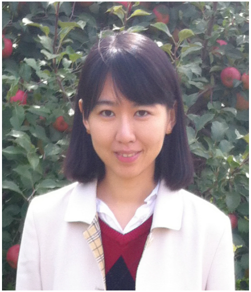
Community Trees in Networks

Yen-Chi Chen

Department of Statistics
University of Washington

Collaborators

Ruqian Chen
UW Math



Wei Guo
UW ISE

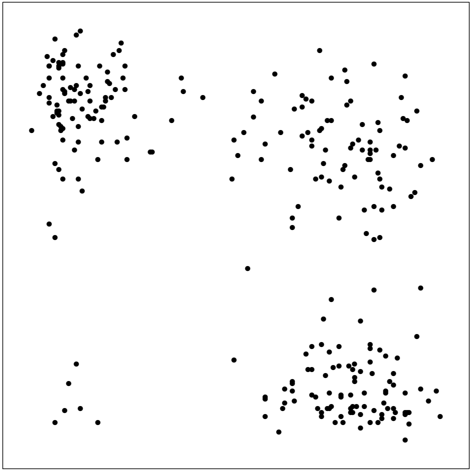
Ashis Banerjee
UW ISE&ME

Outline

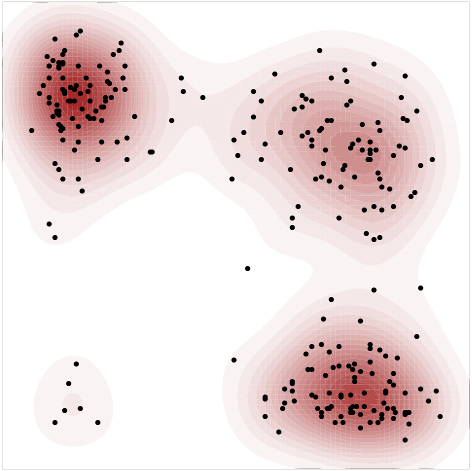
- Review: Density Tree
- Community Tree in Networks
- Future Work

Density Tree

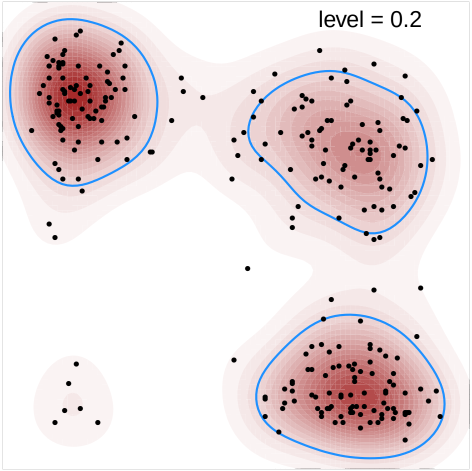
Clusters and Density Function: an Illustration



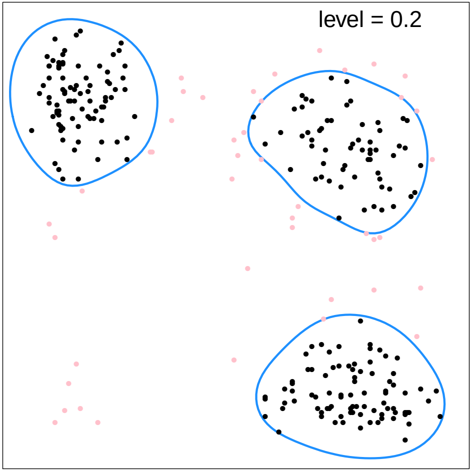
Clusters and Density Function: an Illustration



Clusters and Density Function: an Illustration



Clusters and Density Function: an Illustration



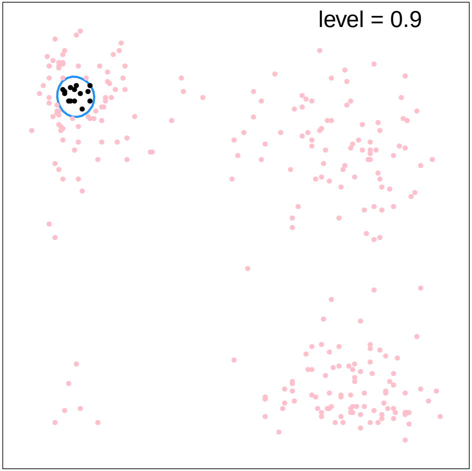
Clusters and Density Function - 1

- The idea of using a density level (threshold) λ leads to clusters representing high density regions.
- Thus, the level λ has an effect on the clustering result.

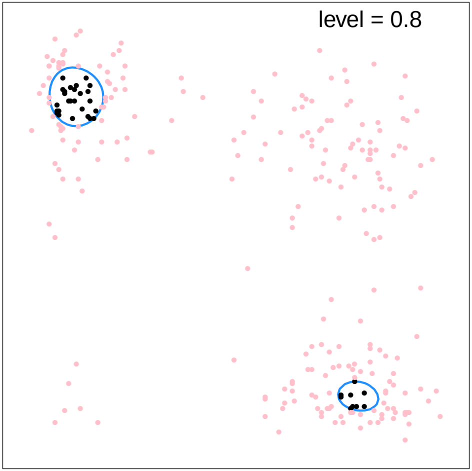
Clusters and Density Function - 1

- The idea of using a density level (threshold) λ leads to clusters representing high density regions.
- Thus, the level λ has an effect on the clustering result.
- Varying the level λ may lead to a creation of a new cluster or a merging of existing clusters.

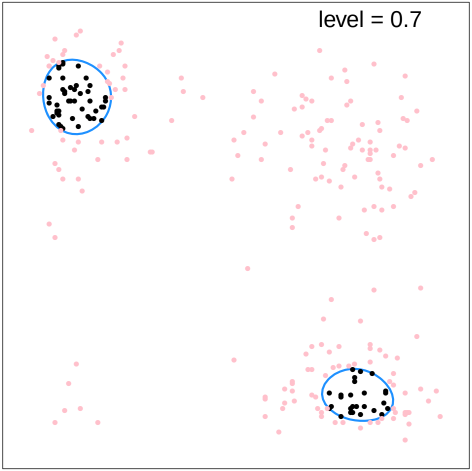
Clusters and Density Function: Different Levels



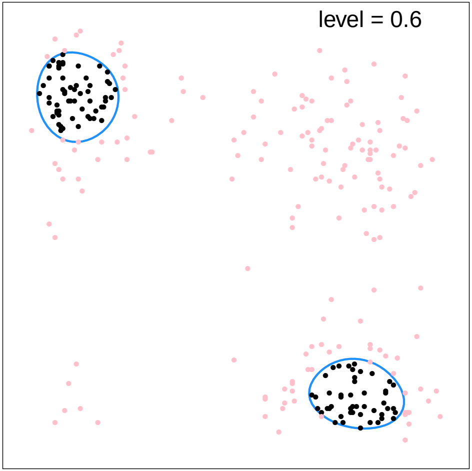
Clusters and Density Function: Different Levels



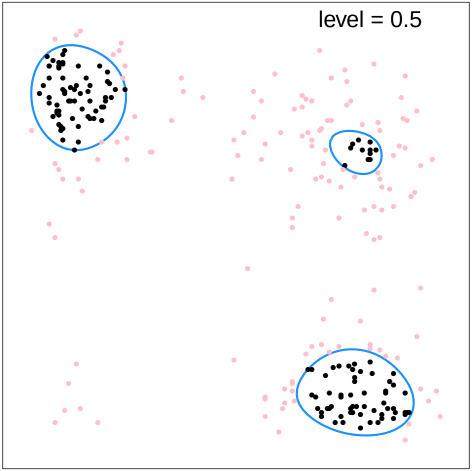
Clusters and Density Function: Different Levels



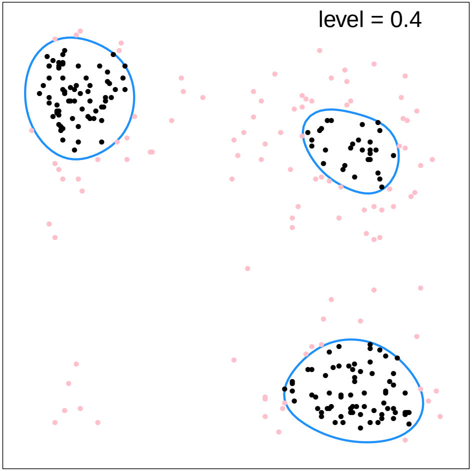
Clusters and Density Function: Different Levels



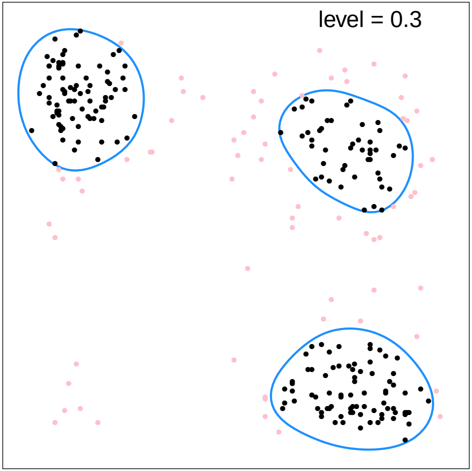
Clusters and Density Function: Different Levels



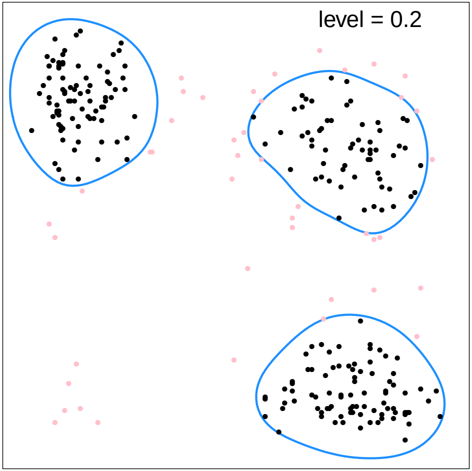
Clusters and Density Function: Different Levels



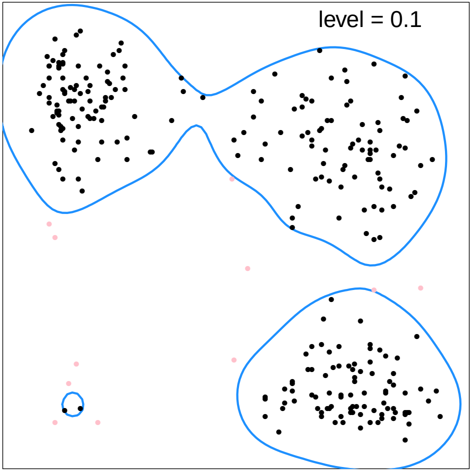
Clusters and Density Function: Different Levels



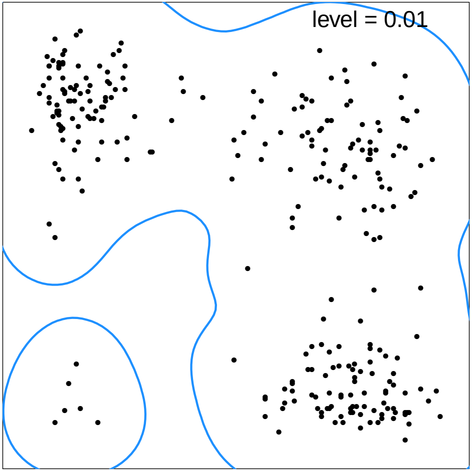
Clusters and Density Function: Different Levels



Clusters and Density Function: Different Levels



Clusters and Density Function: Different Levels



Clusters and Density Function - 2

- When the level changes, we see the *evolution* of clusters.

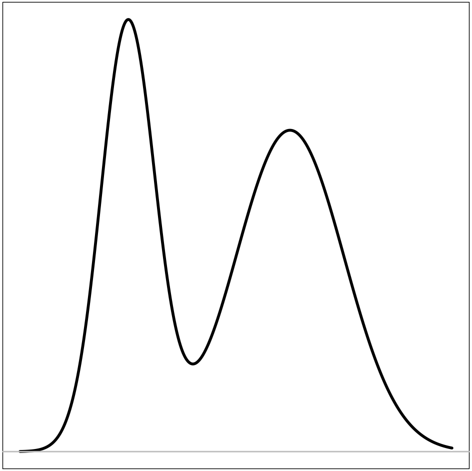
Clusters and Density Function - 2

- When the level changes, we see the *evolution* of clusters.
- Cluster tree (Stuetzle 2003) is to summarize such an evolution process by a tree.

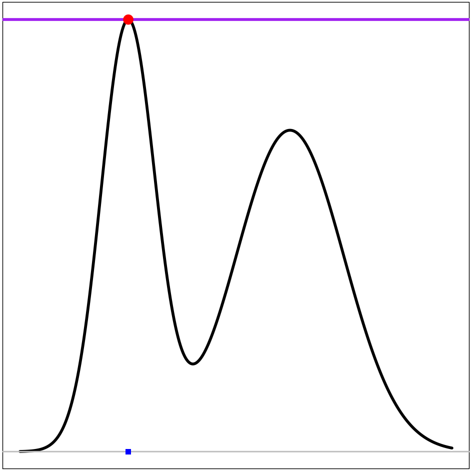
Clusters and Density Function - 2

- When the level changes, we see the *evolution* of clusters.
- Cluster tree (Stuetzle 2003) is to summarize such an evolution process by a tree.
- When applied to a density function, a cluster tree is also called a density tree (Klemelä 2004).

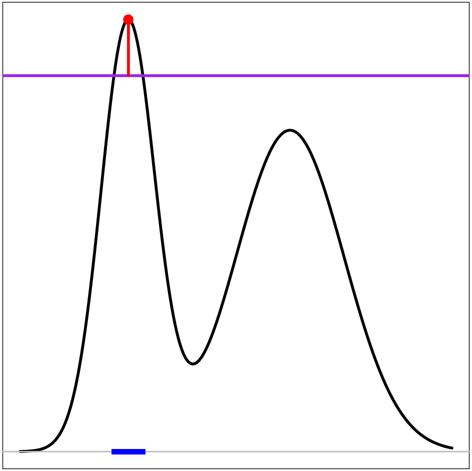
Density Tree: an Illustration



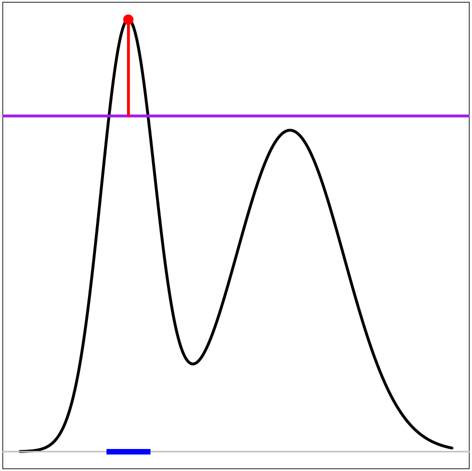
Density Tree: an Illustration



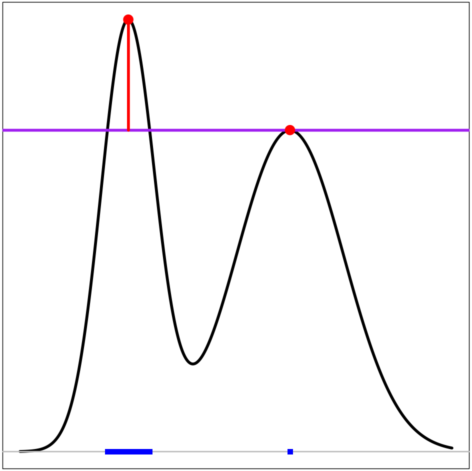
Density Tree: an Illustration



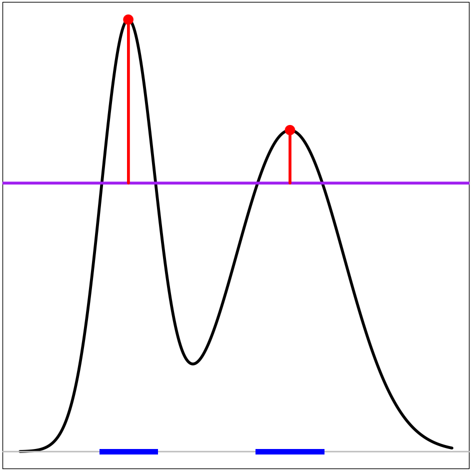
Density Tree: an Illustration



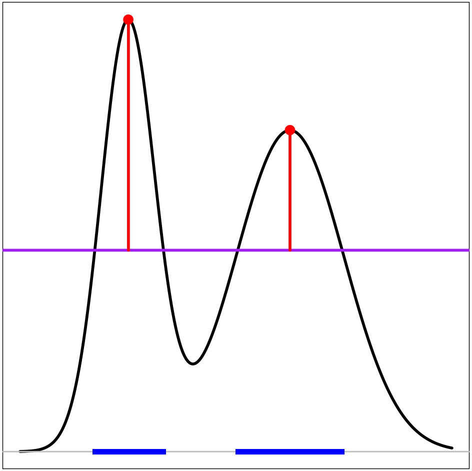
Density Tree: an Illustration



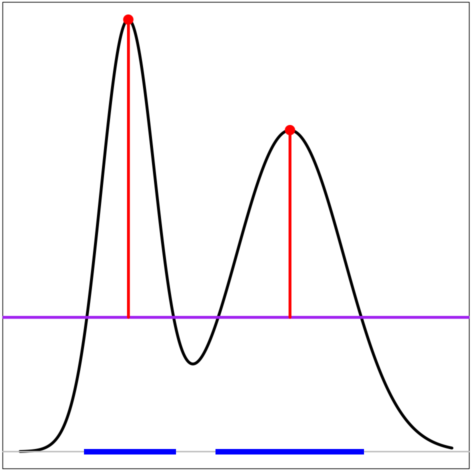
Density Tree: an Illustration



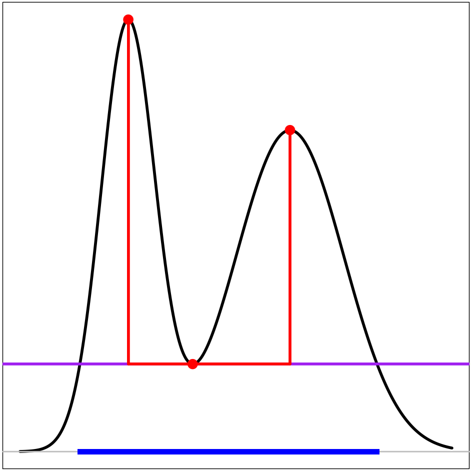
Density Tree: an Illustration



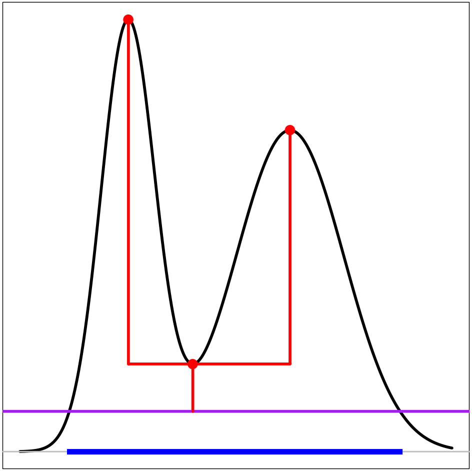
Density Tree: an Illustration



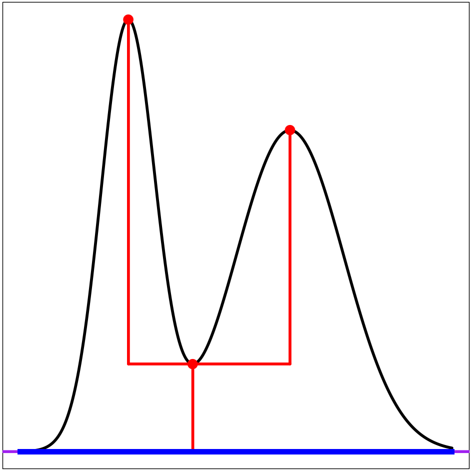
Density Tree: an Illustration



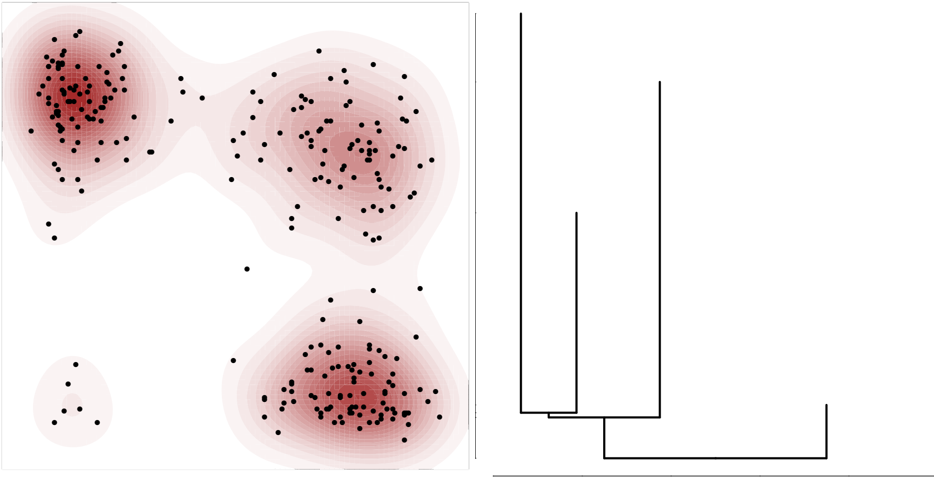
Density Tree: an Illustration



Density Tree: an Illustration



An Example of 2D Density Tree



Features of Density Trees

- Density trees provide topological information about the density function and they can be transformed into the persistent diagrams easily.

Features of Density Trees

- Density trees provide topological information about the density function and they can be transformed into the persistent diagrams easily.
- When using density level sets to define clusters, the density tree contains the information about the evolution and stability of clusters.

Features of Density Trees

- Density trees provide topological information about the density function and they can be transformed into the persistent diagrams easily.
- When using density level sets to define clusters, the density tree contains the information about the evolution and stability of clusters.
- Moreover, density trees can always be displayed in 2D plane. So they are good tools for visualizing multivariate functions.

Community Trees

Community Detection in Networks

- We consider simple networks – undirected, unweighted graphs.
- While there are many methods for analyzing a network, we focus on one particular method – the clique percolation method (CPM; Palla et al. 2005, 2007).

Community Detection in Networks

- We consider simple networks – undirected, unweighted graphs.
- While there are many methods for analyzing a network, we focus on one particular method – the clique percolation method (CPM; Palla et al. 2005, 2007).
- CPM is a popular and powerful method in community detection.

- We consider simple networks – undirected, unweighted graphs.
- While there are many methods for analyzing a network, we focus on one particular method – the clique percolation method (CPM; Palla et al. 2005, 2007).
- CPM is a popular and powerful method in community detection.
- CPM uses cliques and their overlapping to define communities.

Community Detection in Networks

- We consider simple networks – undirected, unweighted graphs.
- While there are many methods for analyzing a network, we focus on one particular method – the clique percolation method (CPM; Palla et al. 2005, 2007).
- CPM is a popular and powerful method in community detection.
- CPM uses cliques and their overlapping to define communities.
- Communities from CPM can be overlapping – this allows a broader way to interpret communities.

A Key Insight

$$
\begin{aligned}
\text{Finding level sets} &\leftrightarrow \text{CPM} \\
\text{Density level } \lambda &\leftrightarrow \text{Clique order } k \\
\text{Clusters} &\leftrightarrow \text{Communities} \\
\text{Density/cluster trees} &\leftrightarrow \text{Community trees}
\end{aligned}
$$

Cliques and Communities

- Cliques: a subgraph such that all vertices are connected.
- k-clique: a clique with k vertices.

Cliques and Communities

- Cliques: a subgraph such that all vertices are connected.
- k-clique: a clique with k vertices.
- Given a level k, the CPM finds all k-cliques in the network and then forms an adjacency matrix A for these cliques.
- If the i-th and j-th k-cliques share the same $(k-1)$ vertices, then $A_{ij} = 1$ and 0 otherwise.

Cliques and Communities

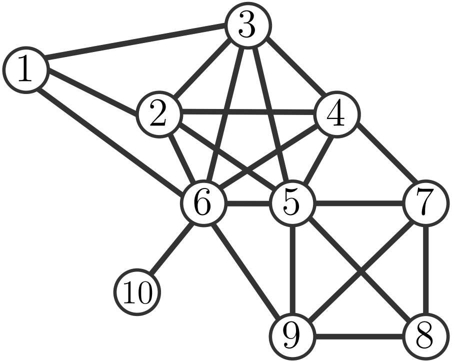
- Cliques: a subgraph such that all vertices are connected.
- k-clique: a clique with k vertices.
- Given a level k, the CPM finds all k-cliques in the network and then forms an adjacency matrix A for these cliques.
- If the i-th and j-th k-cliques share the same $(k-1)$ vertices, then $A_{ij} = 1$ and 0 otherwise.
- A k-clique community (or k-community for short) is a subgraph generated by the union of k-cliques in the same connected component of A.

Cliques and Communities

- Cliques: a subgraph such that all vertices are connected.
- k-clique: a clique with k vertices.
- Given a level k, the CPM finds all k-cliques in the network and then forms an adjacency matrix A for these cliques.
- If the i-th and j-th k-cliques share the same $(k-1)$ vertices, then $A_{ij} = 1$ and 0 otherwise.
- A k-clique community (or k-community for short) is a subgraph generated by the union of k-cliques in the same connected component of A.
- The number k of a clique community is called the order.

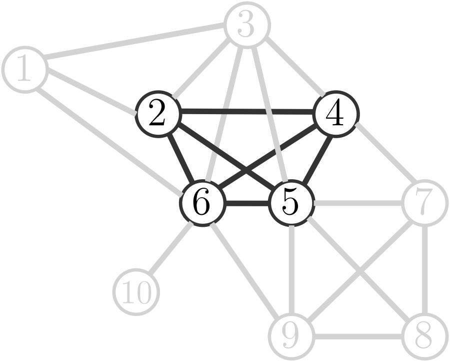
An Example: 4-communities

What are the 4-communities in the following network?



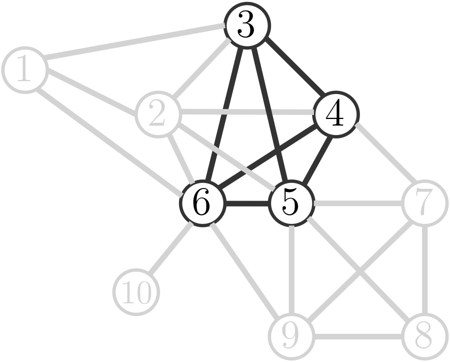
An Example: 4-communities

What are the 4-communities in the following network?



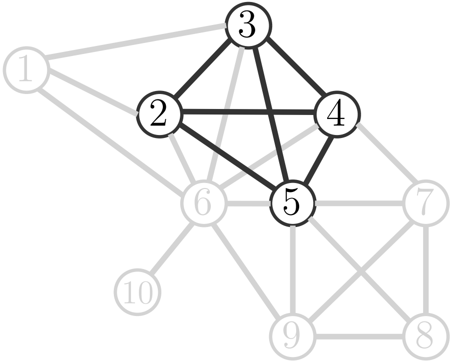
An Example: 4-communities

What are the 4-communities in the following network?



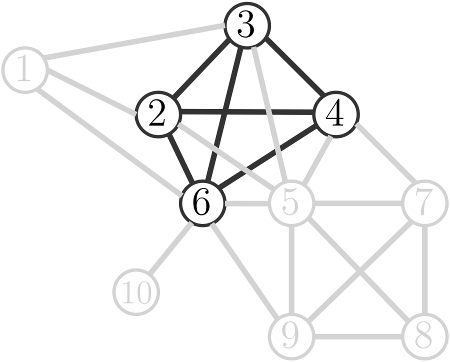
An Example: 4-communities

What are the 4-communities in the following network?



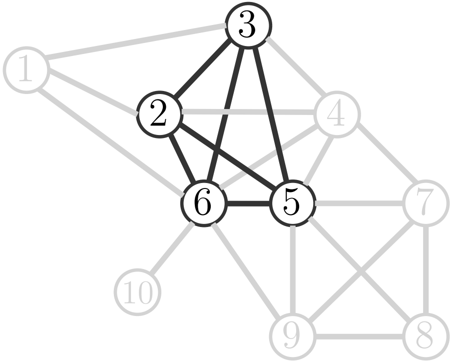
An Example: 4-communities

What are the 4-communities in the following network?



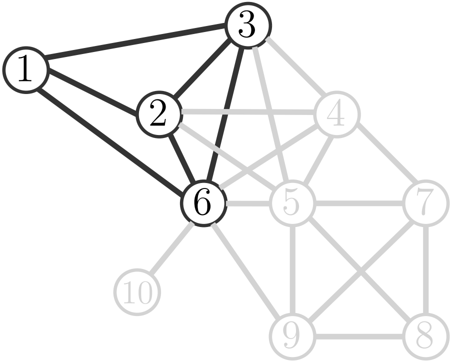
An Example: 4-communities

What are the 4-communities in the following network?



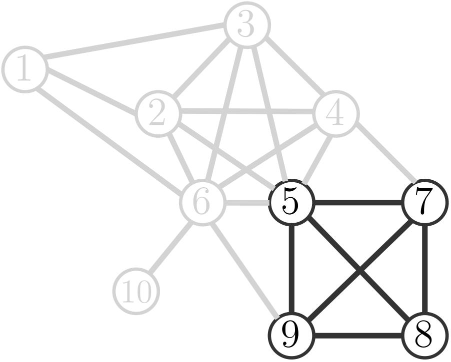
An Example: 4-communities

What are the 4-communities in the following network?



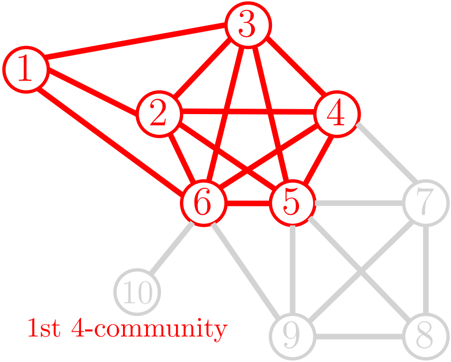
An Example: 4-communities

What are the 4-communities in the following network?



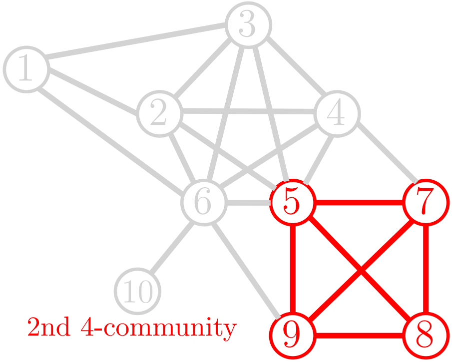
An Example: 4-communities

What are the 4-communities in the following network?

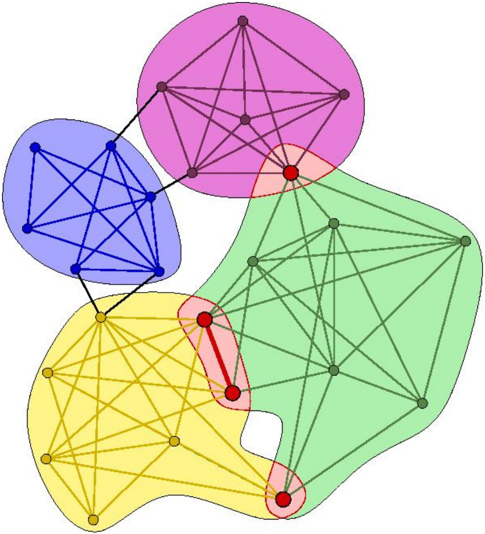


An Example: 4-communities

What are the 4-communities in the following network?



A more Complex Network



Another example of 4-communities (source: wikipedia).

Communities of Different Orders

- What happens if we vary the number k?

Communities of Different Orders

- What happens if we vary the number k?
- A k-cliques can be written as the union of k distinct $(k-1)$-cliques.
- Thus, a k-community will be a subgraph of a $(k-1)$-community.

Communities of Different Orders

- What happens if we vary the number k?
- A k-cliques can be written as the union of k distinct $(k-1)$-cliques.
- Thus, a k-community will be a subgraph of a $(k-1)$-community.
- Then for a k-community $\mathscr{C}_k$, there exists a sequence of subgraphs

$$\mathscr{C}_k \subset \mathscr{C}_{k-1} \subset \cdots \subset \mathscr{C}_1,$$

where $\mathscr{C}_\omega$ is an ω-community.

Communities of Different Orders

- What happens if we vary the number k?
- A k-cliques can be written as the union of k distinct $(k-1)$-cliques.
- Thus, a k-community will be a subgraph of a $(k-1)$-community.
- Then for a k-community $\mathscr{C}_k$, there exists a sequence of subgraphs

$$\mathscr{C}_k \subset \mathscr{C}_{k-1} \subset \cdots \subset \mathscr{C}_1,$$

  where $\mathscr{C}_\omega$ is an ω-community.
- This property, which we refer to as the nested property, defines a tree structure of all (clique) communities within a graph.

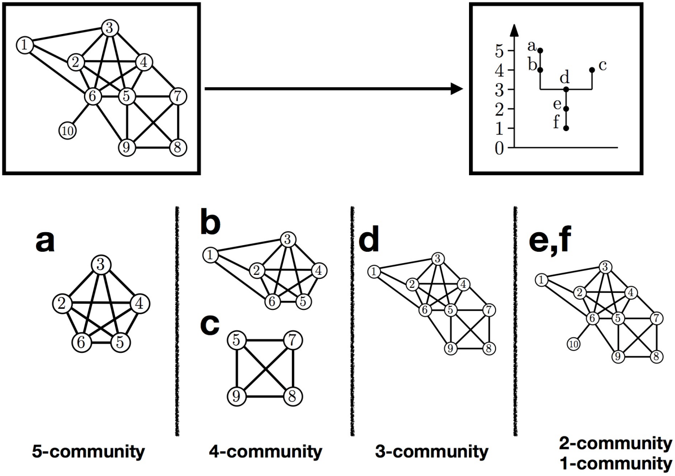
Communities of Different Orders

- What happens if we vary the number k?
- A k-cliques can be written as the union of k distinct $(k-1)$-cliques.
- Thus, a k-community will be a subgraph of a $(k-1)$-community.
- Then for a k-community $\mathscr{C}_k$, there exists a sequence of subgraphs

  $$\mathscr{C}_k \subset \mathscr{C}_{k-1} \subset \cdots \subset \mathscr{C}_1,$$

  where $\mathscr{C}_\omega$ is an ω-community.
- This property, which we refer to as the nested property, defines a tree structure of all (clique) communities within a graph.
- The resulting tree is called the **community tree**.

Community Tree: an Example



A Key Insight (revisited)

Finding level sets $\leftrightarrow$ CPM

Density level $\lambda \leftrightarrow$ Clique order k

Clusters $\leftrightarrow$ Communities

Density/cluster trees $\leftrightarrow$ Community trees

Community Tree

- Each node of the tree represents a clique community.
- This tree shows how each community evolves when we vary the order k.

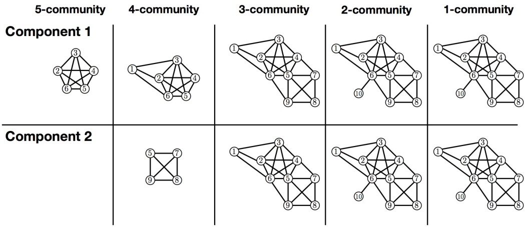
Community Tree

- Each node of the tree represents a clique community.
- This tree shows how each community evolves when we vary the order k.
- The merging node represents the order where multiple communities at a higher order are merged in to the same community.
- In a sense, the community tree can be viewed as a generalization of the cluster tree to networks.

Community Tree

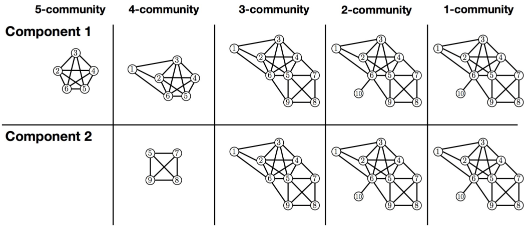
- Each node of the tree represents a clique community.
- This tree shows how each community evolves when we vary the order k.
- The merging node represents the order where multiple communities at a higher order are merged in to the same community.
- In a sense, the community tree can be viewed as a generalization of the cluster tree to networks.
- Moreover, the community tree leads to a persistent diagram.

Components in a Community Tree



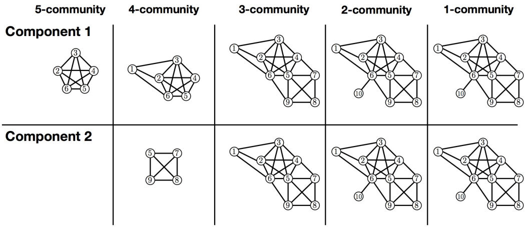
- A component in a community tree is a series of nodes starting from a leave that moves all the way down.

Components in a Community Tree



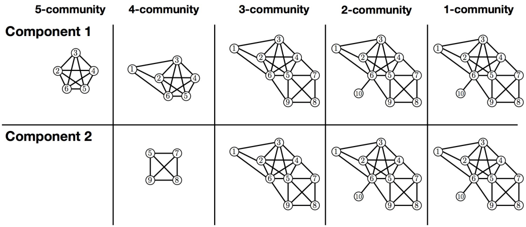
- A component in a community tree is a series of nodes starting from a leave that moves all the way down.
- The birth time of a component is the highest order of its nodes.

Components in a Community Tree



- A component in a community tree is a series of nodes starting from a leave that moves all the way down.
- The birth time of a component is the highest order of its nodes.
- Two components merge at an order if they share the same node. When two components merge, the one that has a lower birth time merged into the other component.

Components in a Community Tree

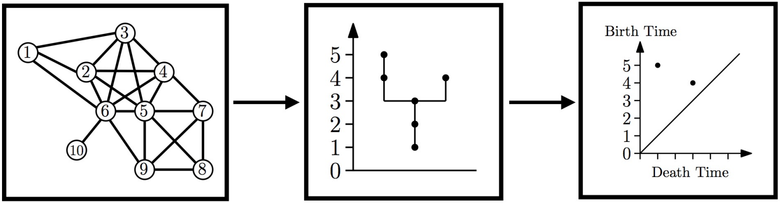


- A component in a community tree is a series of nodes starting from a leave that moves all the way down.
- The birth time of a component is the highest order of its nodes.
- Two components merge at an order if they share the same node. When two components merge, the one that has a lower birth time merged into the other component.
- The death time of a component is the highest order that it merge into another component.

Persistent Diagram of a Community Tree

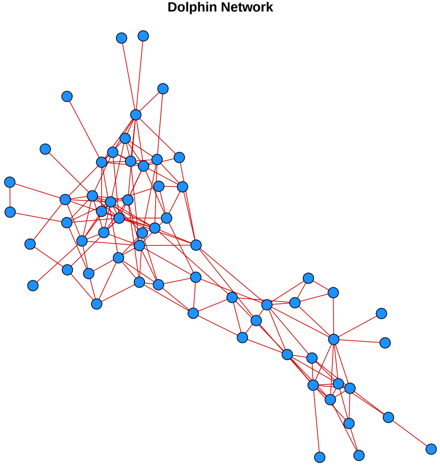
- Using the birth and death time of components, we obtain the persistent diagram of a community tree.
- Let $(b_1, d_1), \cdots, (b_K, d_K)$ be the birth time and death time of components of a community tree. The persistent diagram is

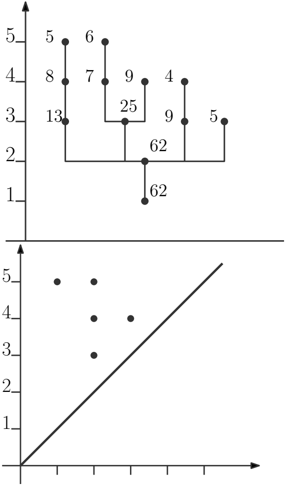
$$\mathsf{PD} = \{(d_i, b_i) : i = 1, \cdots, K\} \cup \{(d, b) : d = b\}.$$



Example: Dophin Network

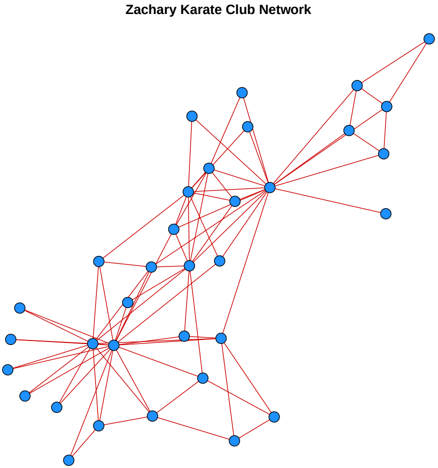
62 Dophins' social network data with 159 edges.

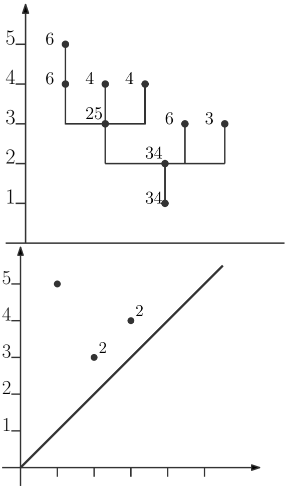




Example: Zachary Karate Club Network

A social network data about Zachary karate club; 34 vertices and 78 edges.





Stability of a Community Tree - 1

- For a network G_0, if we only perturb it a little bit, how will its community tree change?
- Namely, we want to understand the stability of a community tree.

Stability of a Community Tree - 1

- For a network G_0, if we only perturb it a little bit, how will its community tree change?
- Namely, we want to understand the stability of a community tree.
- However, quantifying the tree difference is not easy.

Stability of a Community Tree - 1

- For a network G_0, if we only perturb it a little bit, how will its community tree change?
- Namely, we want to understand the stability of a community tree.
- However, quantifying the tree difference is not easy.
- Here we measure their difference using the bottleneck distance between the corresponding persistent diagrams.

Stability of a Community Tree - 2

- Given two persistent diagrams $\mathsf{PD}_1, \mathsf{PD}_2$, their bottleneck distance is

$$d_\infty(\mathsf{PD}_1, \mathsf{PD}_2) = \inf_{\gamma} \sup_{A \in \mathsf{PD}_1} \|A - \gamma(A)\|_\infty,$$

where the infimum is taking over all bijective mappings between PD_1 and PD_2.

Stability of a Community Tree - 2

- Given two persistent diagrams $\mathsf{PD}_1, \mathsf{PD}_2$, their bottleneck distance is

$$d_\infty(\mathsf{PD}_1, \mathsf{PD}_2) = \inf_{\gamma} \sup_{A \in \mathsf{PD}_1} \|A - \gamma(A)\|_\infty,$$

  where the infimum is taking over all bijective mappings between PD_1 and PD_2.

- Let $\mathsf{PB}(T)$ be the persistent diagram of a community tree T. Then we define a distance d_B for community trees T_1 and T_2 as

$$d_B(T_1, T_2) = d_\infty(\mathsf{PB}(T_1), \mathsf{PB}(T_2)).$$

Stability of a Community Tree - 3

- Given two networks G_1 and G_2, let $T(G_1)$ and $T(G_2)$ be their corresponding community trees.
- It turns out that the difference between their community trees are bounded by a quantity called the total star number $\mathsf{TSN}(G_1, G_2)$:

Theorem (Chen et al. 2017)

Let G_1 and G_2 be two networks. Then

$$d_B(T(G_1), T(G_2)) \leq \mathsf{TSN}(G_1, G_2).$$

Total Star Number

- The total star number

$$\mathsf{TSN}(G_1, G_2) = \mathsf{RSN}(G_1, G_2) + \mathsf{RSN}(G_2, G_1).$$

- $\mathsf{RSN}(G_1, G_2)$ is the removal star number which is defined as

$$\mathsf{RSN}(G_1, G_2) = \min\{|V_0| : v(e) \cap V_0 \neq \emptyset \;\; \forall e \in E(G_1) \backslash E(G_2)\},$$

  where V_0 is a collection of vertices and $|V_0|$ is the number of elements in the set V_0 and $E(G)$ is the edge of a network G and $v(e)$ is the vertices attached to the edge e.

Total Star Number

- The total star number

$$\mathsf{TSN}(G_1, G_2) = \mathsf{RSN}(G_1, G_2) + \mathsf{RSN}(G_2, G_1).$$

- $\mathsf{RSN}(G_1, G_2)$ is the removal star number which is defined as

$$\mathsf{RSN}(G_1, G_2) = \min\{|V_0| : v(e) \cap V_0 \neq \emptyset \ \ \forall e \in E(G_1) \backslash E(G_2)\},$$

  where V_0 is a collection of vertices and $|V_0|$ is the number of elements in the set V_0 and $E(G)$ is the edge of a network G and $v(e)$ is the vertices attached to the edge e.
- $\mathsf{RSN}(G_1, G_2)$ can be interpreted as the minimal number of vertices we need to remove so that G_1 is a subgraph of G_2.

Total Star Number

- The total star number

$$\mathsf{TSN}(G_1, G_2) = \mathsf{RSN}(G_1, G_2) + \mathsf{RSN}(G_2, G_1).$$

- $\mathsf{RSN}(G_1, G_2)$ is the removal star number which is defined as

$$\mathsf{RSN}(G_1, G_2) = \min\{|V_0| : v(e) \cap V_0 \neq \emptyset \;\; \forall e \in E(G_1) \backslash E(G_2)\},$$

  where V_0 is a collection of vertices and $|V_0|$ is the number of elements in the set V_0 and $E(G)$ is the edge of a network G and $v(e)$ is the vertices attached to the edge e.
- $\mathsf{RSN}(G_1, G_2)$ can be interpreted as the minimal number of vertices we need to remove so that G_1 is a subgraph of G_2.
- Informally, the total star number can be interpreted as the minimal number of vertices that the network difference can be attributed to.

Stability of a Community Tree - 4

Theorem (Chen et al. 2017)

Let G_1 and G_2 be two networks. Then

$$d_B(T(G_1), T(G_2)) \leq \mathsf{TSN}(G_1, G_2).$$

- The TSN can be small while many edges are removed.

Stability of a Community Tree - 4

Theorem (Chen et al. 2017)

Let G_1 and G_2 be two networks. Then

$$d_B(T(G_1), T(G_2)) \leq \mathsf{TSN}(G_1, G_2).$$

- The TSN can be small while many edges are removed.
- For instance, if G_1 is the same as G_2 except removing all edges connecting to a particular vertex of G_1, then $\mathsf{TSN}(G_1, G_2) = 1$.

Stability of a Community Tree - 4

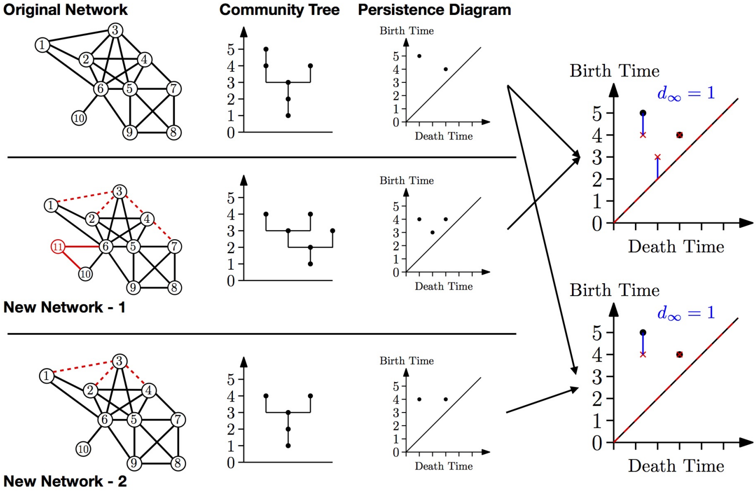
Theorem (Chen et al. 2017)

Let G_1 and G_2 be two networks. Then

$$d_B(T(G_1), T(G_2)) \leq \mathsf{TSN}(G_1, G_2).$$

- The TSN can be small while many edges are removed.
- For instance, if G_1 is the same as G_2 except removing all edges connecting to a particular vertex of G_1, then $\mathsf{TSN}(G_1, G_2) = 1$.
- Computing the total star number *does not* require building a community tree.

Community Tree: an Example



Computing the Total Star Number

- Although the total star number provides a useful bound for community trees, it cannot be computed easily.

Theorem (Chen et al. 2017)

Computing the total star number is an NP-complete problem.

- Note that the proof relies only on one simple observation: computing the total star number is the same as finding the minimum vertex cover.

Future Work

Future Directions

- Practical algorithm for bounding the total star number.
- Visualization tool using community trees.
- Effects from stochastic updates on community trees.
- Connections to overlapping communities.

Thank You!

More details can be found in Chen et al. (2017):
"A Note on Community Trees in Networks"
(https://arxiv.org/abs/1710.03924)

References

1. Chen, Ruqian, et al. "A Note on Community Trees in Networks." arXiv preprint arXiv:1710.03924 (2017).
2. Stuetzle, Werner. "Estimating the cluster tree of a density by analyzing the minimal spanning tree of a sample." Journal of classification 20, no. 1 (2003): 025-047.
3. Klemelä, Jussi. "Visualization of multivariate density estimates with level set trees." Journal of Computational and Graphical Statistics 13, no. 3 (2004): 599-620.
4. Palla, Gergely, Albert-László Barabási, and Tamás Vicsek. "Quantifying social group evolution." Nature 446.7136 (2007): 664-667.
5. Palla, Gergely, et al. "Uncovering the overlapping community structure of complex networks in nature and society." Nature 435.7043 (2005): 814-818.