Econ 7217 Economic Analysis of Social Networks
Community Detection

Chih-Sheng Hsieh

Department of Economics
National Taiwan University

May 20, 2024

Background

- Communities, also called clusters or modules, are groups of vertices which probably share common properties and/or play similar roles within the network (graph).
- The aim of community detection in graphs is to identify the modules and, possibly, their hierarchical organization, by only using the information encoded in the graph topology.
- This lecture slide is written based on Fortunato (2010), which is a comprehensive survey article on network community detection, and the lecture slides of Leonid E Zhukov.
- Orman et al. (2012) and Harenberg et al. (2014) also provide useful references which compare community detection methods.

Background

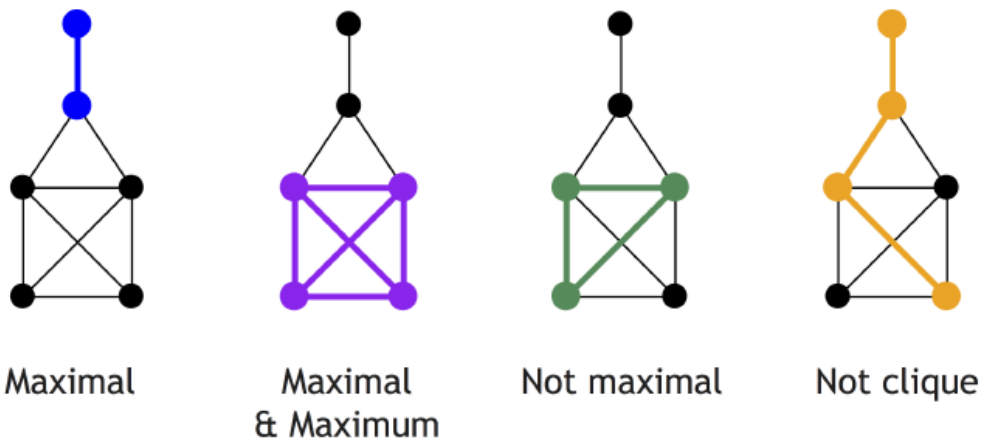
- Community detection have concrete applications:
  1. improve the service performance on the World Wide Web (WWW), in that each cluster of clients could be served by a dedicated mirror server.
  2. identify clusters of customers with similar interests in the network of purchase relationships between customers and products of online retailers, which enables one to set up efficient recommendation systems.
  3. clusters of large graphs can be used to create data structures in order to efficiently store the graph data and to handle navigational queries, like path searches.
  4. identify the hierarchical organization displayed by most networked systems in the real world. For example, Bai et al. (2019) study conglomerate formation in China.

Communities v.s Cliques

- We first distinguish **communities** and **cliques** in graphs.
- A network community is a set of nodes which have dense connections within the group, and sparse connections outside the group.
- The community-finding algorithms generally optimize some parameter (usually related to the number of within-group and between-group edges). They are also generally stochastic, meaning that you may get slightly different answers on different runs.
- A clique is in some sense a stronger version of a community. A set of nodes forms a clique (equivalently, a complete subgraph) where each pair of vertices is connected. For example, a two-node clique is simply two connected nodes. A three node clique is also known as a triangle.

Maximal and Maximum Cliques

- A **maximal** clique is a clique that cannot be extended by including one or more adjacent vertex.
- A **maximum** clique is a clique of the largest possible size in a given graph



- See the demonstration code to find the maximum clique using R.

Relaxation of a Clique: k-plex and k-core

- k-plex of size n is a maximal subset of n vertices such that each vertex is connected to at least $n - k$ others in the subset, i.e., any vertex can be lacking ties of no more than k members.
- k-core is a maximal subset of vertices such that each is connected to at least k others in the subset (degree of every vertex in k-core is $\geq k$). $(k+1)$-core is always a subgraph of k-core.

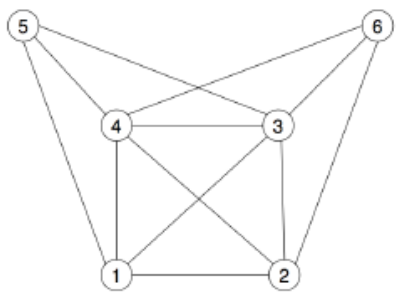


Figure: Example of 3-plex

Relaxation of a Clique: k-plex and k-core

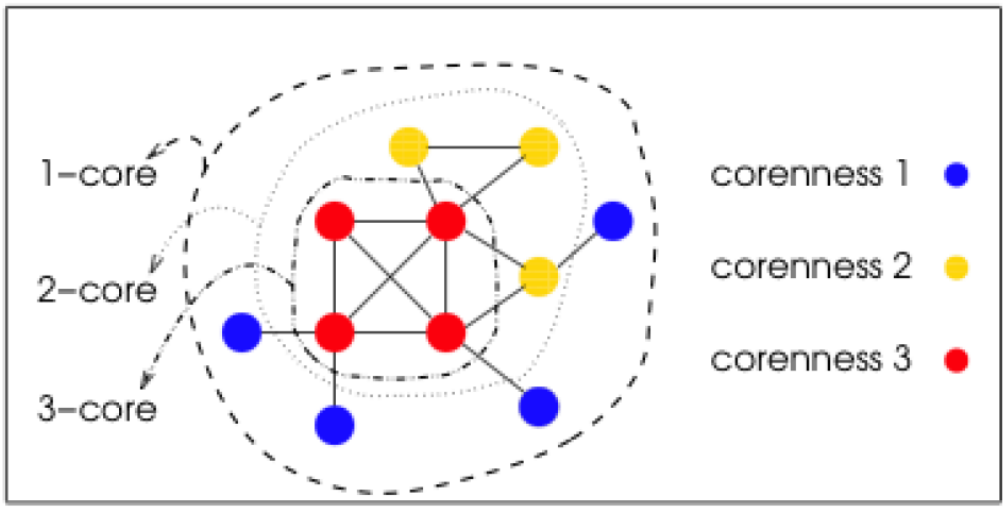
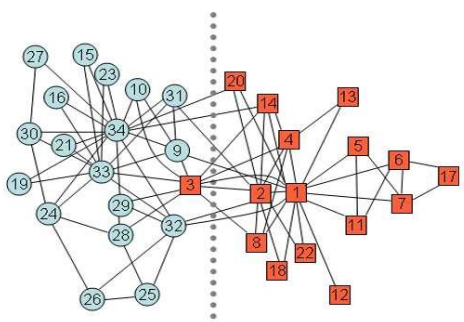


Figure: Example of k-core: Alvarez-Hamelin et al. (2005)

Classic Examples

Zachary's karate club

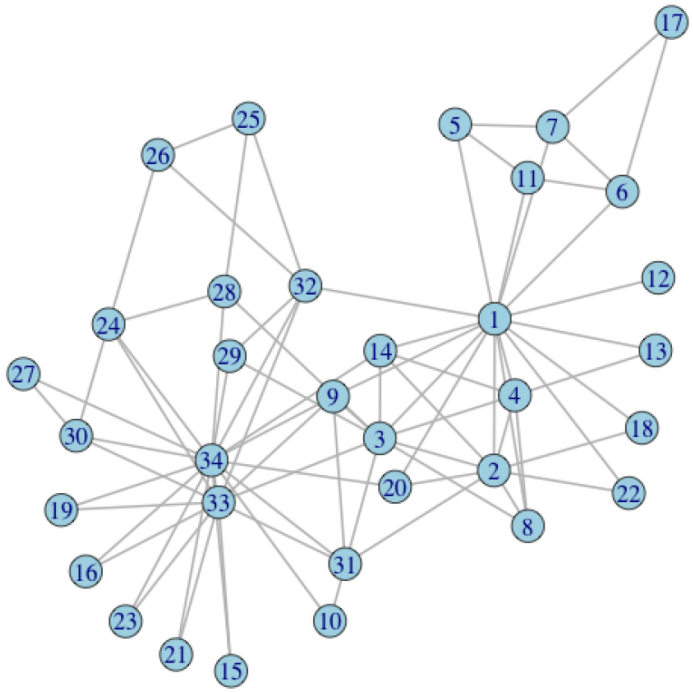
- Zachary's karate club is a social network of a university karate club, described in Zachary (1977) "An Information Flow Model for Conflict and Fission in Small Groups".[1]
- The network captures 34 members of a karate club, documenting links between pairs of members who interacted outside the club.
- During the study a conflict arose between the administrator "John A" and instructor "Mr. Hi", which led to the split of the club into two.



[1] It is available at http://tuvalu.santafe.edu/~aaronc/datacode.htm.

Classic Examples

Zachary's karate club

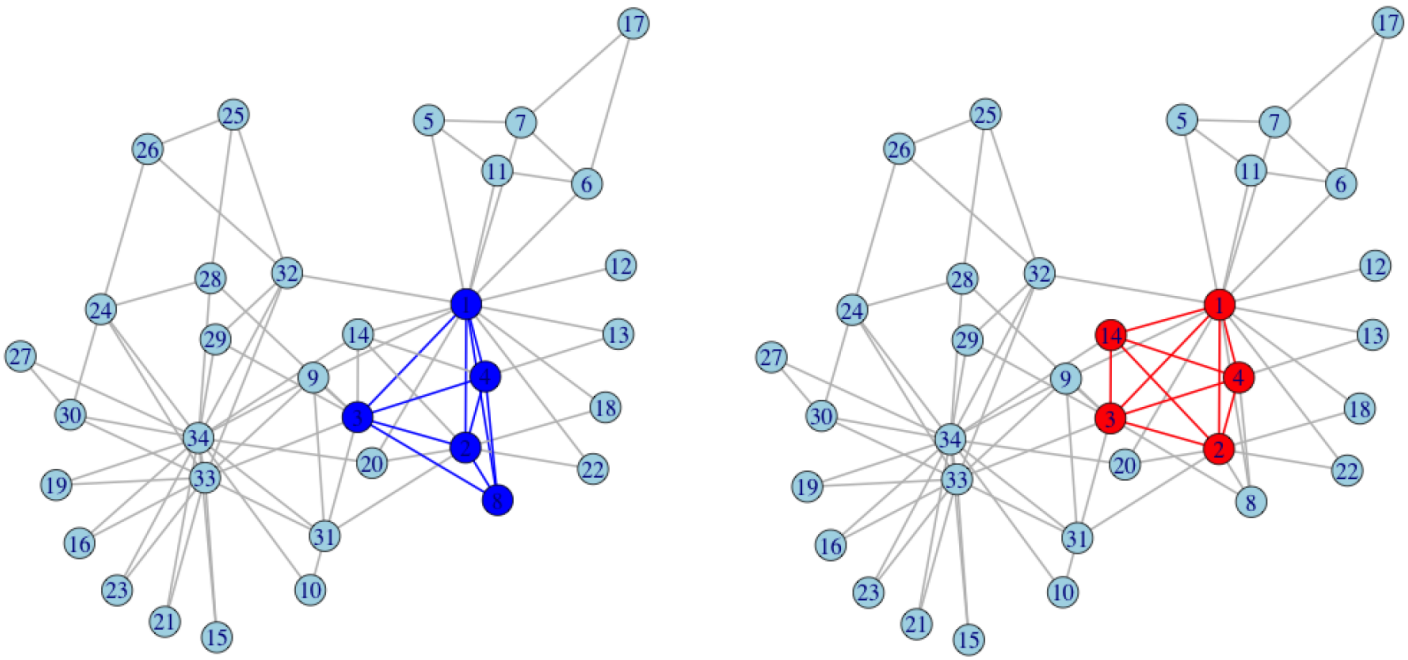


Maximal cliques:

Clique size:	2	3	4	5
Number of cliques:	11	21	2	2

Classic Examples

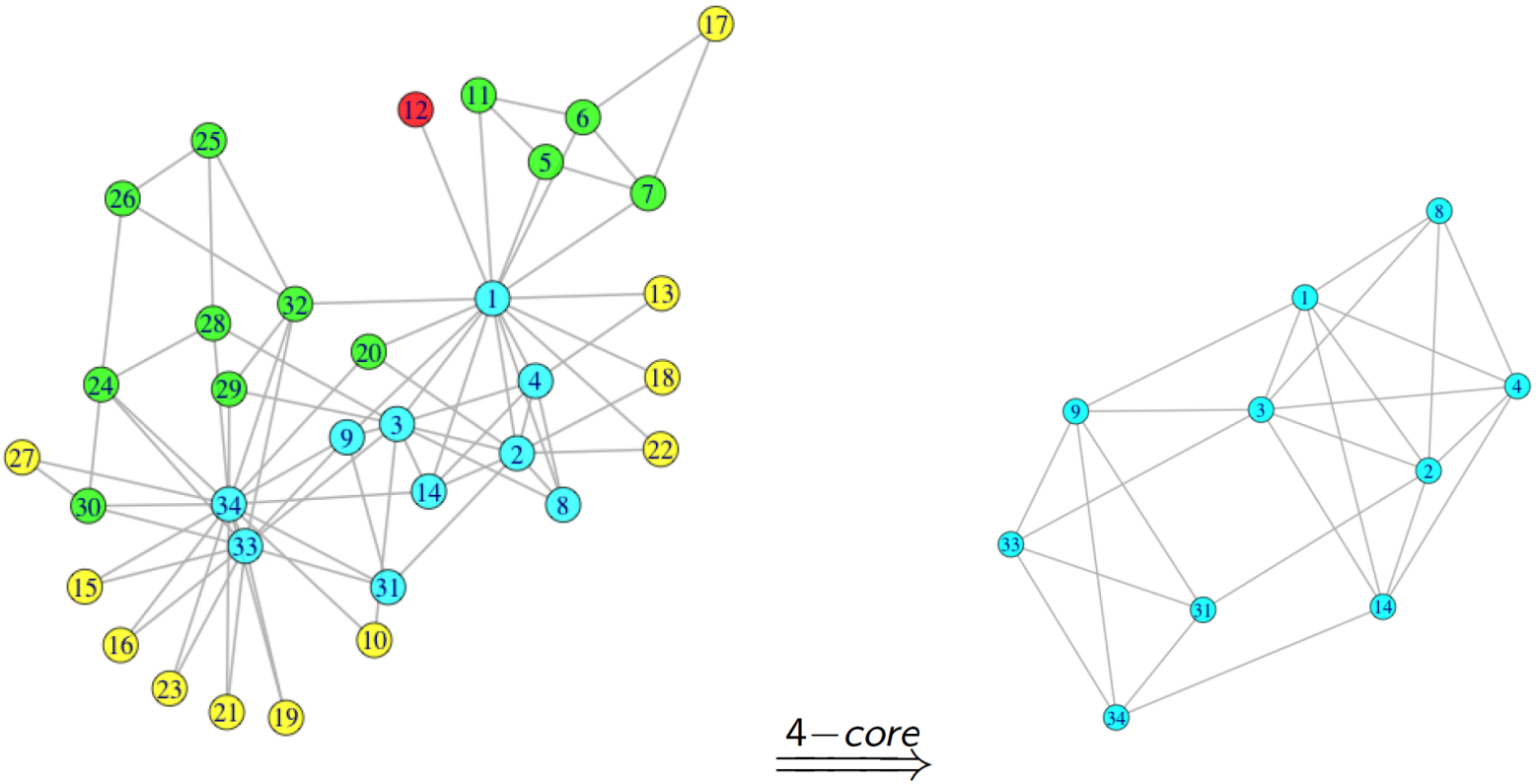
Zachary's karate club



Maximum cliques

Classic Examples

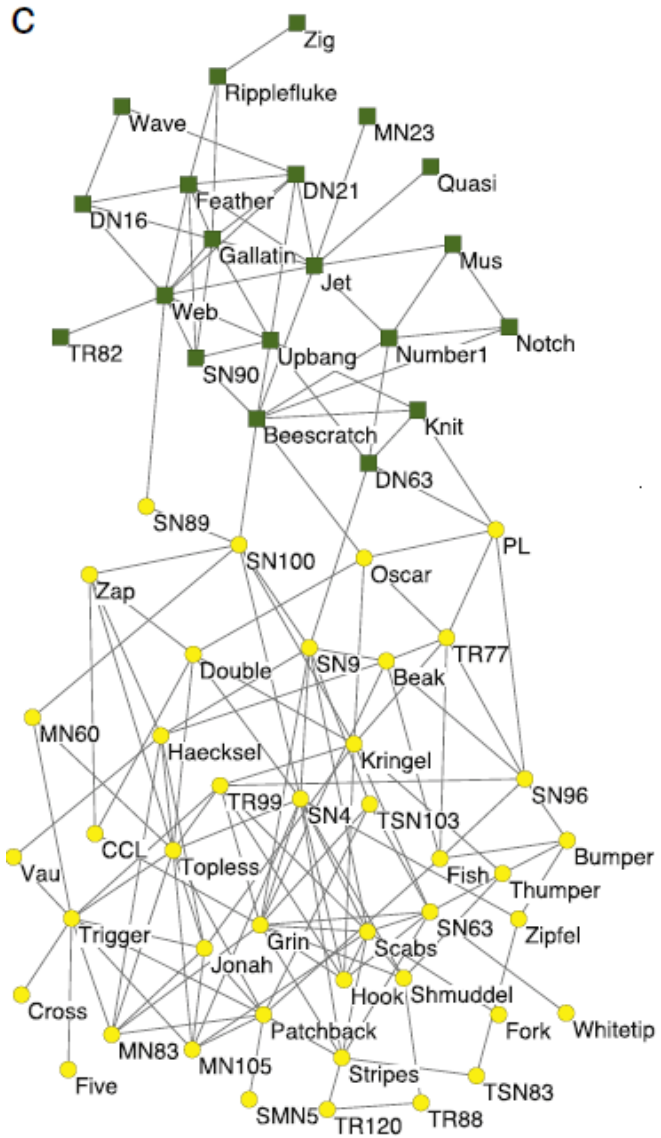
Zachary's karate club



Classic Examples

Social network of bottlenose dolphins

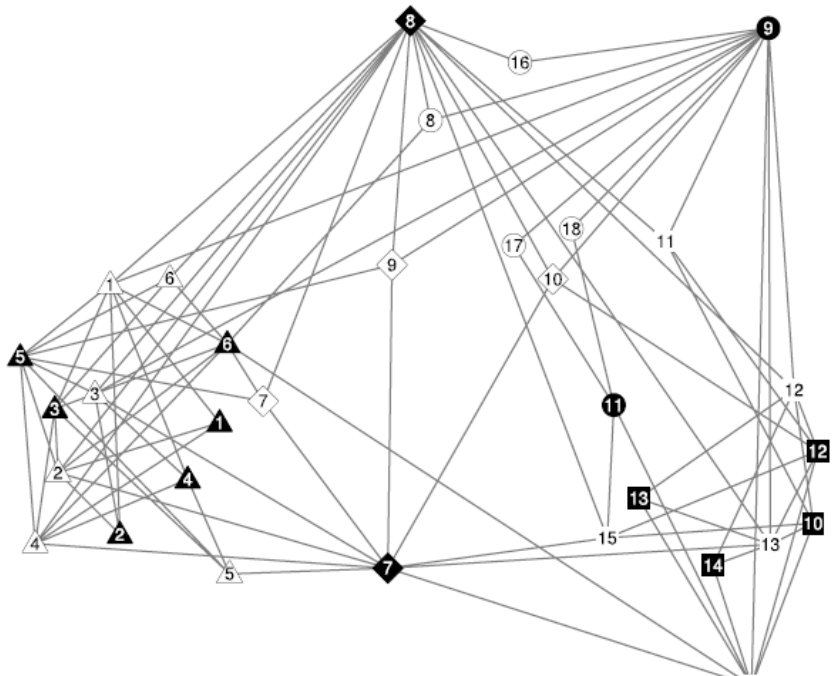
- Network of bottlenose dolphins living in Doubtful Sound (New Zealand) (Lusseau, 2003)
- Edges were set between animals that were seen together more often than expected by chance.



Classic Examples

Women event participation network

- Famous bipartite network of Southern Women studied by Davis et al. (1941).
- There are 32 vertices, representing 18 women from the area of Natchez, Mississippi, and 14 social events.
- The data is available in R package **tnet** as *Davis.Southern.women*
- Edges represent the participation of the women in the events.



Traditional Methods

Graph Partition method

- The problem of graph partitioning consists of dividing the vertices in predetermined g groups of predefined size, such that the number of edges lying between the groups is minimal.
- Most graph partitioning problems are NP-hard.
- The Kernighan-Lin algorithm is the frequently used method for graph partition. See illustration on https://www.youtube.com/watch?v=GsMZYDBFJv4.
- For the purpose of community detection, algorithms of graph partitioning are not good because it is necessary to provide as inputs the number of groups and in cases even their sizes, about which in principle one knows nothing.

Traditional Methods

Partitional Clustering method – k means

- Partitional clustering (or partitioning clustering) are clustering methods used to classify observations, within a data set, into pre-assigned k groups based on their distance.
- The goal is to separate the vertices into k clusters in order to minimize a given cost function based on distances between vertices and/or from points to *centroids*, i.e., suitably defined positions.
- One most popular partitional technique in the literature is K-means clustering (MacQueen et al., 1967)
- The cost function is the total intra-cluster distance, or squared error function

$$\sum_{i=1}^{k} \sum_{\mathbf{x_j} \in S_i} \|\mathbf{x_j} - \mathbf{c_i}\|^2,$$

S_i indicates the subset of points of the i^{th} cluster and $\mathbf{c_i}$ is its centroid.

Traditional Methods

Partitional Clustering method – k means

- The k-means problem can be easily solved by the Lloyd's algorithm (Lloyd, 1982).
- One starts from an initial distribution of centroids such that they are as far as possible from each other.
- In the first iteration, each vertex is assigned to the nearest centroid.
- Next, the centers of mass of the k clusters are estimated and become a new set of centroids, which allows for a new classification of the vertices (by comparing the cost function), and so on.
- After a small number of iterations, the positions of the centroids will be stable.
- One issue of k-means algorithm is that the solution found may not be global optimal – the result can be improved by performing more runs starting from different initial conditions.

Traditional Methods

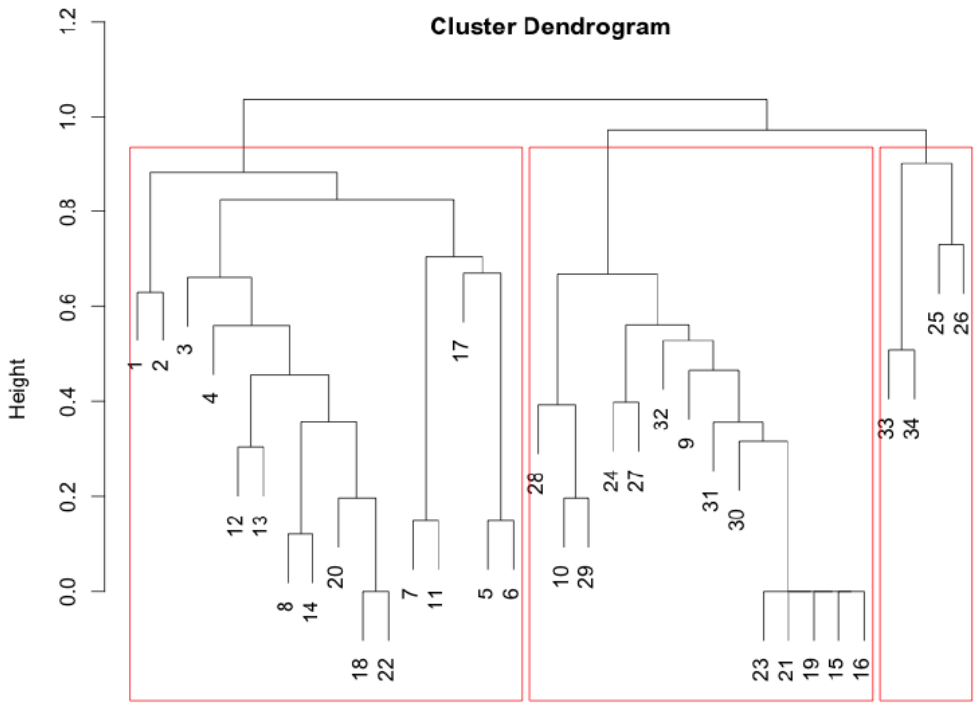
Hierarchical Clustering method

- Hierarchical clustering (HC) algorithms are clustering techniques that reveal the multilevel structure of the graph.
- HC algorithms start from measuring the similarity between vertices (there are several possible definitions of similarity).
- HC algorithms aim at identifying groups of vertices with similarity, and can be classified in two categories:
  - **Agglomerative algorithm**: This is a "bottom-up" approach. Each vertex starts in its own community, and pairs of vertices are merged as one moves up the hierarchy.
  - **Divisive algorithm**: This is a "top-down" approach. All vertices start in one community, and splits are performed recursively as one moves down the hierarchy.

Traditional Methods

Hierarchical Clustering method

The HC procedure can be better illustrated by means of dendrograms: Sometimes stopping conditions are imposed to select a partition or a group of partitions that satisfy a special criterion, like a given number of clusters or optimization of a quality function (e.g., modularity).



Divisive Algorithm

- One idea to identify communities in a graph is to detect the edges that connect vertices of different communities and remove them.
- The most popular algorithm is that proposed by Girvan and Newman (2002), where edges are selected according to the measures of *edge centrality*, estimating the importance of edges according to some property of process running on the graph.
- Algorithm:
  1. Computation of the centrality for all edges;
  2. Removal of edge with largest centrality: in case of ties with other edges, one of them is picked at random;
  3. Recalculation of centralities on the running graph;
  4. Steps 2 and 3 are repeated until no edges remain.

Divisive Algorithm

- Girvan and Newman focused on the concept of *betweenness*.
- The end result of the Girvan–Newman algorithm is a dendrogram.
- As the Girvan–Newman algorithm runs, the dendrogram is produced from the top down (i.e., the network splits up into different communities with the successive removal of links).
- A disadvantage of Girvan–Newman is that it is very computationally expensive, $O(n^3)$.

Modularity-based methods

- Modularity was first proposed by Newman and Girvan (2004) to define a stopping criterion for the algorithm of discovering community structure in networks, and has rapidly become an essential element of many community detection methods.
- Modularity is the fraction of the edges that fall within the given groups minus the expected fraction if edges were random.
- Modularity:

$$Q = \frac{1}{2m} \sum_{ij} \left(A_{ij} - \frac{d_i d_j}{2m} \right) \delta(c_i, c_j),$$

where m is the number of edges in the network, A_{ij} is the (i, j) element of adjacency matrix A, d_i is the degree of vertex i, c_i is the community label of node i, $\delta(c_i, c_j)$ is Kronecker delta.

$$\delta(c_i, c_j) = \begin{cases} 0 & \text{if } c_i \neq c_j \\ 1 & \text{if } c_i = c_j \end{cases}$$

Modularity-based methods

- Its value lies in the range $[-0.5, 1]$, positive when the number of edges within groups exceeds the number expected on the basis of chance.
- The higher the modularity, the more close-knit is the community.
- Biological networks, including animal brains, exhibit a high degree of modularity.
- Newman and Girvan (2004) use the modularity as the objective metric to choose the optimal number of communities that the divisive algorithm proposed by Girvan and Newman (2002) should divide.
- A shortcoming of modularity is that it suffers a resolution limit and, therefore, it is unable to detect small communities.

Modularity-based methods

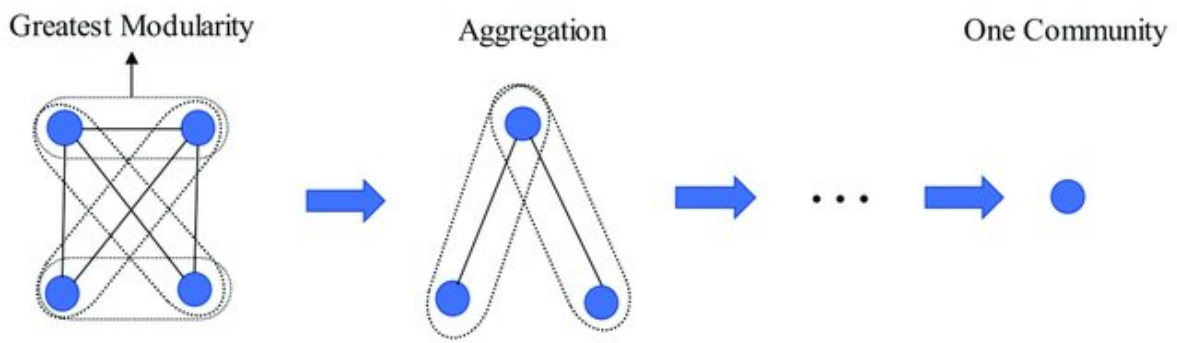
Fast Greedy algorithm

- The Fast Greedy algorithm (Newman, 2004) is an agglomerative hierarchical method.[2]
- It starts with a state in which each node is in its own community and the algorithm repeatedly joins pairs of communities together to obtain larger ones.
- At each step, the joined communities are selected by considering the largest increase (or smallest decrease) in modularity.
- FastGreedy produces a set of community structures organized hierarchically, with increasing granularity. The one obtaining the maximal modularity is considered as the best.

[2] A greedy algorithm is a simple, intuitive algorithm that is used in optimization problems. The algorithm makes the optimal choice at each step as it attempts to find the overall optimal way to solve the entire problem. https://en.wikipedia.org/wiki/Greedy_algorithm

Modularity-based methods

Fast Greedy algorithm



Modularity-based methods

Louvain Method

- Louvain method (Blondel et al., 2008) is also a agglomerative hierarchical clustering (greedy optimization) method that attempts to optimize the "modularity" of a partition of the network.
- Louvain method obtains its name because the method was devised when all coauthors were at the Université catholique de Louvain.
- It is slightly different from the greedy optimization process, and includes an additional aggregation step to improve processing on large networks.
- Louvain Method has been shown to be a simple, efficient, and scalable method for identifying communities in large networks. Examples include networks with 100 million nodes and billions of links.

Modularity-based methods

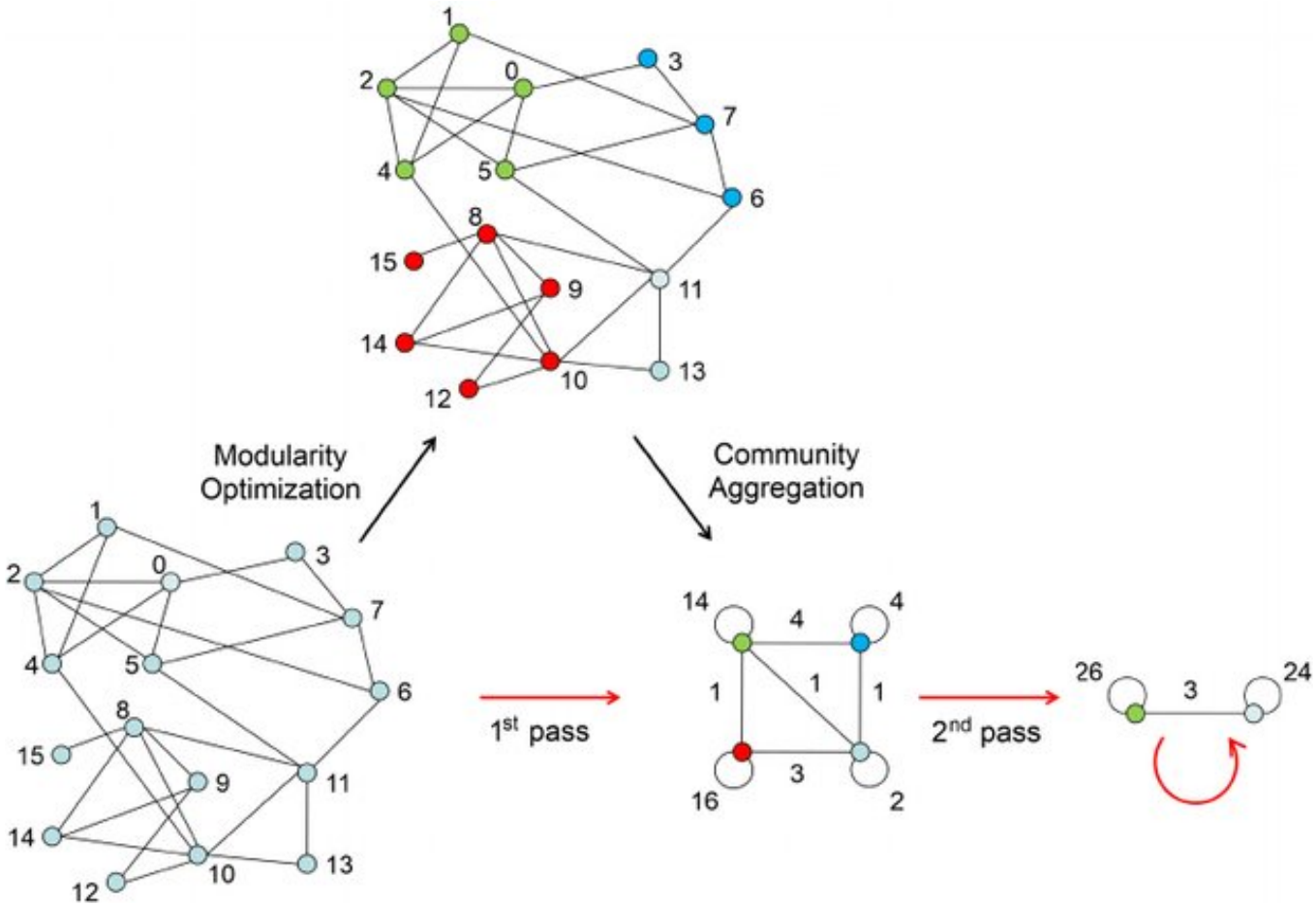
Louvain Method

Algorithms:

- Assign every node to its own community
- Phase 1
  - For every node evaluate modularity gain of removing node from its community and placing it in the community of its neighbor
  - move the node to the community maximizing the modularity gain, or stays in its original community if no gain is possible.
  - repeat until no more improvement (local max of modularity)
- Phase 2
  - Nodes from communities merged into “super nodes”
  - The inter- and intra-community links are represented in the new network by weighted regular links and self-loops, respectively
- Repeat until no more changes (max modularity)

Modularity-based methods

Louvain Method



Random Walk method

Walktrap Method

- Pons and Latapy (2006) proposed the Walktrap approach to detect community structures in networks.
- It is based on the fact that a random walker tends to be trapped in dense part of a network corresponding to communities.
- Algorithm:
  - Assign each vertex to its own community
  - Compute distance between adjacent vertices (based on the concept of random walks proposed in their paper).
  - Vertices are then grouped into communities through an agglomerative hierarchical clustering technique.

  Modularity is used to select the best partition of the resulting dendrogram.
- The expected complexity in practical computations is $O(n^2 log(n))$.

Label Propagation Algorithm

- The main idea behind LPA (Raghavan et al., 2007) is to propagate the labels of node throughout the network and form communities through the process of label propagation.
- At first, each node is initialized with a unique label denoting the community it belongs to.
- Then, every node updates its label iteratively. At every step of iteration, a node updates its label as most of its adjacent neighbors currently have.

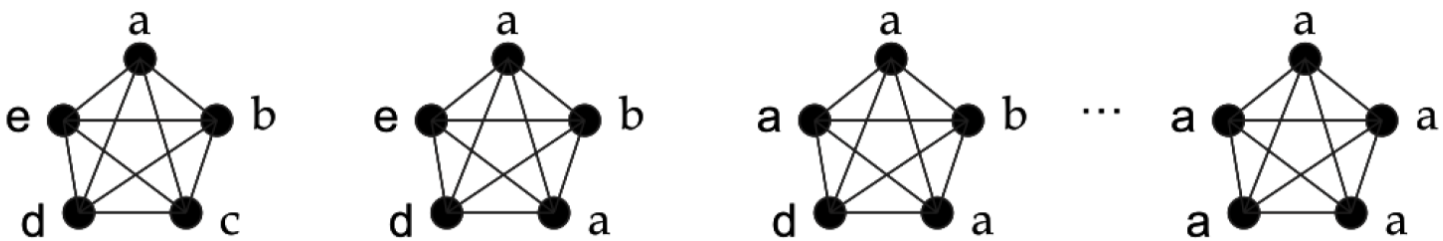
Label Propagation Algorithm

The LPA runs as follows:

- At the beginning, each node is initialized with a unique label, e.g., $L_i = i$.
- At every step of iterations, each node updates its label to the one shared by the most of its neighbors, $L_i = argmax_k |N^k(i)|$, where $N^k(i)$ denotes the set of neighbors of node i that have the label k.
- If more than one label are the most frequent ones, a new label is chosen randomly from them.
- The update-order of nodes is also random in iterations.
- The LPA is performed iteratively until each node has the most frequent label of its neighbors, that is, none of the nodes need to change it label.

The expected complexity in practical computation is $O(rm)$ where r is the number of iterations and m is the number of edges.

Label Propagation Algorithm



Infomap Algorithm

- The procedure of the Infomap algorithm (Rosvall and Bergstrom, 2008) is in the core identical to the Louvain method.
- The algorithm repeats the two described phases until an objective function is optimized.
- However, as an objective function to be optimized, Infomap does not use modularity but the so-called "map equation".
- The map equation exploits the duality between finding community structures in networks and minimizing the "description length" of the motion of a random walker.
- The map equation is designed to compress the descriptive length of the random walk when the random walker lasts for extended periods of time in certain regions of the network.

Infomap Algorithm

- This random walker randomly moves from vertex to vertex in the network. The more the connections of an vertex is weighted, the more likely the random walker will use its connections to reach the next vertex.
- The goal of Infomap is to form clusters in which the random walker stays as long as possible, i.e., the weights of the connections within the cluster should take on greater values than the weights of the connections between objects of different clusters.
- An insight behind the Infomap algorithm is to use community partitions of the graph as a Huffman code that compresses the information about a random walker exploring the graph.

Conglomerate Formation in China

- This example is based on Bai, Hsieh, Song, and Wang (2019) "Conglomerate Formation in China."
- China's growth is a puzzle to many researchers – why China's economy can grow so rapidly under such a closed political system with many institutional frictions?
- This paper proposes one possible answer: conglomerate acts as a special institutional arrangement to overcome economic and institutional frictions.
- This paper provides facts from firm ownership networks in China's economic universe with 17 million firms.

Conglomerate Formation in China

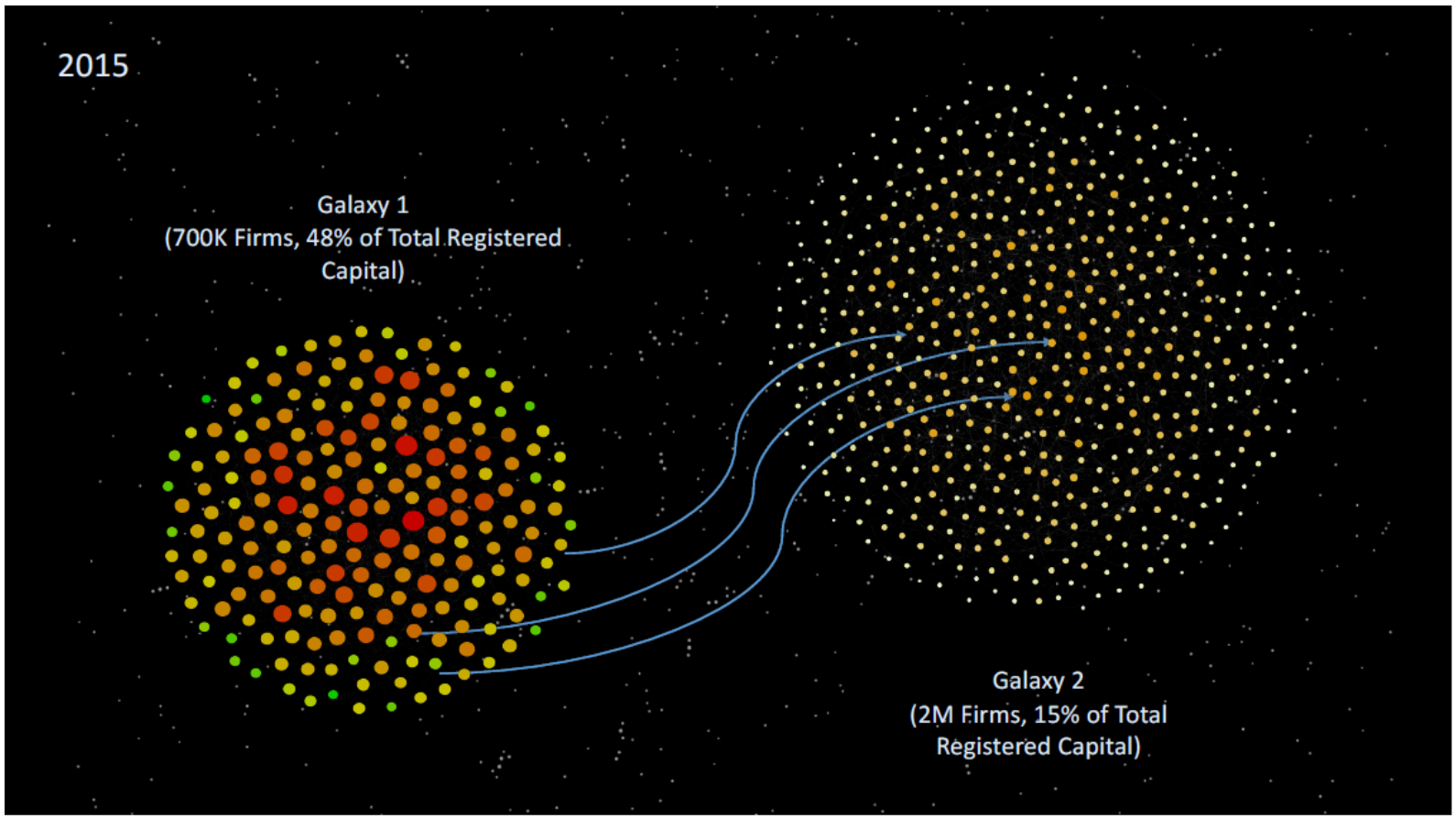
China's firm ownership network data

- Firm registration data of **State Administration for Industry and Commerce**.
- All firms, including holding companies (28 million firms, 11 million exited).
- Owner can be individual or legal persons (firms or holding companies).
- For owners, they know
  - Name and ID of legal person and individuals
  - Equity share in 2015 of each owner
  - Change in legal person owners from 2004 to 2014 for 11 provinces (however, no equity information).
- Registered capital, year of establishment, exit year are also known.
- Matched with China NBS (national Bureau of Statistics) on industrial firms.

Conglomerate Formation in China

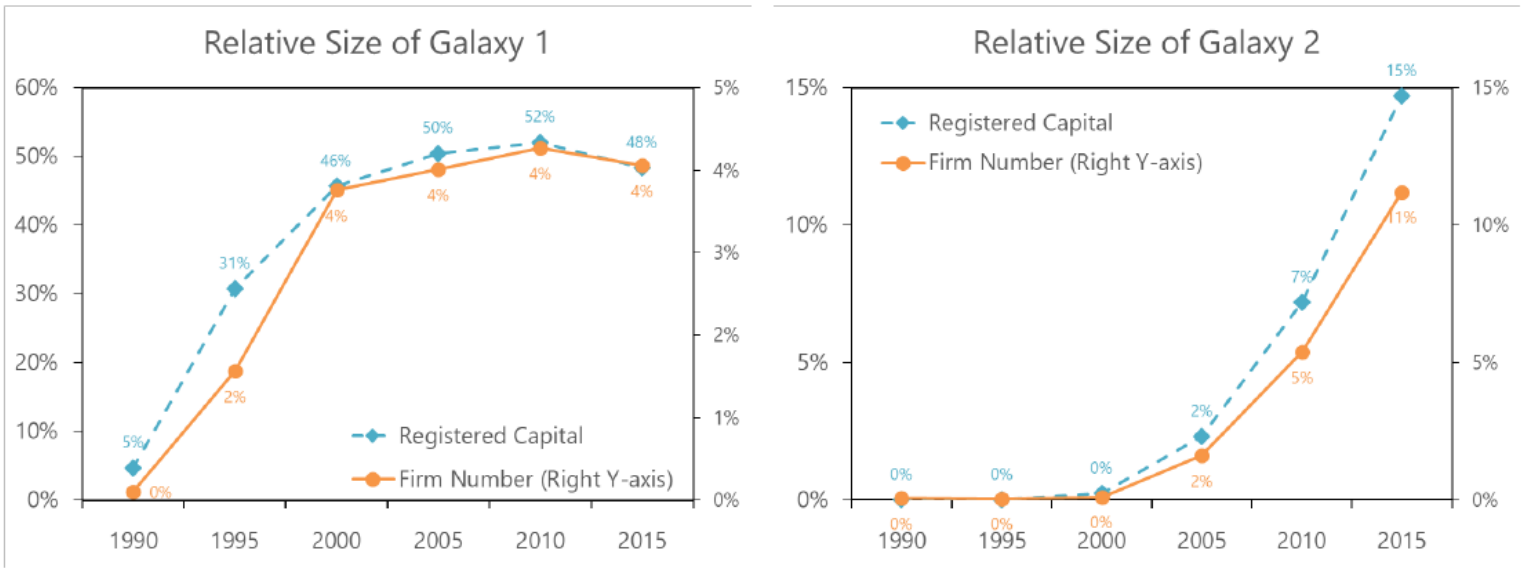
- Firms are connected by legal person ownership or individual person ownership.
- They call the giant component of the network connected by legal persons "Galaxy 1" and the giant component of the network connected by individual persons "Galaxy 2"
- There are firms appearing in both galaxies.

Conglomerate Formation in China



Conglomerate Formation in China

Size of Galaxy 1 and 2 Relative to the Universe

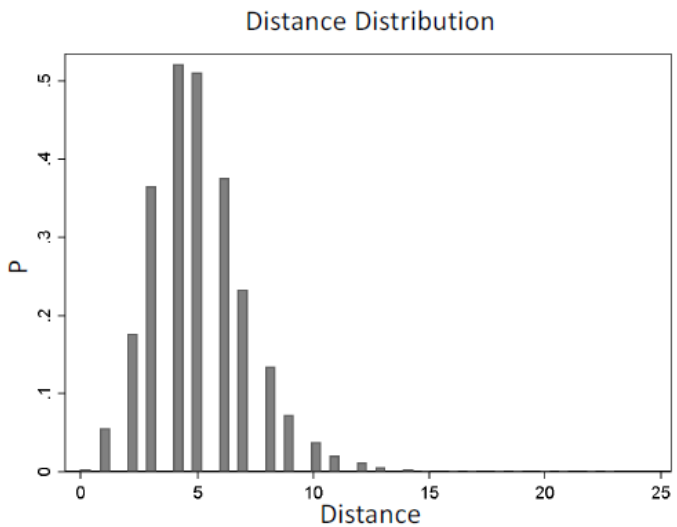


- Galaxy 1 grew rapidly in 1990 and stabilized after 2000, and Galaxy 2 grew after 2000.

Conglomerate Formation in China

Some stylized facts about Galaxy 1:

- Correlations between closeness and the distance to the center of the galaxy.
- Center of G1 (by closeness): 772 big SOEs (state owned enterprises)



Correlation between closeness and distance: -0.86
Closeness(i) = 1 / Sum(distance(i,j), j≠i), standardized into [0,1].
Distance: distance to the set of core firms (central and provincial SOEs).

Conglomerate Formation in China

State-Centered $\neq$ State-Controlled

		Direct + Indirect Equity Shares		Controlling Shareholder	
		Firm Number	RC share in Galaxy 1	Firm Number	RC share in Galaxy 1
772 SOEs	Threshold	772	10.5%	772	10.5%
Firms Controlled by 772 SOEs	50%	40,234	26.0%	43,943	26.6%
	25%	52,446	29.9%	57,897	30.9%
	10%	63,678	32.8%	73,457	34.0%
	5%	69,385	33.9%	81,524	35.7%

Conglomerate Formation in China

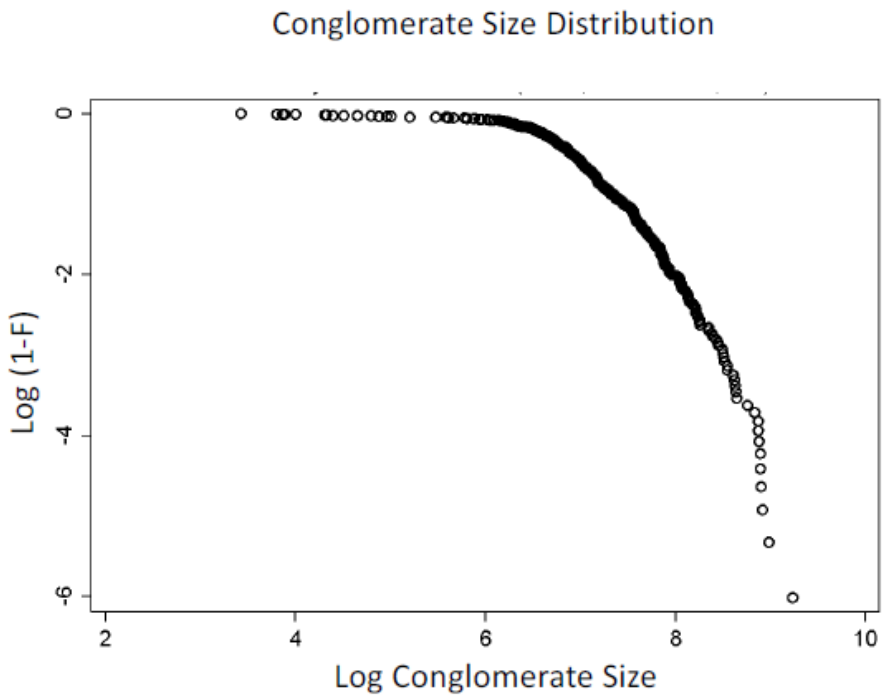
Identify "Conglomerates" in the Firm Network

- Identify communities (conglomerates) in the first galaxy (700k firms)
- Method: Louvain algorithm ("Fast Unfolding of Communities in Large Networks", Blondel et al., 2008)
- Results: 413 communities (or conglomerates) identified, with modularity = 0.94.
- Other algorithms: infomap (m=0.86), walktrap (m=0.83) and label propogation (m=0.82).

Conglomerate Formation in China

Summary Statistics

	Mean	Median	Std	Min	Max
Firm Number	1702.8	1187.0	1544.0	31.0	11014.0
RC (0.1 Billion Yuan)	1544.4	686.8	2491.1	0.9	23098.5



Conglomerate Formation in China

Growth of Conglomerates

Year	Number of Conglomerate	Mean of Conglomerate Size
1995	212	245.4
2000	308	512.5
2005	334	731.5
2010	372	1088.1
2015	413	1702.8

Conglomerate Formation in China

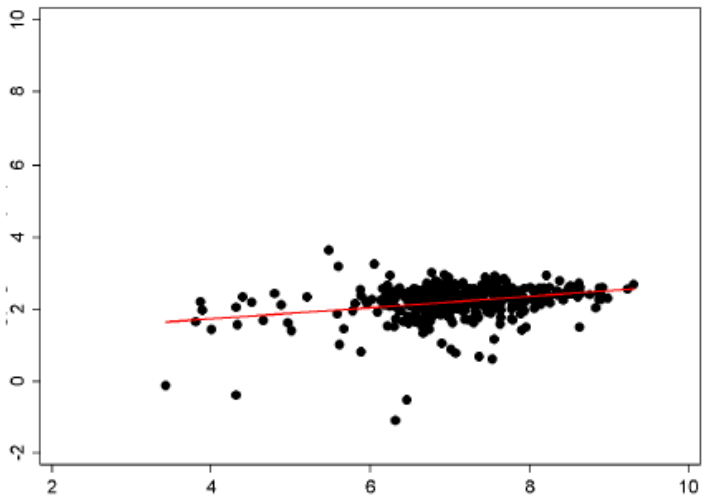
State-Centered Conglomerates

- 772 central and provincial SOEs in 210 conglomerates
- These SOEs are in the center of their conglomerates (with much higher closeness, larger size and lower YK ratio).
- Closeness within conglomerate is highly correlated with distance to the 772 SOEs (correlation: -0.72).
- The 210 "state-centered" conglomerates account for two-thirds of Galaxy 1 by firm number and registered capital.

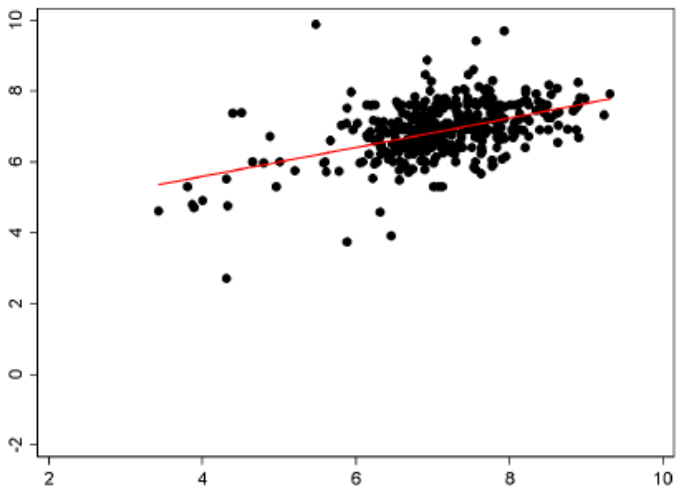
Conglomerate Formation in China

Conglomerate Size vs. Average/Top Firms

Average Log RC in a Conglomerate



The 99th Percentile Log RC in a Conglomerate



Conglomerate Formation in China

Summary of the Stylized Facts

- fast growing of Galaxy 1 and 2 in the universe of China firms.
- In Galaxy 1, growth is mainly driven by the expansion of incumbent conglomerates.
- Firm size, age and SOE share are negatively correlated with distance to the center of the galaxy, but the YK ratio is positively correlated with the distance.
- Among conglomerates (communities):
  - Strong correlations between conglomerate size and the top firm size or the bottom YK ratio **between** conglomerates.
  - Firm size, age and SOE share (YK) are negatively (positively) correlated with distance to the center of a conglomerate.

Alvarez-Hamelin, José Ignacio, Luca Dall'Asta, Alain Barrat, and Alessandro Vespignani (2005) "K-core decomposition of internet graphs: hierarchies, self-similarity and measurement biases," *arXiv preprint cs/0511007*.

Blondel, Vincent D, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre (2008) "Fast unfolding of communities in large networks," *Journal of Statistical Mechanics: Theory and Experiment*, Vol. 2008, p. P10008.

Davis, Allison, Burleigh Bradford Gardner, and Mary R Gardner (1941) *Deep South: A social anthropological study of caste and class*: Univ of South Carolina Press.

Fortunato, Santo (2010) "Community detection in graphs," *Physics reports*, Vol. 486, pp. 75–174.

Girvan, Michelle and Mark EJ Newman (2002) "Community structure in social and biological networks," *Proceedings of the national academy of sciences*, Vol. 99, pp. 7821–7826.

Harenberg, Steve, Gonzalo Bello, La Gjeltema, Stephen Ranshous, Jitendra Harlalka, Ramona Seay, Kanchana Padmanabhan, and Nagiza Samatova (2014) "Community detection in large-scale networks: a survey and empirical evaluation," *Wiley Interdisciplinary Reviews: Computational Statistics*, Vol. 6, pp. 426–439.

Lloyd, Stuart (1982) "Least squares quantization in PCM," *IEEE transactions on information theory*, Vol. 28, pp. 129–137.

Lusseau, David (2003) "The emergent properties of a dolphin social network," *Proceedings of the Royal Society of London. Series B: Biological Sciences*, Vol. 270, pp. S186–S188.

MacQueen, James et al. (1967) "Some methods for classification and analysis of multivariate observations," in *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, Vol. 1, pp. 281–297, Oakland, CA, USA.

Newman, Mark EJ (2004) "Fast algorithm for detecting community structure in networks," *Physical Review E*, Vol. 69, p. 066133.

Newman, Mark EJ and Michelle Girvan (2004) "Finding and evaluating community structure in networks," *Physical review E*, Vol. 69, p. 026113.

Orman, Günce Keziban, Vincent Labatut, and Hocine Cherifi (2012) "Comparative evaluation of community detection algorithms: a topological approach," *Journal of Statistical Mechanics: Theory and Experiment*, Vol. 2012, p. P08001.

Pons, Pascal and Matthieu Latapy (2006) "Computing communities in large networks using random walks.," *J. Graph Algorithms Appl.*, Vol. 10, pp. 191–218.

Raghavan, Usha Nandini, Réka Albert, and Soundar Kumara (2007) "Near linear time algorithm to detect community structures in large-scale networks," *Physical review E*, Vol. 76, p. 036106.

Rosvall, Martin and Carl T Bergstrom (2008) "Maps of random walks on complex networks reveal community structure," *Proceedings of the National Academy of Sciences*, Vol. 105, pp. 1118–1123.

Zachary, Wayne W (1977) "An information flow model for conflict and fission in small groups," *Journal of Anthropological Research*, Vol. 33, pp. 452–473.