Advanced Algorithms and Computational Models (module A)

Communities

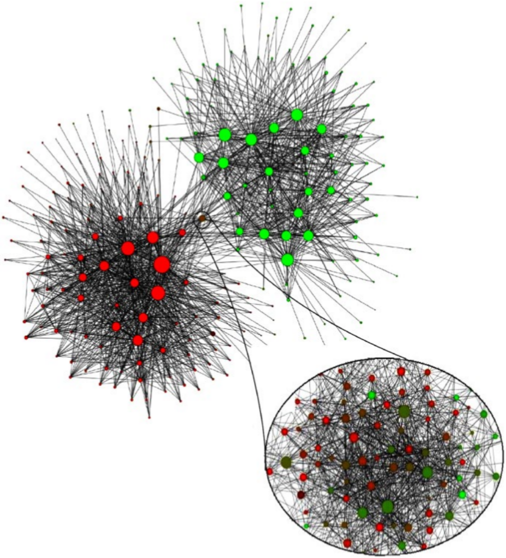
Giacomo Fiumara
giacomo.fiumara@unime.it

Introduction

- Belgium appears to be the model bicultural society: 59% of its citizens are Flemish, speaking Dutch and 40% are Walloons who speak French
- As multiethnic countries break up all over the world, we must ask: How did this country foster the peaceful coexistence of these two ethnic groups since 1830?
- Is Belgium a densely knitted society, where it does not matter if one is Flemish or Walloon?
- Or we have two nations within the same borders, that learned to minimize contact with each other?

Introduction

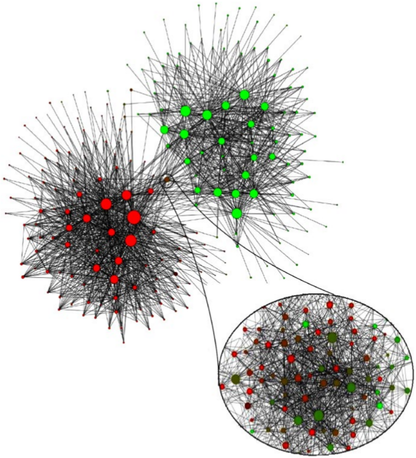
Communities in Belgium



Introduction

Definition

A **community** is a group of nodes that have a higher likelihood of connecting to each other than to nodes from other communities



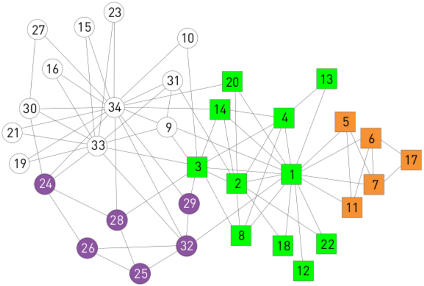
Introduction

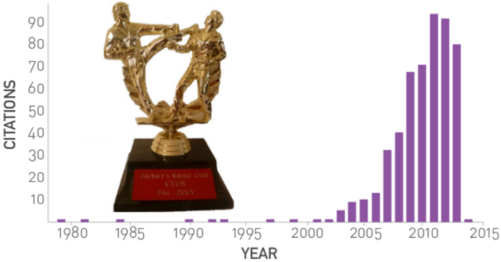
An example: social networks

- Social networks easily contain communities
- For example, the employees of a company are more likely to interact with their coworkers than with employees of other companies
- Therefore, work places appear as densely interconnected communities within the social network
- A social network that has received particular attention in the context of community detection is known as Zachary's Karate Club, capturing the links between 34 members of a karate club. The interest in the dataset is driven by a singular event: a conflict between the president and the instructor split the club into two

Introduction

Zachary's Karate Club





Introduction

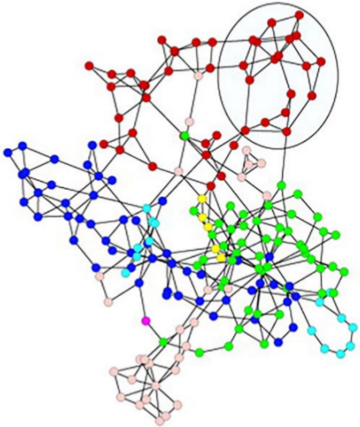
An example: biological networks

- Communities play a fundamental role in our understanding of how specific biological functions are encoded in cellular networks

- For example, proteins that are involved in the same disease tend to interact with each other: each disease can be linked to a well-defined neighbourhood of the cellular network

Introduction

A metabolic network



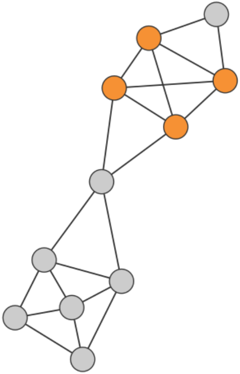
Basics of Communities

- One of the first studies on community structure defined a community as group of individuals whose members all know each other
- In graph theoretic terms this means that a community is a **complete subgraph** or a **clique**
- This is wrong: triangles are frequent in networks, while larger cliques are rare
- Moreover, requiring that a community to be a complete subgraph may be too restrictive

Basics of Communities

Cliques

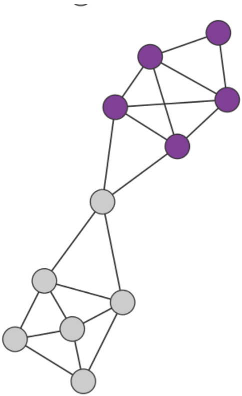
A *clique* corresponds to a complete subgraph. The highest order clique of this network is a square



Basics of Communities

Strong communities

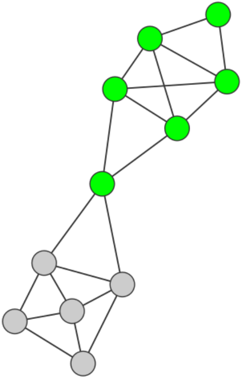
A *strong community* is a connected subgraph whose nodes have more links to other nodes in the same community that to nodes that belong to other communities



Basics of Communities

Weak communities

A *weak community* is a connected subgraph whose nodes' internal degree exceeds their external degree



Basics of Communities

Number of communities

- How many ways the nodes of a network can be grouped into communities?
- An approximate answer is provided by the simplest community finding problem, called *graph bisection*
- The idea consists in dividing a network into two non-overlapping subgraphs such that the number of links between the nodes in the two groups (*cut size*) is minimized
- The graph bisection problem can be solved by inspecting all possible divisions into two groups and choosing the one with the smallest cut size

Basics of Communities

Number of communities

- The computational cost of this approach is roughly given by the number of distinct ways we can partition a network of N nodes into groups N_1 and N_2 nodes is

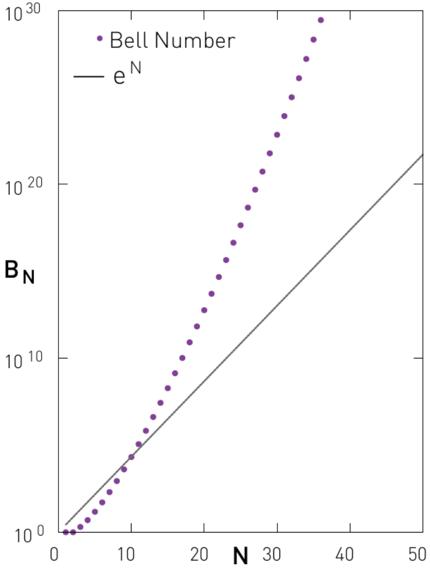
$$\frac{N!}{N_1!N_2!}$$

- In the (very simple) case of a network with 10 nodes which is bisected into two subgraphs of $N_1 = N_2 = 5$, 252 bisections must be checked to find the one with the smallest cut size

- In the (simple) case of a network with 100 nodes which is bisected into two subgraphs of $N_1 = N_2 = 50$, 10^{29} bisections must be checked

Basics of Communities

Number of communities



Hierarchical Clustering

Introduction

- Hierchical clustering is used to unveil the community structure of large networks in polynomial time
- Its starting point is a *similarity matrix*, whose elements x_{ij} express the distance of node i from node j
- Similarity matrix is then used to iteratively identify groups of nodes with high similarity
- Two procedures can be used: *agglomerative algorithms*, which merge nodes with high similarity into the same community, while *divisive algorithms* isolate communities by removing low similarity links that tend to connect communities
- Both procedures produce a *dendrogram*, a hierchical tree that helps in predicting possible communities

Hierarchical Clustering

Agglomerative procedures: the Ravasz algorithm

- Define the similarity matrix
- Decide group similarity
- Apply hierarchical clustering
- Dendrogram

Hierarchical Clustering

The Ravasz algorithm: define the similarity matrix

- Similarity should be high for node pairs belonging to the same community and low for node pairs belonging to different communities
- Nodes that connect to each other and share neighbors likely belong to the same community, hence their x_{ij} should be large
- The topological overlap matrix is defined as

$$x_{ij}^0 = \frac{J(i,j)}{min(k_i, k_j) + 1 - \Theta(A_{ij})}$$

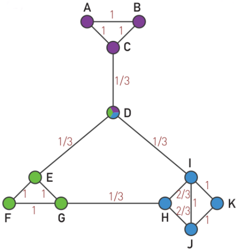
- $J(i,j)$ is the number of common neighbors of nodes i and j
- $\Theta(x)$ is the Heaviside step function, zero for $x \leq 0$ and one for $x > 0$
- $min(k_i, k_j)$ is the smaller of the degrees k_i and k_j

Hierarchical Clustering

The Ravasz algorithm: define the similarity matrix

$$x_{ij}^0 = \frac{J(i,j)}{min(k_i, k_j) + 1 - \Theta(A_{ij})}$$

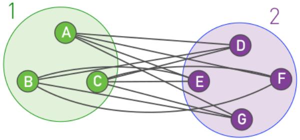
- $x_{ij}^0 = 1$ if nodes i and j are linked and share the same neighbors (nodes A and B)
- $x_{ij}^0 = 0$ if nodes i and j do not have common neighbors
- Members of the same dense local network neighborhood have high topological overlap



Hierarchical Clustering

The Ravasz algorithm: decide group similarity

- As nodes are merged into small communities, the similarity of two communities must be measured
- The Ravasz algorithms used the *average cluster similarity*
- The similarity of two communities is defined as the average of x_{ij} over all node pairs i and j that belong to distinct communities



Hierarchical Clustering

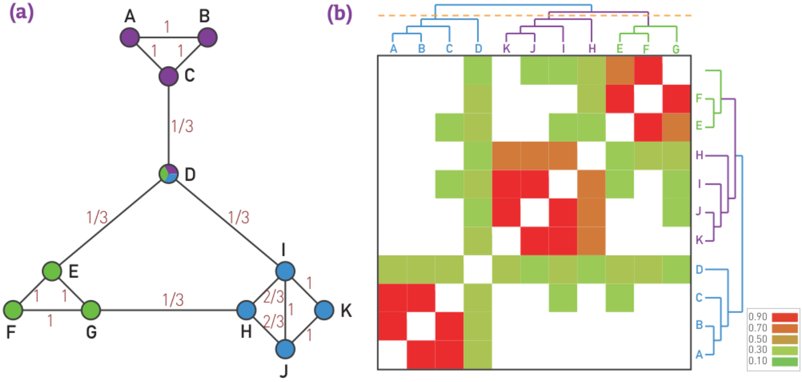
The Ravasz algorithm: apply hierchical clustering

The Ravasz algorithm uses the following procedure to identify the communities:

1. Assign each node to a community of its own and evaluate x_{ij} for all node pairs
2. Find the community pair or the node pair with the highest similarity and merge them into a single community
3. Calculate the similarity between the new community and all other communities
4. Repeat Steps 2 and 3 until all nodes form a single community

Hierarchical Clustering

The Ravasz algorithm: Dendrogram



Hierarchical Clustering
Divisive procedures: the Girvan-Newman algorithm

Divisive procedures systematically remove the links connecting nodes that belong to different communities, eventually breaking a network into isolated communities

- Define centrality
- Hierarchical clustering

Hierarchical Clustering

The Girvan-Newman algorithm: define centrality

- In divisive algorithms x_{ij}, called *centrality*, selects node pairs that are in different communities
- Therefore x_{ij} must be high (or low) if nodes i and j belong to different communities and small if they are in the same community
- The most widely used centrality (in this context) is the *link betweenness*, defining x_{ij} as the number of shortest paths that go through the link (i, j)
- Links connecting different communities are expected to have large x_{ij}, while links within a community have small x_{ij}

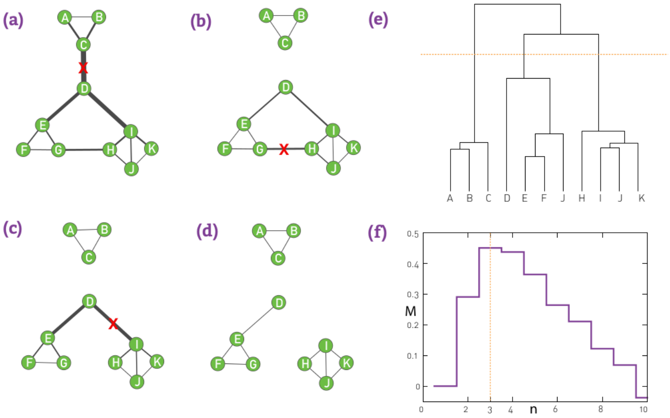
Hierarchical Clustering

The Girvan-Newman algorithm: Hierarchical clustering

1. Compute the centrality x_{ij} of each link
2. Remove the link with the highest centrality. In case of a tie, choose one link randomly
3. Recalculate the centrality of each link for the altered network
4. Repeat Steps 2 and 3 until all links are removed

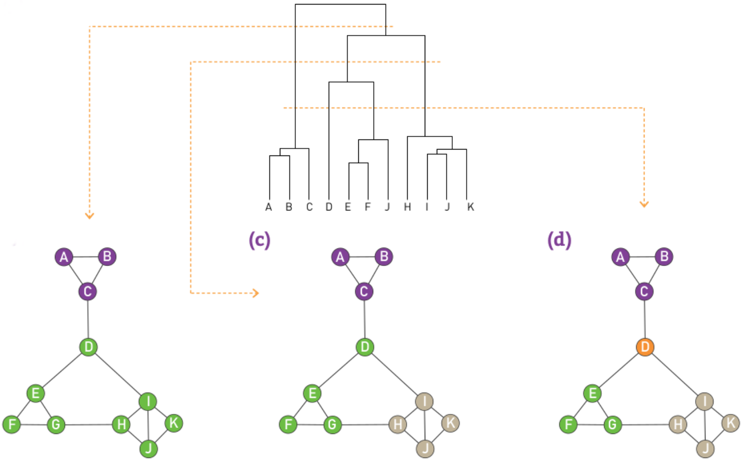
Hierarchical Clustering

The Girvan-Newman algorithm: Hierarchical clustering



Hierarchical Clustering

Ambiguity



Modularity
Introduction

- Consider a network with N nodes, L links
- Consider also a partition into n_c communities
- Each partition has N_c nodes connected to each other by L_c links
- If L_c is larger than the **expected** number of links between the N_c nodes, then the nodes of the subgraph C_c could be part of a true community

Modularity
Introduction

- We therefore measure

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

  which expresses the difference between the real number of links and the **expected** number of links if the subgraph were randomly wired

- p_{ij} can be estimated in the case of random networks, in which

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

Modularity

Introduction

- The expression

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

can then be simplified as

$$M_c = \frac{L_c}{L} - \left(\frac{k_c}{2L}\right)^2$$

Modularity

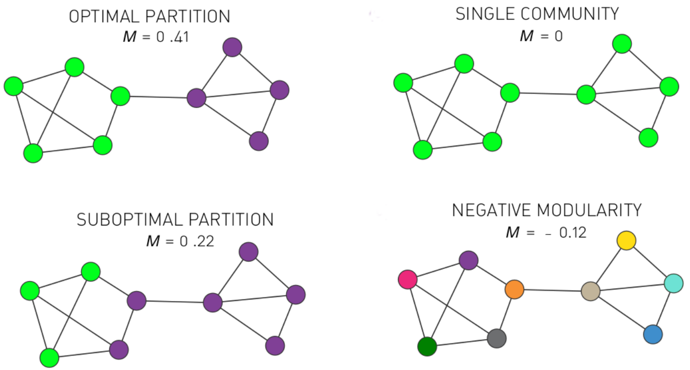
Introduction

- This concept can be generalized to a full network
- Consider the complete partition that breaks the network into n_c communities
- Then

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

- which expresses the difference between the local link density and the expected link density summed over all communities

Modularity



OPTIMAL PARTITION
M = 0 .41
SINGLE COMMUNITY
M = 0
SUBOPTIMAL PARTITION
M = 0 .22
NEGATIVE MODULARITY
M = - 0.12

Modularity

The greedy algorithm

- Partitions with higher modularity correspond to partitions that more accurately capture the communities
- Therefore it is reasonable to conclude that it the partition with maximum modularity corresponds to the optimal community structure
- The greedy algorithm (Newman) finds partitions with close to maximal M

Modularity

The greedy algorithm

1. Assign each node to a community of its own, starting with N communities of single nodes
2. Inspect each community pair connected by at least one link and compute the modularity difference ΔM obtained if we merge them. Identify the community pair for which ΔM is the largest and merge them. Note that modularity is always calculated for the full network
3. Repeat Step 2 until all nodes merge into a single community, recording M for each step
4. Select the partition for which M is maximal

Modularity

The greedy algorithm