**Republican and Democrat Retweet Network**

Image source: Slides for Slides for Menczer, Fortunato, Davis, *A First Course in Network Science*



**Network of Protein interaction of yeast.**

Image source: Slides for Slides for Menczer, Fortunato, Davis, *A First Course in Network Science*



# Seminar Readings for Today

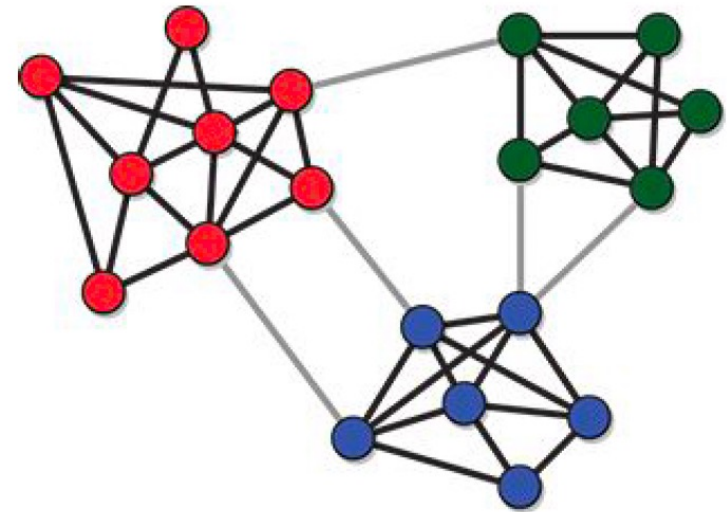
- ▶ Examples of network modelling & simulations with focus on Social Sciences
  - ▶ Muhammad & Kasahara, 'Agent-based simulation of fake news dissemination: the role of trust assessment and big five personality traits on news spreading', *Social Network Analysis and Mining* (2024), 14:75. DOI: 10.1007/s13278-024-01235-8.
  - ▶ Bianchi & Squazzoni, 'Agent-based models in sociology', *WIREs Comput Stat* (2015), 7:284–306. DOI: 10.1002/wics.1356.
  - ▶ van Woudenberg et al, 'Identifying Influence Agents That Promote Physical Activity Through the Simulation of Social Network Interventions: Agent-Based Modeling Study', *J Med Internet Res* (2019), 21(8):e12914. DOI: 10.2196/12914.

# Clusters, communities, modules

- ▶ A **community (cluster, module)** is a group of graph vertices that “belong together” according to some precisely defined criteria which can be measured.
- ▶ It is commonly defined as a group of vertices such that the **density of edges between vertices of the group is higher than the average edge density in the graph.**

- ▶ Definition based on: *Computational Complexity* ed by Meyers (2012).

Katarzyna Anna Kapitan, *Network Analysis for Humanists*,  
Paris 2025

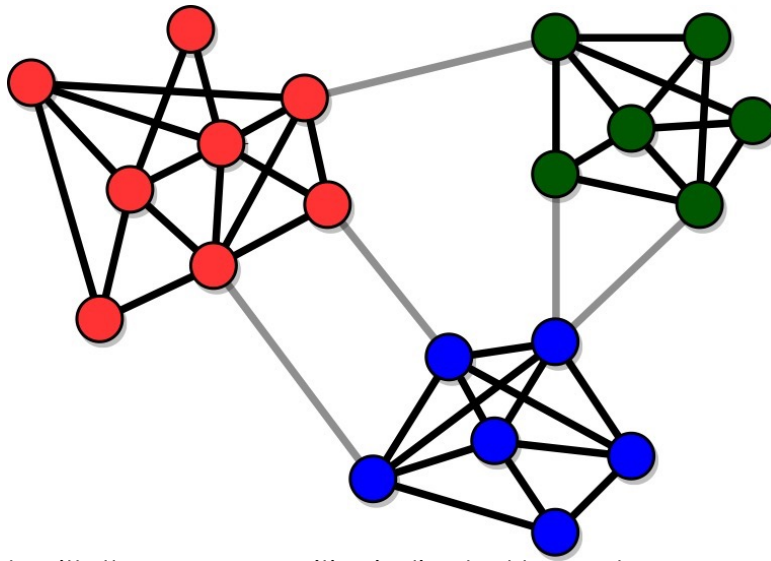


A graph with three communities indicated by node colours. Image source: Menczer, *A First Course...*, Fig. 6.1

# Why study communities?

- ▶ To uncover the organization of the network
- ▶ To classify the nodes based on their position
- ▶ To uncover relationships between features of nodes and their position in the network
- ▶ To infer missing links

# Basic Definitions: Communities



- ▶ **High cohesion:** communities have many internal links, so their nodes stick together
- ▶ **High separation:** communities are connected to each other by few links

A graph with three communities indicated by node colours. Image source: Menczer, A First Course..., Fig. 6.1

Katarzyna Anna Kapitan, Network Analysis for Humanists,  
Paris 2025

# Cohesion (Internal Connectivity)

- ▶ Measures how strongly connected a group of nodes is.
  - ▶ High cohesion means nodes within a group have dense intra-community links.
  - ▶ Important for community detection, resilience, and influence propagation.
- ▶ Examples:
  - ▶ Social circles in a social network.
  - ▶ Functionally similar proteins in biological networks.



# Separation (External Connectivity)

- ▶ Measures how isolated a community is from the rest of the network.
  - ▶ High separation means few inter-community links.
  - ▶ Helps detect borders, bottlenecks, and inter-group influence.
- ▶ Examples:
  - ▶ Independent research domains in citation networks.
  - ▶ Different political or opinion networks.

# Cohesion & Separation

- ▶ Communities should have **high cohesion** and **high separation**
- ▶ *How to measure that?*



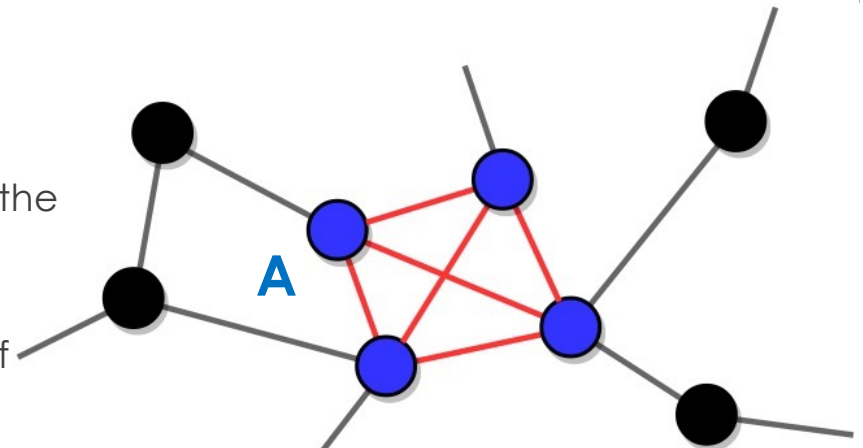
# Cohesion & Separation

- ▶ Communities should have **high cohesion** and **high separation**
- ▶ *How to measure that?*
  - ▶ Check whether the number of internal links exceeds the number of external links

# Basic Definitions: Variables in Communities

- ▶ **Internal degree of a node:** number of neighbours of the node in its community
- ▶ **External degree of a node:** number of neighbours of the node outside of its community
- ▶ **Community degree:** sum of the degrees of the nodes in the community
- ▶ **Internal link density:** ratio between the number of links inside a community and the maximal possible number of links that can lie inside this community.

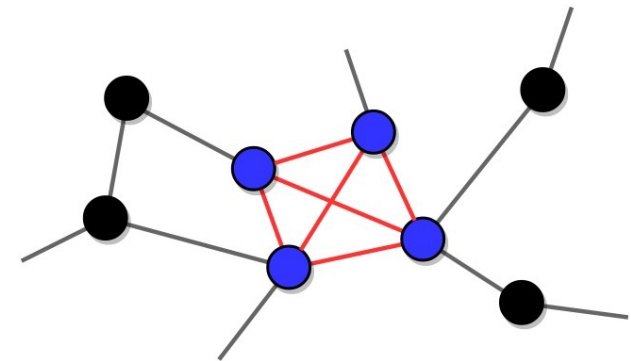
Calculate the respective values for node A  
and the community to which it belongs  
(blue nodes)



# Strong & Weak Communities

- ▶ **Strong community:** subnetwork such that the internal degree of each node is greater than its external degree
- ▶ **Weak community:** subnetwork such that the sum of the internal degrees of its nodes is greater than the sum of their external degrees

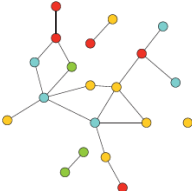
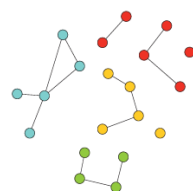
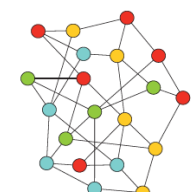
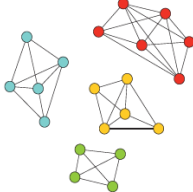
Is the blue community a strong or weak community?





# Balance Between Cohesion & Separation

- Cohesion is especially important in sociology
  - Too much cohesion with little separation → overlapping clusters.
  - Too much separation with little cohesion → fragmented communities.

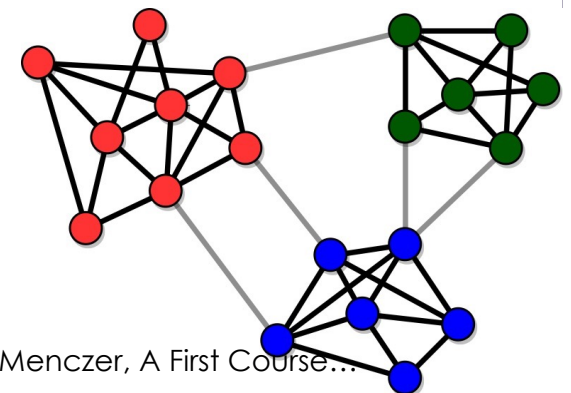
		Nominal Group Boundary	
		Porous/weak	Impermeable/strong
Social Connectivity	Low	Undifferentiated, anomic social structure 	Nominal interest group structure 
	High	Socially cohesive, integrated 	Segmented society, highly modular 

**Image source:** Rawlings CM, Smith JA, Moody J, McFarland DA. Cohesion and Groups. In: *Network Analysis: Integrating Social Network Theory, Method, and Application with R*. Structural Analysis in the Social Sciences. Cambridge University Press; 2023:161-189.

# Grouping nodes of a network

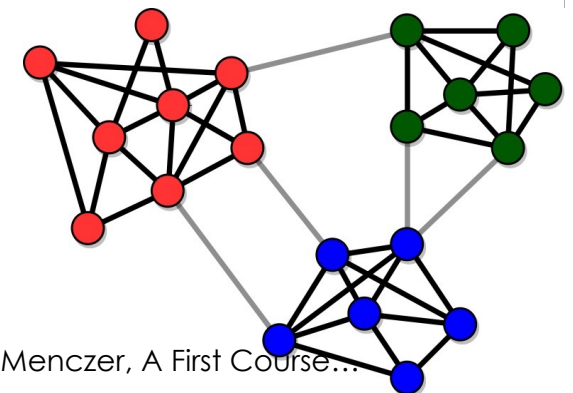
# Network Partitioning

- ▶ **Partitioning** - dividing the nodes of a network into a number of groups such that the number of links between the groups is minimal
- ▶ In other words: a graph **partition** is the reduction of a graph to a smaller graph by partitioning its set of nodes into **mutually exclusive groups**.
- ▶ How many different partitions you can distinguish in the network illustrated on this slide?



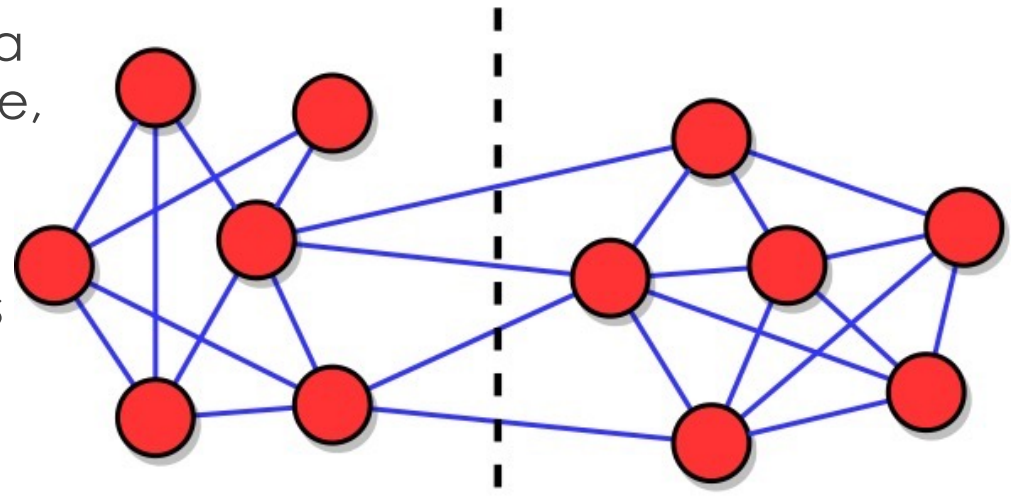
# Network Partitioning

- ▶ In Network Science there are two partitions which are referred to as the **trivial partitions**:
  - ▶ 1. The partition with one set containing every node (**Singleton partition**).
  - ▶ 2. The partition with  $N$  sets, each containing a single node (**Partition of singletons**).
- ▶ A valid partition thus contains between 1 and  $N$  sets.



# Network Partitioning

- ▶ **Bisectioning** - dividing the nodes of a network into two groups of equal size, such that the number of links between the groups is minimal
- ▶ The number of links between groups is called **cut size**



**What is the cut size of the network on the slide?**



# Kernighan-Lin algorithm

- ▶ A popular algorithm used for graph bisection is **Kernighan-Lin algorithm**
- ▶ The idea behind this algorithm is simple:
  - ▶ Given an initial bisection of the network, we swap pairs of nodes between clusters, such as to obtain the greatest decrease of the cut size, while the size of the clusters does not change.
- ▶ **NetworkX :**
  - ▶ `kernighan_lin_bisection(G, partition=None, max_iter=10, weight='weight', seed=None)`.

Source: [https://networkx.org/documentation/stable/reference/algorithms/generated/networkx.algorithms.community.kernighan\\_lin.kernighan\\_lin\\_bisection.html](https://networkx.org/documentation/stable/reference/algorithms/generated/networkx.algorithms.community.kernighan_lin.kernighan_lin_bisection.html)

# Kernighan-Lin algorithm

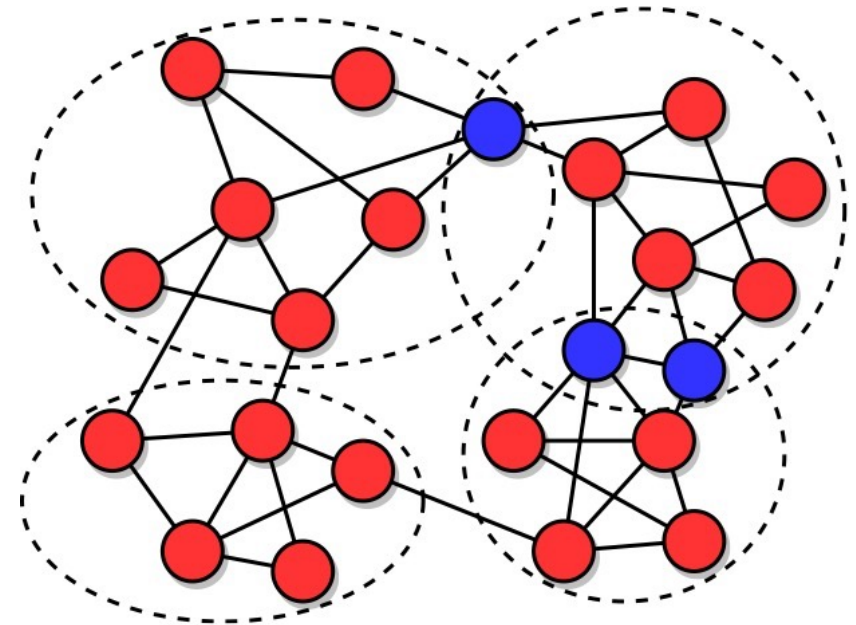
- ▶ Take a graph of  $N$  nodes and create arbitrary **Partition  $P$**  consisting of two **clusters  $A$  and  $B$**  by randomly assigning nodes to either group
- ▶ For each pair of **nodes  $i, j$ , with  $i \in A$  and  $j \in B$** , compute the variation in **cut size** between the current partition and the one obtained by swapping  $i$  and  $j$ .
- ▶ The pair of nodes  $i$  and  $j$  yielding the largest decrease in cut size is selected and swapped. This pair of nodes is locked; they will not be touched again during this iteration
- ▶ Repeat steps 2 and 3 until no more swaps of unlocked nodes yield a decrease in cut size. This yields a new bipartition, that is used as a starting configuration for the next iteration
- ▶ The procedure ends when the cut size of partitions obtained after consecutive iterations is the same, meaning that the algorithm is unable to improve the result.

# Kernighan-Lin algorithm

- ▶ The described procedure is **greedy**, in that at each step one looks for the partition with the smallest cut size. Because of that, the algorithm gets stuck in **local optima**, i.e., solutions whose cut size is not as low as it can be.
- ▶ Greedy is an algorithmic paradigm that builds up a solution piece by piece, always choosing the next piece that offers the most obvious and immediate benefit. Greedy algorithms are used for optimization problems
  - ▶ Further tutorials on greedy algorithms here:  
<https://www.geeksforgeeks.org/introduction-to-greedy-algorithm-data-structures-and-algorithm-tutorials/>

# Network Partitioning vs Community Detection

- ▶ A graph **partition** is the reduction of a graph to a smaller graph by partitioning its set of nodes into **mutually exclusive groups**.
- ▶ For partitioning, clusters have to be **well-separated**, but they **do not need to have high internal link density**.
- ▶ (Usually) clusters found via graph partitioning are not communities.



# Community Detection

- ▶ There are many different methods for community detection with many different concepts involved
  - ▶ For more info see Chapter 6 in Menczer & Chapter 11 in Newman.
- ▶ Today we focus only two concepts:
  - ▶ Bridge removal
  - ▶ Modularity optimisation
- ▶ Bibliography:
  - ▶ **Menczer**, Fortunato, Davis (2020), *A First Course in Network Science* (Cambridge: Cambridge University Press).
  - ▶ **Newman** (2010), *Networks: An Introduction* (Oxford: Oxford University Press)



# Bridge Removal

- ▶ Bridge Removal is the idea behind many community detection methods.
- ▶ Strictly speaking, a **bridge** is a link whose removal breaks a connected network into two parts. For this lesson, let's call **every link joining two communities a bridge**
- ▶ 'If we were able **to locate all bridges**, we would have a natural way to detect the clusters' => clusters would be the connected components after bridge removal
- ▶ **But how to mathematically identify bridges?**

Katarzyna Anna Kapitan, Network Analysis for Humanists,  
Paris 2025

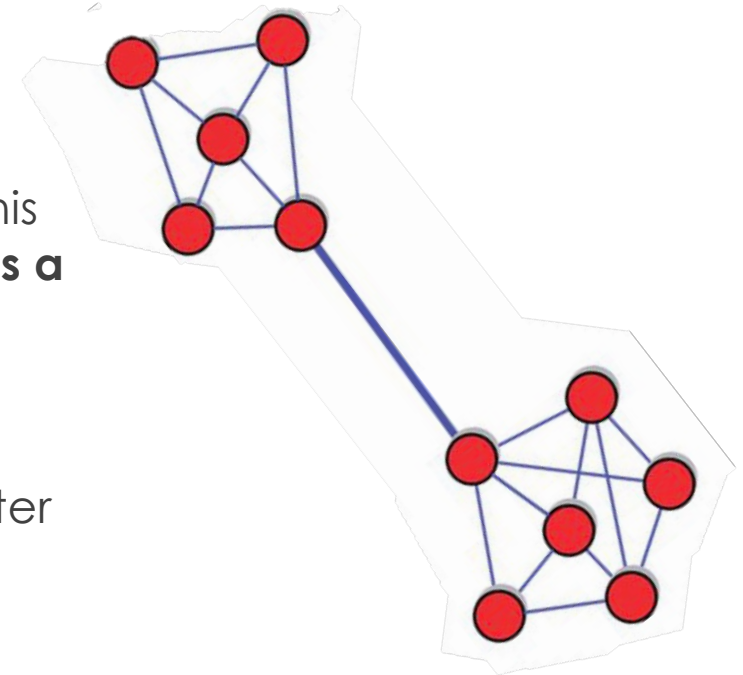


Image source: Menczer, A First Course...

# Bridge Removal

- ▶ Use **link betweenness** to identify bridges:
  - ▶ ‘**Bridges** are expected to have **large betweenness** values because shortest paths between nodes in different communities run through bridges’
  - ▶ ‘**Internal links** are expected to have **comparatively lower betweenness** values, because there are many alternative routes going from one node of the community to another, due to the high density of links inside the cluster’
- ▶ Link betweenness is used, for example, in **Girvan-Newman algorithm**

# Girvan-Newman Algorithm: Procedure

- ▶ Start by computing the betweenness for all links
- ▶ Each iteration consists of two steps:
  - ▶ Remove the link with largest betweenness (in case of ties, one of them is picked at random)
  - ▶ Recalculate the betweenness of the remaining links
- ▶ The algorithm stops when all links are removed, so all the nodes are isolated

# Girvan-Newman Algorithm: Limits

- ▶ The recalculation of the betweenness at each iteration is necessary, but it makes the algorithm slow
  - ▶ not practical for large networks with, let's say, more than 10,000 nodes.
- ▶ The method delivers a full hierarchy of  $N$  partitions, from the one consisting of a single partition covering the entire network (**Singleton partition**) to the one where each node is in its own partition (**Partition of singletons**).
- ▶ Which ones with elements between 1 and  $N$  are meaningful communities?
- ▶ **How can we select best partitions that allow us to identify communities?**



Question: How can we evaluate how good a partition is?



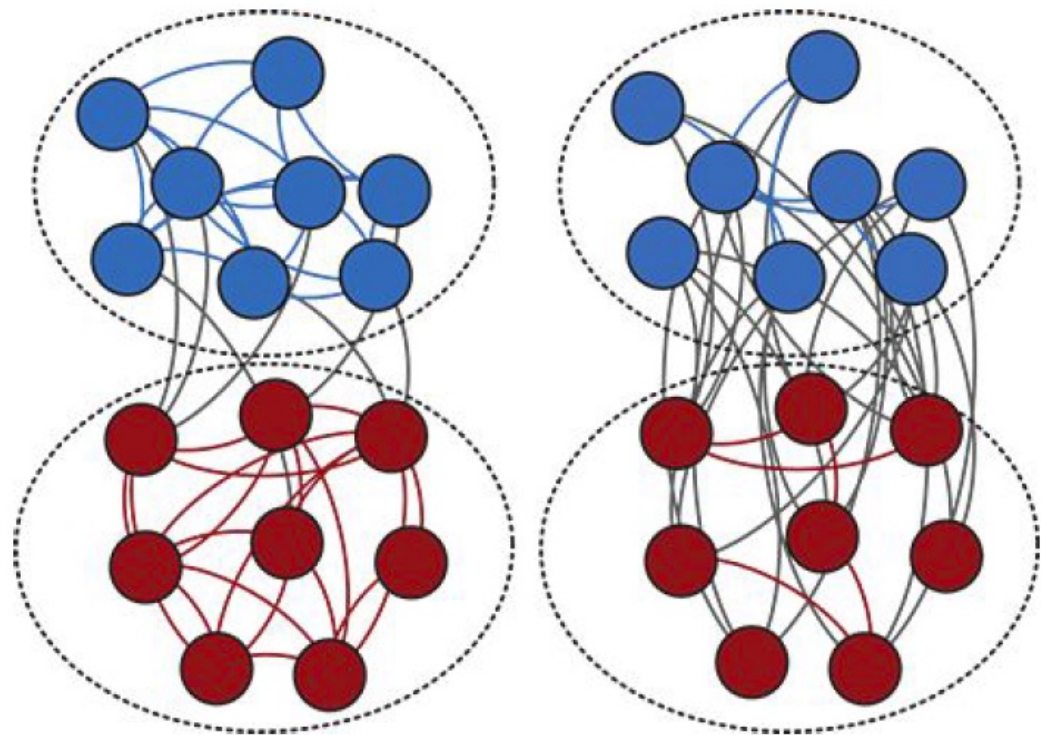
Answer: Modularity Optimisation



# Modularity

- ▶ 'The **modularity of a partition evaluates the communities** not in absolute terms, but **with respect to a random baseline**. It does so by discounting the internal links that could be attributed to a randomized version of the original network.'
- ▶ Baseline: randomized versions of the original network, preserving its degree sequence (See Week 6 slides and Chapter 5 in Menczer)
- ▶ For each community of a partition, modularity computes the difference between the number of internal links in the community and the expected value of this number in the set of randomized networks

- If the **network is random** the **modularity** of any partition is supposed to be **low**, because the number of internal links of any cluster of the partition should be close to the expected value in the randomized networks
- If the **number of links within the clusters** is much larger than its expected random value **modularity can reach high values**



Network modularity (left – real, right – the same but randomized)

**Image source:** Menczer (2020), based on Fortunato and Hric (2016).

# Modularity

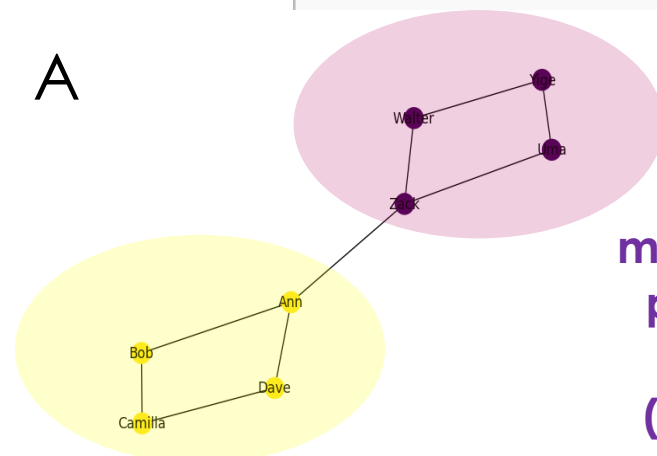
Higher modularity ( $\sim 0.3$  to  $0.7$ )  $\rightarrow$  Stronger community structure.

Lower modularity ( $\sim 0.0$  to  $0.1$ )  $\rightarrow$  Weak or no community structure.

Negative modularity  $\rightarrow$  Worse than random clustering.

$$Q = \frac{1}{L} \sum_C \left( L_C - \frac{k_C^2}{4L} \right), \quad (6.4)$$

where the sum runs over all clusters of the partition,  $L_C$  is the number of internal links in cluster  $C$ ,  $k_C$  is the total degree of the nodes in  $C$  [Eq. (6.2)], and  $L$  is the number of links in the network.



Calculate the  
modularity for these two  
partitions of the same  
network  
(with 2 and 3 clusters)

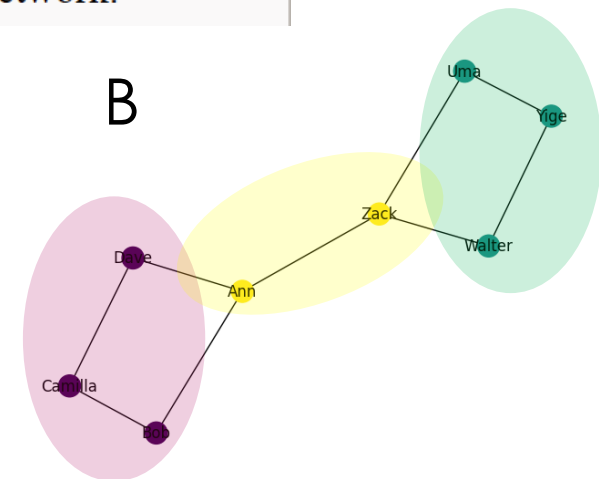


Image source: Menczer, A First Course...

# Modularity

- ▶ How to find best partition, for example, from a list generated by Girvan-Newman?
  - ▶ Search for the partition with the largest possible modularity => **Modularity Maximisation**
  - ▶ The modularity maximisation method explores a hierarchy of partitions, with the initial partition into singletons which has negative modularity. The modularity increases steadily until a positive peak is reached, and finally it goes down until it hits zero when all nodes are in the same community. The partition corresponding to the peak value is the best solution found by the algorithm.



# NetworkX

```
greedy_modularity_communities(G, weight=None, resolution=1, cutoff  
                               =1, best_n=None)
```

This function uses the Clauset-Newman-Moore greedy modularity maximization to find the community partition with the largest modularity

Source:[https://networkx.org/documentation/stable/reference/algorithms/generated/networkx.algorithms.community.modularity\\_max.greedy\\_modularity\\_communities.html#rce363827c0a4-2](https://networkx.org/documentation/stable/reference/algorithms/generated/networkx.algorithms.community.modularity_max.greedy_modularity_communities.html#rce363827c0a4-2)

Clauset, A., Newman, M. E., & Moore, C. 'Finding community structure in very large networks', Physical Review E 70(6), 2004.



# NetworkX

```
modularity(G, communities, weight='weight')
```

- ▶ This function returns the modularity of the given partition of the graph.
- ▶ Source: <https://networkx.org/documentation/networkx-2.5/reference/algorithms/generated/networkx.algorithms.community.quality.modularity.html>



Katarzyna Anna Kapitan, Network Analysis for Humanists,  
Paris 2025

Lab