

Graph Clustering Algorithms

Andrea Marino

PhD Course on Graph Mining Algorithms,
Università di Pisa

February, 2018

Clustering: Intuition to Formalization

Task

Partition a graph into natural groups so that the nodes in the same cluster are more close to each other than to those in other clusters.

Paradigm

Intra-cluster density vs. inter-cluster sparsity

Mathematical Formalization

Quality measures for clusterings

- Many exist, optimization generally (NP-)hard
- More constraints: clusterings often "nice" if balanced
- There is no single, universally best strategy

Given a graph and a clustering, a quality measure should behave as follows:

- more intra-edges \Rightarrow higher quality
- less inter-edges \Rightarrow higher quality
- cliques must never be separated
- clusters must be connected
- disjoint cliques should approach maximum quality
- double the instance, what should happen . . . same result

A Theorem of Impossibility

A warning theorem on the field of data clustering:

Theorem (Jon Kleinberg: An Impossibility Theorem for Clusterings, 2002)

Given set S . Let $f : d \rightarrow \Gamma$ be a function on a distance function d on set S , returning a clustering Γ . No function f can simultaneously fulfill the following.

- *Scale-Invariance: for any distance function d and any $\alpha > 0$, we have $f(d) = f(\alpha \cdot d)$.*
- *Richness: for any given clustering Γ , we should be able to define a distance function d such that $f(d) = \Gamma$.*
- *Consistency: if we build d' from d by reducing intra-distances and increasing inter-distances, we should have $f(d') = f(d)$.*

- Cut-based Measures
 - thickness of bottleneck
 - worst bottleneck induced
 - best bottleneck still left uncut inside some cluster
- Counting Measures
 - fraction of covered edges
 - modularity: how clear is the clustering, compared to random network

Cut-based Measures: Conductance and Expansion

- conductance of a cut $(C, V \setminus C)$: thickness of bottleneck which cuts off C

$$\phi(C, V \setminus C) = \frac{E(C, V \setminus C)}{|C||V \setminus C|}$$

- expansion of a cut $(C, V \setminus C)$:

$$\frac{E(C, V \setminus C)}{\min\{\sum_{u \in C} \deg(u), \sum_{u \in V \setminus C} \deg(u)\}}$$

intra e inter-cluster expansion analogously

- Criterion: average or minimum/maximum ?

Cut-based Measures: Conductance and Expansion

- inter-cluster conductance (\mathcal{C}): worst bottleneck induced by some $C \in \mathcal{C}$

$$1 - \max_{C \in \mathcal{C}} \phi(C, V \setminus C)$$

- intra-cluster conductance (\mathcal{C}): best bottleneck still left uncut inside some $C \in \mathcal{C}$

$$\min_{C \in \mathcal{C}} \min_{P \cup Q = C} \phi(C(P, Q))$$

- coverage: fraction of covered edges

$$\text{cov}(\mathcal{C}) = \frac{\textit{intracluster edges}}{\textit{edges}}$$

- performance: fraction of correctly classified pairs of nodes

$$\text{perf}(\mathcal{C}) = \frac{\textit{intracluster edges} + \textit{absent intercluster edges}}{\frac{1}{2}n(n-1)}$$

- density: fractions of correct intra- and inter-edges

$$\text{den}(\mathcal{C}) = \frac{1}{2} \frac{\text{intracluster edges}}{\text{possible intracluster edges}} + \frac{1}{2} \frac{\text{absent intercluster edges}}{\text{possible intercluster edges}}$$

- modularity: how clear is the clustering, compared to random network

$$\text{mod}(\mathcal{C}) := \text{cov}(\mathcal{C}) - \mathbb{E}[\text{cov}(\mathcal{C})]$$





- Criterion: average or minimum/maximum ?

Optimization of quality function:

- Bottom-up:
 - start with singletons and merge clusters
- Top-down:
 - start with the one-cluster and split clusters
- Local Opt.:
 - start with random clustering and migrate nodes

Possible Implementations:

- Variants of recursive min-cutting
- Percolation of network by removal of highly central edges
- Direct identification of dense substructures
- Random walks
- Geometric approaches
- ...

-  Brandes, Erlebach (eds.) 2005, Network Analysis, Methodological Foundations
-  Satu Elisa Schaeffer: Graph Clustering, 2007
-  Santo Fortunato: Community Structure in Graphs, 2009
-  Robert Gorke: An algorithmic walk from static to dynamic graph clustering, 2010

Main Idea

Every cluster is identified by a center and at the beginning start with k arbitrary disjoint sets. Iteratively, calculate center of the partition and modify these partitions adding closest nodes.

Given an initial set of k means $m_1^{(1)}, \dots, m_k^{(1)}$, the algorithm proceeds by alternating between two steps:

- 1 Assignment step: assign each observation to the cluster with the closest mean (i.e. partition the observations according to the Voronoi diagram¹ generated by the means).
- 2 Update step: calculate the new means to be the centroid of the observations in the cluster.

¹To each mean one associates a corresponding Voronoi cell, namely the set of all points in the given mean whose distance to the given object is not greater than their distance to the other means.

The algorithm is deemed to have converged when the assignments no longer change.

Several variations possible:

- K -means: minimize the distance squared
- K -center: minimize the maximum distance
- K -median: minimize the average distance

Greedy Global Agglomeration

- 1 Start: singletons
 - 2 iterative agglomerations, yielding highest gain in quality (or least decrease)
 - 3 result: best intermediate clustering
- The objective function has to be bounded²
 - and connected, i.e. if merging unconnected clusters is never the best option with respect to f .

²An objective function measure f is unbounded if for any clustering \mathcal{C} with $|\mathcal{C}| > 1$ there exists a merge that does not deteriorate f .

An example of Greedy Global Agglomeration is Greedy Significance:

- 1 For a given significance measure S starts with the singleton clustering
- 2 Iteratively merge those two clusters that yield largest increase or the smallest decrease in significance.
- 3 After $n - 1$ merges the clustering that achieved the highest significance is returned.

The algorithm maintains a symmetric matrix ΔS with entries $\Delta S_{i,j} = S(\mathcal{C}_{i,j}) - S(\mathcal{C})$, where \mathcal{C} is the current clustering and $\mathcal{C}_{i,j}$ is obtained from \mathcal{C} by merging clusters C_i and C_j .



Gaertler, Gorke, and Wagner, Significance-Driven Graph Clustering, 2007

Local Moving and Multilevel

- Locally greedy
- Node shifts:
 - nodes can change their cluster during the algorithm
- Hierarchical contractions



Blondel et al.: Fast unfolding of communities in large networks, 2008

Clustering with Minimum-Cut Tree

- Given a graph $G = (V, E)$, the min cut tree is defined on V and has the property that the minimum cut between two nodes s, t in G can be found by inspecting the path that connects s and t in T .
- For every undirected graph, there always exists a min-cut tree
- Require $O(n)$ computations of min-cut



Dan Gusfield (1990). "Very Simple Methods for All Pairs Network Flow Analysis". SIAM J. Comput. 19 (1): 143155.



Gomory, R. E.; Hu, T. C. (1961). "Multi-terminal network flows". Journal of the Society for Industrial and Applied Mathematics. 9.

Cut Clustering

- Given $G = (V, E)$ and α .
- Define $G' = (V \cup \{t\}, E \cup \{(v, t) : v \in V\})$ where all the edges (v, t) has weight α .
- Calculate the minimum-cut tree T' of G'
- Remove t from T'
- Return all the connected components as the clusters of G .
- α bounds the cut between each pair of clusters.



Flake, Tarjan, Tsioutsoulis, Clustering methods based on minimum-cut trees, 2002

- 1 Introduce decision variables
- 2 Ensure valid clustering with constraints (transitivity):
- 3 Reflexivity and symmetry for free
- 4 Optimize target function



Gorke: An algorithmic walk from static to dynamic graph clustering, 2010



Schumm et al.: Density-constrained graph clustering (technical report), 2011

Clique-Percolation

- It builds up the communities from k -cliques,
- Two k -cliques are considered adjacent if they share $k - 1$ nodes.
- A cluster is defined as the maximal union of k -cliques that can be reached from each other through a series of adjacent k -cliques.
- Overlapping clusters



Palla et al.: Uncovering the overlapping community structure of complex networks in nature and society, 2005

Network Percolation Clustering

- Iteratively remove most central edges in graph
- Stop at threshold
- Components induce clusters



Girvan and Newman: Finding and evaluating community structure in networks 2002

Markov Clustering Algorithm and Random Walks

- Simulate long random walk through graph
- Random Walks are calculated by Markov Chains



Stijn van Dongen, Graph Clustering by Flow Simulation. PhD thesis, University of Utrecht, May 2000.

Markov Clustering Algorithm

- 1 Normalize the adjacency matrix.
- 2 Expand by taking the e -th power of the matrix
 - $M^r[i, j]$ is the probability that starting from i after r steps a random walk is in j , where M is the normalized adjacency matrix.
- 3 Inflate by taking inflation of the resulting matrix with parameter r
 - Normalize again and any element of the matrix is multiplied by itself r times.
- 4 Repeat until a steady state is reached (convergence).

Variants using Spectral Clustering

Spectral graph theory

Spectral graph theory studies how the eigenvalues of the adjacency matrix of a graph, which are purely algebraic quantities, relate to combinatorial properties of the graph.

Spectral clustering studies the relaxed ratio sparsest cut through spectral graph theory.

Some variants project points using spectral graph theory.

- Project points into k -dimensional space and assign points to closest axes, or



Ravi Kannan, Santosh Vempala, Adrian Vetta: On clusterings: Good, bad and spectral. J. ACM 51(3): 497-515 (2004)

- Use k -means on embedding.



Jianbo Shi, Jitendra Malik: Normalized Cuts and Image Segmentation. IEEE Trans. Pattern Anal. Mach. Intell. 22(8): 888-905 (2000)

Part II

Sparsest Cut and Spectral graph theory

Sparsest Cut

- The sparsity of S is the ratio between the fraction of edges that need to be removed in order to disconnect S from $V - S$ and the fraction of pairs of vertices that would be so disconnected.
- The sparsest cut is the set of minimum sparsity.

Definition (Sparsest Cut)

Let $G = (V, E)$ be a graph and let $(S, V - S)$ be a partition of the vertices (a cut). Then the sparsity of the cut is^a.

$$\phi(S) := \frac{E(S, V - S)}{|E|} \cdot \left(\frac{|S| \cdot |V - S|}{|V|^2/2} \right)^{-1}$$

where $E(S, V - S)$ is the number of edges in E that have one endpoint in S and one endpoint in $V - S$.

The sparsity of a graph $G = (V, E)$ is

$$\phi(G) := \min_{S \subseteq V: S \neq \emptyset, S \neq V} \phi(S)$$

^aIt is more common to define the sparsity as $\frac{E(S, V - S)}{|S| \cdot |V - S|}$ without the normalizing factor $(V^2/2|E|)$; the normalized definition yields simpler formulas

If G is a d -regular graph

$$\phi(S) = \frac{E(S, V - S)}{\frac{d}{|V|} \cdot |S| \cdot |V - S|}$$

- $h(S)$ is the ratio between the number of edges between S and $V - S$ and the obvious upper bound given by the total number of edges incident on the smaller side of the cut.

$$h(S) = \frac{E(S, V - S)}{d \cdot \min\{|S|, |V - S|\}}$$

- The edge expansion $h(G)$ of a graph is the minimum of $h(S)$ over all non-trivial partitions $(S, V - S)$.

For every regular graph G , for every set S ,

$$\phi(S) \leq h(S) \leq 2 \cdot \phi(S)$$

The adjacency matrix

- If $G = (V, E)$ is a graph, the adjacency matrix A of G , is such that $A_{ij} = 1$ if $(i, j) \in E$ and $A_{ij} = 0$ otherwise.
- If G is a multigraph or a weighted graph, then A_{ij} is equal to the number of edges between (i, j) , or the weight of the edge (i, j) , respectively.

Theorem

if A is the adjacency matrix of an undirected graph then it has n real eigenvalues, counting multiplicities of the number of solutions to $\det(A - \lambda I) = 0$.

If G is a d -regular graph, the normalized matrix is $M := \frac{1}{d} \cdot A$.

Eigenvalues and Eigenvectors as Solutions to Optimization Problems

In order to relate the eigenvalues of the adjacency matrix of a graph to combinatorial properties of the graph, we need to first express the eigenvalues and eigenvectors as solutions to optimization problems, rather than solutions to algebraic equations.

Lemma

If M is a symmetric matrix and λ_1 is its largest eigenvalue, then

$$\lambda_1 = \sup_{\mathbf{x} \in \mathbb{R}^n: \|\mathbf{x}\|=1} \mathbf{x}^T M \mathbf{x}$$

The vectors achieving it are precisely the eigenvectors of λ_1 .

Lemma

If M is a symmetric matrix, λ_1 is its largest eigenvalue, and \mathbf{v}_1 is an eigenvector of λ_1 , then

$$\lambda_2 = \sup_{\mathbf{x} \in \mathbb{R}^n: \|\mathbf{x}\|=1, \mathbf{x} \perp \mathbf{v}_1} \mathbf{x}^T M \mathbf{x}$$

The vectors achieving it are precisely the eigenvectors of λ_2 .

Lemma

If M is a symmetric matrix and λ_n is its smallest eigenvalue, then

$$\lambda_n = \inf_{\mathbf{x} \in \mathbb{R}^n: \|\mathbf{x}\|=1} \mathbf{x}^T M \mathbf{x}$$

The vectors achieving it are precisely the eigenvectors of λ_n .

Theorem

Let G be a d -regular undirected graph, and $M = \frac{1}{d} \cdot A$ be its normalized adjacency matrix. Let $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n$ be the real eigenvalues of M with multiplicities. Then

- 1 $\lambda_1 = 1$.
- 2 $\lambda_2 = 1$ if and only if G is disconnected.
- 3 $\lambda_n \geq -1$ and $\lambda_n = -1$ if and only if at least one of the connected components of G is bipartite.

Cheeger's Inequalities

- $\lambda_2 = 1$ if and only if G is disconnected. This is equivalent to saying that $1 - \lambda_2 = 0$ if and only if $h(G) = 0$.
- This statement admits an approximate version that, qualitatively, says that $1 - \lambda_2$ is small if and only if $h(G)$ is small. Quantitatively, we have

Theorem (Cheeger's Inequalities)

$$\frac{1 - \lambda_2}{2} \leq h(G) \leq \sqrt{2 \cdot (1 - \lambda_2)}$$

Lemma (One Direction of Cheeger's Inequality)

$$1 - \lambda_2 \leq \phi(G)^a$$

^aSince $\phi(G) \leq 2h(G)$, $\frac{1-\lambda_2}{2} \leq h(G)$

Equivalent restatement of the sparsest cut problem. If represent a set $S \subseteq V$ as a bit-vector $x \in \{0, 1\}^V$, then

$$E(S, V - S) = \frac{1}{2} \cdot \sum_{ij} A_{ij} \cdot |x_i - x_j|$$

$$|S| \cdot |V - S| = \frac{1}{2} \cdot \sum_{ij} |x_i - x_j|$$

so that, after some simplifications, we can write

$$\phi(G) = \min_{x \in \{0,1\}^V - \{0,1\}} \frac{\sum_{ij} M_{ij} |x_i - x_j|}{\frac{1}{n} \sum_{ij} |x_i - x_j|}$$

Note that, when x_i, x_j take boolean values, then so does $|x_i - x_j|$, so that we may also equivalently write

$$\phi(G) = \min_{\mathbf{x} \in \{0,1\}^V - \{\mathbf{0}, \mathbf{1}\}} \frac{\sum_{ij} M_{ij} |x_i - x_j|^2}{\frac{1}{n} \sum_{ij} |x_i - x_j|^2}$$

We have the following characterization of $1 - \lambda_2$:

$$1 - \lambda_2 = \min_{\mathbf{x} \in \mathbb{R}^V - \{\mathbf{0}\}, \mathbf{x} \perp \mathbf{1}} \frac{\sum_{ij} M_{ij} |x_i - x_j|^2}{2 \cdot \sum_i x_i^2}$$

It is possible to prove that the following characterization is also true

$$1 - \lambda_2 = \min_{\mathbf{x} \in \mathbb{R}^V - \{\mathbf{0}, \mathbf{1}\}} \frac{\sum_{ij} M_{ij} |x_i - x_j|^2}{\frac{1}{n} \sum_{ij} |x_i - x_j|^2}$$

The quantity $1 - \lambda_2$ is a continuous relaxation of $\phi(G)$, and hence $1 - \lambda_2 \leq \phi(G)$.

Lemma (The Other Direction of Cheeger's Inequality)

$$h(G) \leq \sqrt{2 \cdot (1 - \lambda_2)}$$

The proof can be seen as an analysis of the following algorithm.

Algorithm: SpectralPartitioning

- Input: graph $G = (V, E)$ and vector $\mathbf{x} \in \mathbb{R}^V$
- Sort the vertices of V in non-decreasing order of values of entries in \mathbf{x} , that is let $V = \{v_1, \dots, v_n\}$ where $x_{v_1} \leq x_{v_2} \leq \dots \leq x_{v_n}$
- Let $i \in \{1, \dots, n-1\}$ be such that $h(\{v_1, \dots, v_i\})$ is minimal
- Output $S = \{v_1, \dots, v_i\}$

The last part algorithm can be implemented to run in time $O(|V| + |E|)$.^a

^abecause once we have computed $h(\{v_1, \dots, v_i\})$ it only takes time $O(\text{degree}(v_{i+1}))$ to compute $h(\{v_1, \dots, v_{i+1}\})$.

We have the following analysis of the quality of the solution:

Lemma (Analysis of Spectral Partitioning)

Let $G = (V, E)$ be a d -regular graph, $\mathbf{x} \in \mathbb{R}^V$ be a vector such that $\mathbf{x} \perp \mathbf{1}$, let M be the normalized adjacency matrix of G , define

$$\delta := \frac{\sum_{i,j} M_{i,j} |x_i - x_j|^2}{\frac{1}{n} \sum_{i,j} |x_i - x_j|^2}$$

and let S be the output of algorithm *SpectralPartitioning* on input G and \mathbf{x} . Then

$$h(S) \leq \sqrt{2\delta}$$

- If we apply the lemma to the case in which \mathbf{x} is an eigenvector of λ_2 , then $\delta = 1 - \lambda_2$, and so we have

$$h(G) \leq h(S) \leq \sqrt{2 \cdot (1 - \lambda_2)}$$

which is the difficult direction of Cheeger's inequalities.

- If we run the SpectralPartitioning algorithm with the eigenvector \mathbf{x} of the second eigenvalue λ_2 , we find a set S whose expansion is

$$h(S) \leq \sqrt{2 \cdot (1 - \lambda_2)} \leq 2\sqrt{h(G)}$$

Even though this doesn't give a constant-factor approximation to the edge expansion, it gives a very efficient, and non-trivial, approximation.

- Assume that $V = \{1, \dots, n\}$ and that $x_1 \leq x_2 \leq \dots \leq x_n$.
- The goal is to prove that there is an i such that $h(\{1, \dots, i\}) \leq \sqrt{2\delta}$
 - by showing that there is a distribution D over sets S of the form $\{1, \dots, i\}$ such that

$$\mathbb{E}_{S \sim D} \frac{1}{d} \text{Edges}(S, V - S) - \sqrt{2\delta} \min\{|S|, |V - S|\} \leq 0$$

- So there must exist a set S in the sample space such that

$$\frac{1}{d} \text{Edges}(S, V - S) - \sqrt{2\delta} \min\{|S|, |V - S|\} \leq 0$$

meaning that, for that set S , $h(S) \leq \sqrt{2\delta}$.

Thanks

These slides are based on a lecture by Dorothea Wagner and the lectures of Luca Trevisan available at <http://lucatrevisan.wordpress.com/category/teaching/cs359g/>