A First Course in Network Theory
Community Detection: a Partial Overview

Luce le Gorrec, Philip Knight, Francesca Arrigo

University of Strathclyde, Glasgow

Community detection VS (Spectral) Graph Partitioning

- (Spectral) Graph Partitioning: For a specific application: parallel or distributed computations, building VLSI, etc.

$\Longrightarrow$ Known target, driven by the application.

Community detection VS (Spectral) Graph Partitioning

- (Spectral) Graph Partitioning: For a specific application: parallel or distributed computations, building VLSI, etc.

$\Longrightarrow$ Known target, driven by the application.

- Community detection: For data analysis: finding **groups of similar nodes** (typically, consistent groups in social networks).

$\Longrightarrow$ Unknown target, driven by data.

Community detection VS (Spectral) Graph Partitioning

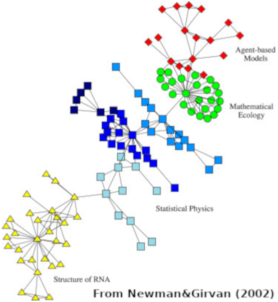
- (Spectral) Graph Partitioning: For a specific application: parallel or distributed computations, building VLSI, etc.

$\Longrightarrow$ Known target, driven by the application.

- Community detection: For data analysis: finding **groups of similar nodes** (typically, consistent groups in social networks).

$\Longrightarrow$ Unknown target, driven by data.

Examples



From Newman&Girvan (2002)

Community detection VS (Spectral) Graph Partitioning

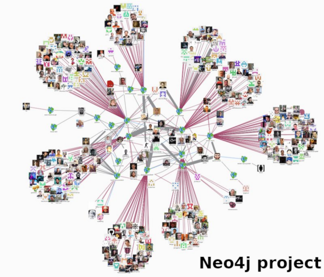
- (Spectral) Graph Partitioning: For a specific application: parallel or distributed computations, building VLSI, etc.

$\Longrightarrow$ Known target, driven by the application.

- Community detection: For data analysis: finding **groups of similar nodes** (typically, consistent groups in social networks).

$\Longrightarrow$ Unknown target, driven by data.

Examples



Community detection VS (Spectral) Graph Partitioning

- (Spectral) Graph Partitioning: For a specific application: parallel or distributed computations, building VLSI, etc.

⟹ Known target, driven by the application.

- Community detection: For data analysis: finding **groups of similar nodes** (typically, consistent groups in social networks).

⟹ Unknown target, driven by data.

Examples

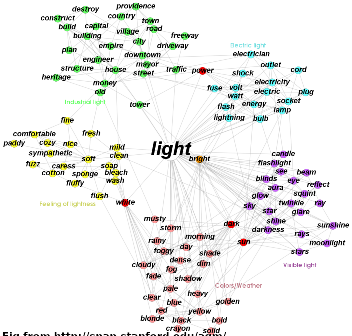


Fig from http://snap.stanford.edu/agm/

Community detection VS (Spectral) Graph Partitioning

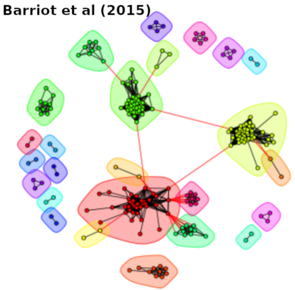
- (Spectral) Graph Partitioning: For a specific application: parallel or distributed computations, building VLSI, etc.

$\Longrightarrow$ Known target, driven by the application.

- Community detection: For data analysis: finding **groups of similar nodes** (typically, consistent groups in social networks).

$\Longrightarrow$ Unknown target, driven by data.

Examples



Community detection VS (Spectral) Graph Partitioning

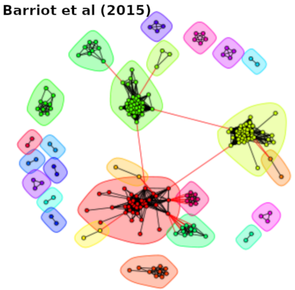
- (Spectral) Graph Partitioning: For a specific application: parallel or distributed computations, building VLSI, etc.

$\Longrightarrow$ Known target, driven by the application.

- Community detection: For data analysis: finding **groups of similar nodes** (typically, consistent groups in social networks).

$\Longrightarrow$ Unknown target, driven by data.

Examples



$\Longrightarrow$ A community $\approx$ a group of densely connected nodes, loosely connected with the rest of the network.

NB In the following, $G = (V, E, \omega)$ is an undirected graph. Community structure is a partitioning of V denoted $\mathcal{C} = \{C_1, ..., C_k\}$.

Attempts for formal definitions (Raddichi et al. 2004)

Given $G = (V, E)$ some **simple** graph, $S \subset V$ and $\overline{S} = V \setminus S$, we say that

- S is a **k-core** of G, with $k \in \mathbb{N}$, if

$$\forall u \in S, |\mathcal{N}(u) \cap S| \geq k$$

- S is a α-**clique** of G, with $\alpha \in]0, 1]$, if

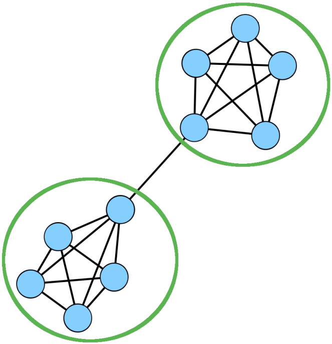
$$2 \times |E \cap S \times S| / (|S| \times (|S| - 1)) \geq \alpha$$

- S is a **Strong Community** of G if

$$\forall u \in S, |\mathcal{N}(u) \cap S| > |\mathcal{N}(u) \cap \overline{S}|.$$

- S is a **Weak Community** of G if

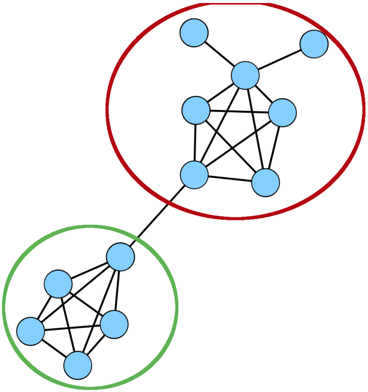
$$\sum_{u \in S} |\mathcal{N}(u) \cap S| > \sum_{u \in S} |\mathcal{N}(u) \cap \overline{S}|$$



Attempts for formal definitions (Raddichi et al. 2004)

Given $G = (V, E)$ some **simple** graph, $S \subset V$ and $\overline{S} = V \setminus S$, we say that

- S is a **k-core** of G, with $k \in \mathbb{N}$, if
$$\forall u \in S, |\mathcal{N}(u) \cap S| \geq k$$
- S is a α-**clique** of G, with $\alpha \in]0, 1]$, if
$$2 \times |E \cap S \times S| / (|S| \times (|S| - 1)) \geq \alpha$$
- S is a **Strong Community** of G if
$$\forall u \in S, |\mathcal{N}(u) \cap S| > |\mathcal{N}(u) \cap \overline{S}|.$$
- S is a **Weak Community** of G if
$$\sum_{u \in S} |\mathcal{N}(u) \cap S| > \sum_{u \in S} |\mathcal{N}(u) \cap \overline{S}|$$

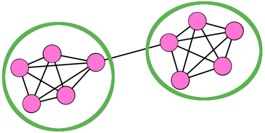


Attempts for formal definitions (Raddichi et al. 2004)

Given $G = (V, E)$ some **simple** graph, $S \subset V$ and $\overline{S} = V \setminus S$, we say that

- S is a **k-core** of G, with $k \in \mathbb{N}$, if
$$\forall u \in S, |\mathcal{N}(u) \cap S| \geq k$$

- S is a α-**clique** of G, with $\alpha \in]0, 1]$, if
$$2 \times |E \cap S \times S| / (|S| \times (|S| - 1)) \geq \alpha$$



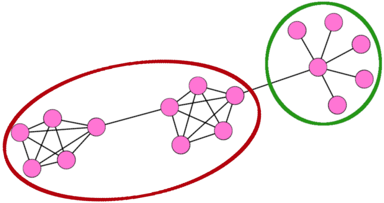
- S is a **Strong Community** of G if
$$\forall u \in S, |\mathcal{N}(u) \cap S| > |\mathcal{N}(u) \cap \overline{S}|.$$

- S is a **Weak Community** of G if
$$\sum_{u \in S} |\mathcal{N}(u) \cap S| > \sum_{u \in S} |\mathcal{N}(u) \cap \overline{S}|$$

Attempts for formal definitions (Raddichi et al. 2004)

Given $G = (V, E)$ some **simple** graph, $S \subset V$ and $\overline{S} = V \setminus S$, we say that

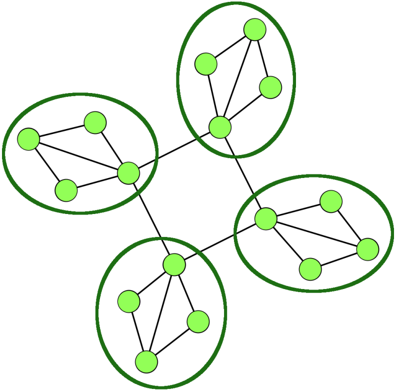
- S is a **k-core** of G, with $k \in \mathbb{N}$, if
$$\forall u \in S, |\mathcal{N}(u) \cap S| \geq k$$
- S is a α-**clique** of G, with $\alpha \in]0,1]$, if
$$2 \times |E \cap S \times S| / (|S| \times (|S| - 1)) \geq \alpha$$
- S is a **Strong Community** of G if
$$\forall u \in S, |\mathcal{N}(u) \cap S| > |\mathcal{N}(u) \cap \overline{S}|.$$
- S is a **Weak Community** of G if
$$\sum_{u \in S} |\mathcal{N}(u) \cap S| > \sum_{u \in S} |\mathcal{N}(u) \cap \overline{S}|$$



Attempts for formal definitions (Raddichi et al. 2004)

Given $G = (V, E)$ some **simple** graph, $S \subset V$ and $\overline{S} = V \setminus S$, we say that

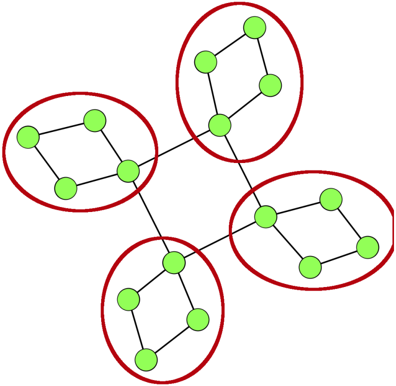
- S is a **k-core** of G, with $k \in \mathbb{N}$, if
$$\forall u \in S, |\mathcal{N}(u) \cap S| \geq k$$
- S is a α-**clique** of G, with $\alpha \in]0, 1]$, if
$$2 \times |E \cap S \times S| / (|S| \times (|S| - 1)) \geq \alpha$$
- S is a **Strong Community** of G if
$$\forall u \in S, |\mathcal{N}(u) \cap S| > |\mathcal{N}(u) \cap \overline{S}|.$$
- S is a **Weak Community** of G if
$$\sum_{u \in S} |\mathcal{N}(u) \cap S| > \sum_{u \in S} |\mathcal{N}(u) \cap \overline{S}|$$



Attempts for formal definitions (Raddichi et al. 2004)

Given $G = (V, E)$ some **simple** graph, $S \subset V$ and $\overline{S} = V \setminus S$, we say that

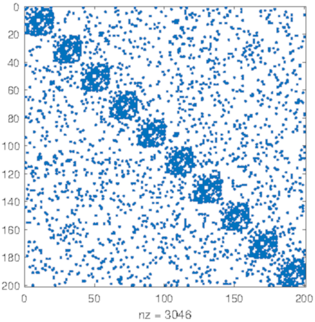
- S is a **k-core** of G, with $k \in \mathbb{N}$, if
$$\forall u \in S, |\mathcal{N}(u) \cap S| \geq k$$
- S is a α-**clique** of G, with $\alpha \in]0, 1]$, if
$$2 \times |E \cap S \times S| / (|S| \times (|S| - 1)) \geq \alpha$$
- S is a **Strong Community** of G if
$$\forall u \in S, |\mathcal{N}(u) \cap S| > |\mathcal{N}(u) \cap \overline{S}|.$$
- S is a **Weak Community** of G if
$$\sum_{u \in S} |\mathcal{N}(u) \cap S| > \sum_{u \in S} |\mathcal{N}(u) \cap \overline{S}|$$



Attempts for formal definitions (Raddichi et al. 2004)

Given $G = (V, E)$ some **simple** graph, $S \subset V$ and $\overline{S} = V \setminus S$, we say that

- S is a **k-core** of G, with $k \in \mathbb{N}$, if
  $$\forall u \in S, |\mathcal{N}(u) \cap S| \geq k$$
- S is a α-**clique** of G, with $\alpha \in]0, 1]$, if
  $$2 \times |E \cap S \times S| / (|S| \times (|S| - 1)) \geq \alpha$$
- S is a **Strong Community** of G if
  $$\forall u \in S, |\mathcal{N}(u) \cap S| > |\mathcal{N}(u) \cap \overline{S}|.$$
- S is a **Weak Community** of G if
  $$\sum_{u \in S} |\mathcal{N}(u) \cap S| > \sum_{u \in S} |\mathcal{N}(u) \cap \overline{S}|$$



Attempts for formal definitions (Raddichi et al. 2004)

Given $G = (V, E)$ some **simple** graph, $S \subset V$ and $\overline{S} = V \setminus S$, we say that

- S is a **k-core** of G, with $k \in \mathbb{N}$, if

$$\forall u \in S, |\mathcal{N}(u) \cap S| \geq k$$

- S is a α-**clique** of G, with $\alpha \in]0, 1]$, if

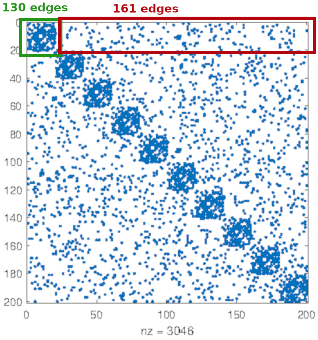
$$2 \times |E \cap S \times S| / (|S| \times (|S| - 1)) \geq \alpha$$

- S is a **Strong Community** of G if

$$\forall u \in S, |\mathcal{N}(u) \cap S| > |\mathcal{N}(u) \cap \overline{S}|.$$

- S is a **Weak Community** of G if

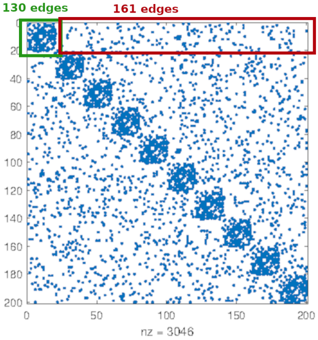
$$\sum_{u \in S} |\mathcal{N}(u) \cap S| > \sum_{u \in S} |\mathcal{N}(u) \cap \overline{S}|$$



Attempts for formal definitions (Raddichi et al. 2004)

Given $G = (V, E)$ some **simple** graph, $S \subset V$ and $\overline{S} = V \setminus S$, we say that

- S is a **k-core** of G, with $k \in \mathbb{N}$, if
$$\forall u \in S, |\mathcal{N}(u) \cap S| \geq k$$
- S is a α-**clique** of G, with $\alpha \in]0, 1]$, if
$$2 \times |E \cap S \times S| / (|S| \times (|S| - 1)) \geq \alpha$$
- S is a **Strong Community** of G if
$$\forall u \in S, |\mathcal{N}(u) \cap S| > |\mathcal{N}(u) \cap \overline{S}|.$$
- S is a **Weak Community** of G if
$$\sum_{u \in S} |\mathcal{N}(u) \cap S| > \sum_{u \in S} |\mathcal{N}(u) \cap \overline{S}|$$



$\Longrightarrow$ No definitive definition, more a thumb rule.

Unformal definitions – Screenshot from (Veldt2019)

"A community is often thought of as a set of nodes that has more connections between its members than to the remainder of the network." [Leskovec et al., 2008]

"Graph clustering is the task of grouping the vertices of the graph into clusters taking into consideration the edge structure of the graph in such a way that there should be many edges within each cluster and relatively few between the clusters." [Schaeffer, 2007]

"One mesoscopic structure, called a community, consists of a group of nodes that are relatively densely connected to each other but sparsely connected to other dense groups in the network." [Porter et al., 2009]

"Communities, or clusters, are usually groups of vertices having higher probability of being connected to each other than to members of other groups, though other patterns are possible." [Fortunato and Hric, 2016]

"The most basic task of community detection, or graph clustering, consists in partitioning the vertices of a graph into clusters that are more densely connected." [Abbe, 2018]

"Generally speaking, techniques for graph partitioning and graph clustering aim at the identification of vertex subsets with many internal and few external edges." [Bader et al., 2013]

"A property that seems to be common to many networks is community structure, the division of network nodes into groups within which the network connections are dense, but between which they are sparser." [Newman and Girvan, 2004]

Unformal definitions – Screenshot from (Veldt2019)

"A community is often thought of as a set of nodes that has more connections between its members than to the remainder of the network." [Leskovec et al., 2008]

"Graph clustering is the task of grouping the vertices of the graph into clusters taking into consideration the edge structure of the graph in such a way that there should be many edges within each cluster and relatively few between the clusters." [Schaeffer, 2007]

"One mesoscopic structure, called a community, consists of a group of nodes that are relatively densely connected to each other but sparsely connected to other dense groups in the network." [Porter et al., 2009]

"Communities, or clusters, are usually groups of vertices having higher probability of being connected to each other than to members of other groups, though other patterns are possible." [Fortunato and Hric, 2016]

"The most basic task of community detection, or graph clustering, consists in partitioning the vertices of a graph into clusters that are more densely connected." [Abbe, 2018]

"Generally speaking, techniques for graph partitioning and graph clustering aim at the identification of vertex subsets with many internal and few external edges." [Bader et al., 2013]

"A property that seems to be common to many networks is community structure, the division of network nodes into groups within which the network connections are dense but between which they are sparser." [Newman and Girvan, 2004]

Unformal definitions – Screenshot from (Veldt2019)

"A community is often thought of as a set of nodes that has more connections between its members than to the remainder of the network." [Leskovec et al., 2008]

"Graph clustering is the task of grouping the vertices of the graph into clusters taking into consideration the edge structure of the graph in such a way that there should be many edges within each cluster and relatively few between the clusters." [Schaeffer, 2007]

"One mesoscopic structure, called a community, consists of a group of nodes that are relatively densely connected to each other but sparsely connected to other dense groups in the network." [Porter et al., 2009]

"Communities, or clusters, are usually groups of vertices having higher probability of being connected to each other than to members of other groups, though other patterns are possible." [Fortunato and Hric, 2016]

"The most basic task of community detection, or graph clustering, consists in partitioning the vertices of a graph into clusters that are more densely connected." [Abbe, 2018]

"Generally speaking, techniques for graph partitioning and graph clustering aim at the identification of vertex subsets with many internal and few external edges." [Bader et al., 2013]

"A property that seems to be common to many networks is community structure, the division of network nodes into groups within which the network connections are dense, but between which they are sparser." [Newman and Girvan, 2004]

How consistent is my community (structure)?

The Normalised Cuts (Shi&Malik2000)

Normalised Cuts: Denoting $vol(C) = \sum_{i \in C} d^{\omega}(i)$, the value of the normalised cuts is defined as:

$$\Phi(G, \mathcal{C}) = \sum_{C \in \mathcal{C}} \frac{Cut(C)}{vol(C)}.$$

How consistent is my community (structure)?
The Normalised Cuts (Shi&Malik2000)

Normalised Cuts: Denoting $vol(C) = \sum_{i \in C} d^{\omega}(i)$, the value of the normalised cuts is defined as:

$$\Phi(G, \mathcal{C}) = \sum_{C \in \mathcal{C}} \frac{Cut(C)}{vol(C)}.$$

Why not $\sum_{C \in \mathcal{C}} \frac{Cut(C)}{|C|}$?

How consistent is my community (structure)?

The Normalised Cuts (Shi&Malik2000)

Normalised Cuts: Denoting $vol(C) = \sum_{i \in C} d^{\omega}(i)$, the value of the normalised cuts is defined as:

$$\Phi(G, \mathcal{C}) = \sum_{C \in \mathcal{C}} \frac{Cut(C)}{vol(C)}.$$

Why not $\sum_{C \in \mathcal{C}} \frac{Cut(C)}{|C|}$? Because of **the real Cheeger's inequality**!

How consistent is my community (structure)?

The Normalised Cuts (Shi&Malik2000)

Normalised Cuts: Denoting $vol(C) = \sum_{i \in C} d^{\omega}(i)$, the value of the normalised cuts is defined as:

$$\Phi(G, \mathcal{C}) = \sum_{C \in \mathcal{C}} \frac{Cut(C)}{vol(C)}.$$

Why not $\sum_{C \in \mathcal{C}} \frac{Cut(C)}{|C|}$? Because of **the real Cheeger's inequality**!

- With $Mass(C) = \sum_{i \in S} Mass(i)$:

$$\lambda_2/2 \leq \frac{Cut(S)}{min(Mass(S), Mass(\overline{S}))} \leq \sqrt{2\lambda_2 \max_{i \in V} \frac{d^{\omega}(i)}{Mass(i)}}.$$

How consistent is my community (structure)?

The Normalised Cuts (Shi&Malik2000)

Normalised Cuts: Denoting $vol(C) = \sum_{i \in C} d^\omega(i)$, the value of the normalised cuts is defined as:

$$\Phi(G, \mathcal{C}) = \sum_{C \in \mathcal{C}} \frac{Cut(C)}{vol(C)}.$$

Why not $\sum_{C \in \mathcal{C}} \frac{Cut(C)}{|C|}$? Because of **the real Cheeger's inequality**!

- With $Mass(C) = \sum_{i \in S} Mass(i)$:

$$\lambda_2/2 \leq \frac{Cut(S)}{min(Mass(S), Mass(\bar{S}))} \leq \sqrt{2\lambda_2 \max_{i \in V} \frac{d^\omega(i)}{Mass(i)}}.$$

$\Longrightarrow$ $Mass(i) = d^\omega(i)$ makes the upper bound not degree-dependant.

How consistent is my community (structure)?

The Modularity (Newman&Girvan2003)

Idea: A **modular** network should have **more edges inside its communities than what is expected in a random graph with the same degree distribution**.

How consistent is my community (structure)?

The Modularity (Newman&Girvan2003)

Idea: A **modular** network should have **more edges inside its communities than what is expected in a random graph with the same degree distribution**.

Property Given $G_R = (V, E_R)$ a random undirected graph with prescribed node degrees $d(1), ..., d(n)$, then $Pr(i \sim j) = \frac{d(i)d(j)}{2m-1} \underset{m \gg 1}{\approx} \frac{d(i)d(j)}{2m}$.

Proof

How consistent is my community (structure)?

The Modularity (Newman&Girvan2003)

Idea: A **modular** network should have **more edges inside its communities than what is expected in a random graph with the same degree distribution**.

Property Given $G_R = (V, E_R)$ a random undirected graph with prescribed node degrees $d(1), ..., d(n)$, then $Pr(i \sim j) = \frac{d(i)d(j)}{2m-1} \underset{m \gg 1}{\approx} \frac{d(i)d(j)}{2m}$.

Proof

Modularity Given $m = |E|$ and assuming that $Pr(i \sim j) = \frac{d(i)d(j)}{2m}$ in G_R, then

$$\mathcal{Q}(G, \mathcal{C}) = \frac{1}{m} \sum_{C \in \mathcal{C}} |E \cap C \times C| - \mathbb{E}[|E_r \cap C \times C|] \quad (1)$$

$$= \frac{1}{m} \sum_{C \in \mathcal{C}} |E \cap C \times C| - \frac{vol(C)^2}{4m}. \quad (2)$$

Exercise Prove that (1) $\iff$ (2).

More on Modularity

Property Given, $\mathbf{A} \in \{0,1\}^{n \times n}$ the adjacency matrix of G, and stating $m = \sum_i d(i)/2$, one can write

$$\mathcal{Q}(G,\mathcal{C}) = \frac{1}{2m} \sum_{C \in \mathcal{C}} \sum_{i,j \in C} \left(a_{i,j} - \frac{d(i)d(j)}{2m} \right).$$

Proof Exercise

More on Modularity

Property Given, $\mathbf{A} \in \{0,1\}^{n\times n}$ the adjacency matrix of G, and stating $m = \sum_i d(i)/2$, one can write

$$\mathcal{Q}(G,\mathcal{C}) = \frac{1}{2m} \sum_{C\in\mathcal{C}} \sum_{i,j\in C} \left(a_{i,j} - \frac{d(i)d(j)}{2m} \right).$$

Proof Exercise

Extension If G is weighted, the Modularity is extended by stating $m = \sum_{i,j} a_{i,j}/2$, and

$$\mathcal{Q}(G,\mathcal{C}) = \frac{1}{2m} \sum_{C\in\mathcal{C}} \sum_{i,j\in C} \left(a_{i,j} - \frac{d^{\omega}(i)d^{\omega}(j)}{2m} \right).$$

More on Modularity

Property Given, $\mathbf{A} \in \{0,1\}^{n\times n}$ the adjacency matrix of G, and stating $m = \sum_i d(i)/2$, one can write

$$\mathcal{Q}(G,\mathcal{C}) = \frac{1}{2m} \sum_{C\in\mathcal{C}} \sum_{i,j\in C} \left(a_{i,j} - \frac{d(i)d(j)}{2m} \right).$$

Proof Exercise

Extension If G is weighted, the Modularity is extended by stating $m = \sum_{i,j} a_{i,j}/2$, and

$$\mathcal{Q}(G,\mathcal{C}) = \frac{1}{2m} \sum_{C\in\mathcal{C}} \sum_{i,j\in C} \left(a_{i,j} - \frac{d^{\omega}(i)d^{\omega}(j)}{2m} \right).$$

Property For a unweighted graph G,

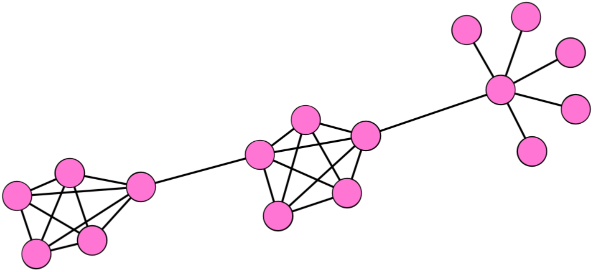
$$-1/2 \leq \mathcal{Q}(G,C) \leq 1.$$

Proof Exercise for $\mathcal{Q}(G,C) \leq 1$.

A Bunch of Algorithms

Edge Betweeness (Newman&Girvan2001)

Idea Edges that **bridge communities** are involved in many **shortest paths**.
Cutting those edges should reveal the modular structure of the network.



A Bunch of Algorithms

Edge Betweeness (Newman&Girvan2001)

Idea Edges that **bridge communities** are involved in many **shortest paths**. Cutting those edges should reveal the modular structure of the network.

Definition (G unweighted) $\forall u, v \in V, k_{min} = min\{k : u \in \mathcal{N}^k(v)\}$ and a k_{min}-path between u and v is called a **shortest path**. Given $e \in E$, the **betweeness** of e is

$$b(e) = \sum_{u \neq v \in V} \frac{\#\text{shortest paths between } u \text{ and } v \text{ that contain } e}{\#\text{shortest paths between } u \text{ and } v}.$$

Examples

A Bunch of Algorithms

Edge Betweeness (Newman&Girvan2001)

Idea Edges that **bridge communities** are involved in many **shortest paths**. Cutting those edges should reveal the modular structure of the network.

Definition (G unweighted) $\forall u, v \in V, k_{min} = min\{k : u \in \mathcal{N}^k(v)\}$ and a k_{min}-path between u and v is called a **shortest path**. Given $e \in E$, the **betweeness** of e is

$$b(e) = \sum_{u \neq v \in V} \frac{\#\text{shortest paths between } u \text{ and } v \text{ that contain } e}{\#\text{shortest paths between } u \text{ and } v}.$$

Examples

Algorithm 1) Compute the betweeness of each edge. 2) Remove the one with highest betweeness. 3) Update the betweeness of affected edges. 4) Go to Step 2.

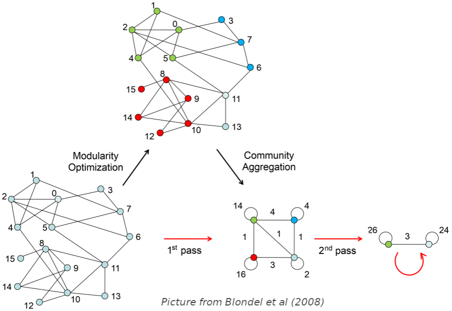
$\Longrightarrow$ A **divisive** algorithm that produces a **dendrogram**.

Example

A Bunch of Algorithms

Louvain (Blondel et al.2008)

Idea An efficient “heuristic” to maximise the Modularity $\mathcal{Q}(G,.)$[1].



Picture from Blondel et al (2008)

$\implies$ An **agglomerative** clustering.

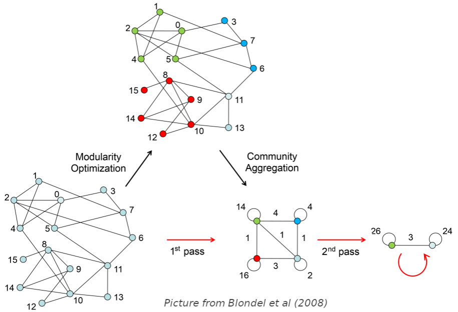
✓ Efficient, accurate, used to maximise other measures (not efficiently for all).

[1] whose actual maximisation is NP complete

A Bunch of Algorithms

Louvain (Blondel et al.2008)

Idea An efficient "heuristic" to maximise the Modularity $\mathcal{Q}(G, .)^1$.



Picture from Blondel et al (2008)

$\Longrightarrow$ An **agglomerative** clustering.

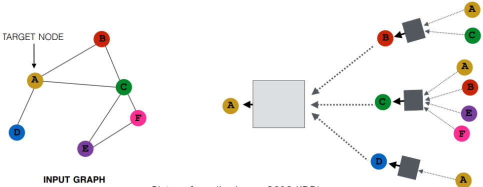
- ✓ Efficient, accurate, used to maximise other measures (not efficiently for all).
- ✗ A community returned by Louvain can be disconnected !
- Still one of the most used algorithms to date.

[1] whose actual maximisation is NP complete

A Bunch of Algorithms

Graph Convolutional Networks (Kipf&Welling2017)

Idea The state of a nodes depends on its neighbours: Convolutional neural networks on graphs.

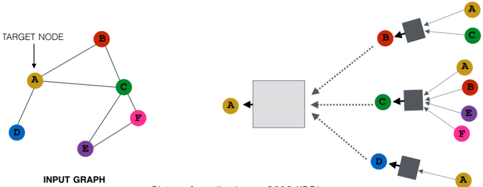


Picture from (Leskovec 2019,KDD)

A Bunch of Algorithms

Graph Convolutional Networks (Kipf&Welling2017)

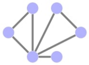
Idea The state of a nodes depends on its neighbours: Convolutional neural networks on graphs.



Picture from (Leskovec 2019,KDD)

Graph Convolutional Layer $\sigma(\mathbf{A}\mathbf{H}^{(t)}\mathbf{W}^{(t)})$, with σ nonlinear function and

- $\mathbf{A} \in \mathbb{R}^{n \times n}$ the adjacency matrix,
- $\mathbf{H}^{(t)} \in \mathbb{R}^{n \times d_t}$ the "features" of nodes at layer t,
- $\mathbf{W}^{(t)} \in \mathbb{R}^{d_t \times d_{t+1}}$ the weights to learn in the tth layer.



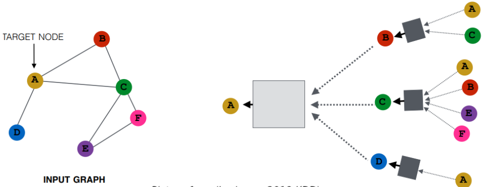
$\rightarrow$ $\mathbf{H}^{(1)} = \sigma(\mathbf{M}\mathbf{X}\mathbf{W}^{(0)})$ $\rightarrow$ $\mathbf{H}^{(2)} = \sigma(\mathbf{M}\mathbf{H}^{(1)}\mathbf{W}^{(1)})$ $\rightarrow \cdots \rightarrow$ $\mathbf{Z} = \sigma(\mathbf{M}\mathbf{H}^{(k)}\mathbf{W}^{(k)})$

$\mathbf{M} \in \mathbb{R}^{n \times n}$ graph. struct.
$\mathbf{X} \in \mathbb{R}^{n \times f}$ node features

A Bunch of Algorithms

Graph Convolutional Networks (Kipf&Welling2017)

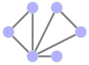
Idea The state of a nodes depends on its neighbours: Convolutional neural networks on graphs.



Picture from (Leskovec 2019,KDD)

Graph Convolutional Layer $\sigma(\mathbf{M}\mathbf{H}^{(t)}\mathbf{W}^{(t)})$, with σ nonlinear function and

- $\mathbf{M} = \widehat{\mathbf{D}}^{-1/2}\widehat{\mathbf{A}}\widehat{\mathbf{D}}^{-1/2}$, with $\widehat{\mathbf{A}} = \mathbf{I} + \mathbf{A}$ and $\widehat{\mathbf{D}} = diag(\widehat{\mathbf{A}}\mathbf{1})$
- $\mathbf{H}^{(t)} \in \mathbb{R}^{n \times d_t}$ the "features" of nodes at layer t,
- $\mathbf{W}^{(t)} \in \mathbb{R}^{d_t \times d_{t+1}}$ the weights to learn in the tth layer.

 → $\mathbf{H}^{(1)} = \sigma(\mathbf{M}\mathbf{X}\mathbf{W}^{(0)})$ → $\mathbf{H}^{(2)} = \sigma(\mathbf{M}\mathbf{H}^{(1)}\mathbf{W}^{(1)})$ → $\cdots$ → $\mathbf{Z} = \sigma(\mathbf{M}\mathbf{H}^{(k)}\mathbf{W}^{(k)})$

$\mathbf{M} \in \mathbb{R}^{n \times n}$ graph. struct.
$\mathbf{X} \in \mathbb{R}^{n \times f}$ node features

How close to the groundtruth is a community structure?

Problem Are two partitionings $\mathcal{C} = \{C_1, ..., C_p\}$ and $\mathcal{K} = \{K_1, ..., K_q\}$ close?

How close to the groundtruth is a community structure?

<u>Problem</u> Are two partitionings $\mathcal{C} = \{C_1, ..., C_p\}$ and $\mathcal{K} = \{K_1, ..., K_q\}$ close?

<u>Rand Index</u> Counting the ratio of agreements:

$$RI = \frac{n_{1,1} + n_{0,0}}{n_{1,1} + n_{0,1} + n_{1,0} + n_{0,0}}, \text{ with}$$

		According to $\mathcal{C}$	
	$\#\{u, v\}$	$=$ community	$\neq$ communities
According to $\mathcal{K}$	$=$ community	$n_{1,1}$	$n_{1,0}$
	$\neq$ communities	$n_{0,1}$	$n_{0,0}$

How close to the groundtruth is a community structure?

Problem Are two partitionings $\mathcal{C} = \{C_1, ..., C_p\}$ and $\mathcal{K} = \{K_1, ..., K_q\}$ close?

Rand Index Counting the ratio of agreements:

$$RI = \frac{n_{1,1} + n_{0,0}}{n_{1,1} + n_{0,1} + n_{1,0} + n_{0,0}}, \text{ with}$$

		According to $\mathcal{C}$	
	$\#\{u, v\}$	$=$ community	$\neq$ communities
According to $\mathcal{K}$	$=$ community	$n_{1,1}$	$n_{1,0}$
	$\neq$ communities	$n_{0,1}$	$n_{0,0}$

Adjusted against chance Assuming a random clustering with number of elts/cluster:

$$ARI = \frac{RI - \mathbb{E}[RI]}{max(RI) - \mathbb{E}[RI]} = \frac{2(n_{0,0}n_{1,1} - n_{0,1}n_{1,0})}{(n_{0,0} + n_{0,1})(n_{1,1} + n_{0,1}) + (n_{0,0} + n_{1,0})(n_{1,1} + n_{1,0})}$$

How close to the groundtruth is a community structure?

Mutual Information measures the mutual dependence between 2 random variables (rv). Based on the **entropy**.

Definition: Given a rv X taking values $x_1, ..., x_k$, the **entropy** of X is

$$H(X) = -\sum_{i=1}^{k} Pr(X = x_i) \times log_2(Pr(X = x_i))$$

Remark Entropy is said to be the **average level of surprise** of X. Why?

How close to the groundtruth is a community structure?

Mutual Information measures the mutual dependence between 2 random variables (rv). Based on the **entropy**.

Definition: Given a rv X taking values $x_1, ..., x_k$, the **entropy** of X is

$$H(X) = -\sum_{i=1}^{k} Pr(X = x_i) \times log_2(Pr(X = x_i))$$

Remark Entropy is said to be the **average level of surprise** of X. Why?

Definition Given 2 rv X, Y with same values, their **mutual information** is

$$MI = H(X) - H(X|Y)$$

How close to the groundtruth is a community structure?

Mutual Information measures the mutual dependence between 2 random variables (rv). Based on the **entropy**.

Definition: Given a rv X taking values $x_1, ..., x_k$, the **entropy** of X is

$$H(X) = -\sum_{i=1}^{k} Pr(X = x_i) \times log_2(Pr(X = x_i))$$

Remark Entropy is said to be the **average level of surprise** of X. Why?

Definition Given 2 rv X, Y with same values, their **mutual information** is

$$MI = H(X) - H(X|Y)$$

Seeing clusterings as rv C, K, with $Pr(C = C_i) = |C_i|/n$, thus

$$MI = \sum_{i=1}^{p} \sum_{j=1}^{q} |C_i \cap K_j|/N \times log_2(n \times |C_i \cap K_j|/(|C_i||K_J|))$$

How close to the groundtruth is a community structure?

Mutual Information measures the mutual dependence between 2 random variables (rv). Based on the **entropy**.

Definition: Given a rv X taking values $x_1, ..., x_k$, the **entropy** of X is

$$H(X) = -\sum_{i=1}^{k} Pr(X = x_i) \times log_2(Pr(X = x_i))$$

Remark Entropy is said to be the **average level of surprise** of X. Why?

Definition Given 2 rv X, Y with same values, their **mutual information** is

$$MI = H(X) - H(X|Y)$$

Seeing clusterings as rv C, K, with $Pr(C = C_i) = |C_i|/n$, thus

$$MI = \sum_{i=1}^{p} \sum_{j=1}^{q} |C_i \cap K_j|/N \times log_2(n \times |C_i \cap K_j|/(|C_i||K_J|))$$

Remark The **normalized MI** is most often used. An **adjusted MI** exists but less used because of its complexity.

Example

Conclusion

- **Community detection** means **finding consistent groups of nodes** within a network.
- What a good community should look like is highly **application dependant** (groups of densely connected nodes, but how?).
- This field has been **built on the fly** to answer real world problem of practitioners.
- **Modularity, Louvain algorithm, GCNs**, etc. are used because **they work globally well**, even if **they have flaws**.
- **No consensus on what does it means to be close for clusterings**.

? And for more complex networks ?