Graph Clustering

Schaeffer: Graph Clustering. Computer Science Review 1(1):27-64,2007.
Kannan, Vempala and Vetta: On clusterings: good, bad and spectral. JACM 51(3), 497-515, 2004.

Ville Lämsä

Researcher: Laboratory for Mechanics of Materials
Ph.D. student: CIS Lab.
Helsinki University of Technology

T-61.6040: Information Networks

Outline

Clustering problem

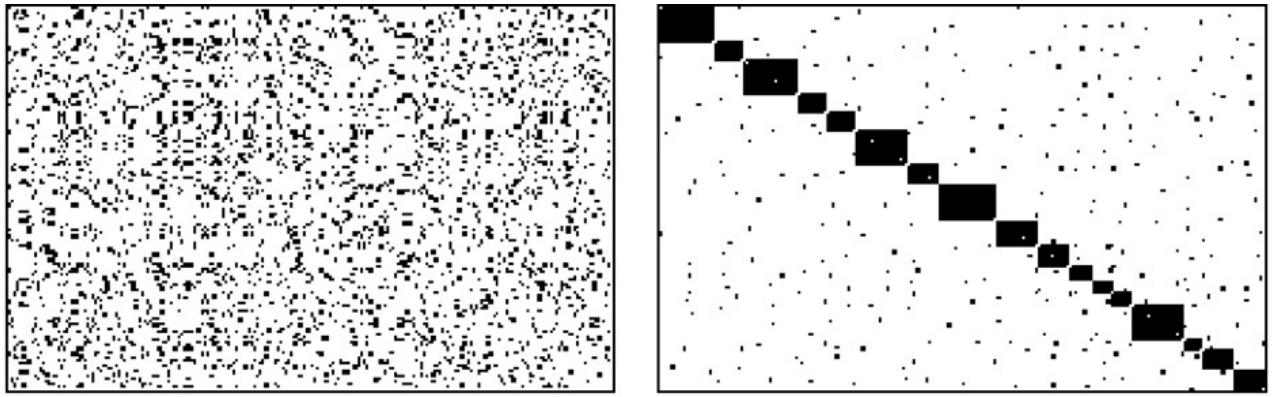

Figure: The random (left) and clustered adjacency matrix.

Generation models

Used as an artificial input data in clustering algorithms for evaluating and benchmarking.

e.g.

- Uniform random graph (the Gilbert model).
- The planted ℓ-partition model.
- The relaxed caveman graph.

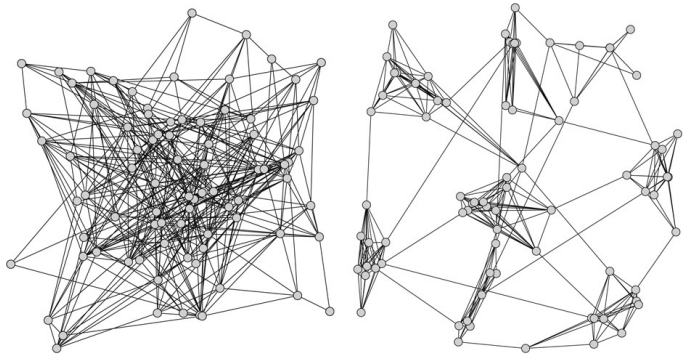

Figure: A random graph (left) and a relaxed caveman model.

Desirable cluster properties

No single definition.

- Vertices should be connected with paths, which are internal to the cluster.
- $deg_{ext}(v) = 0 \Rightarrow$ a good cluster.
- Homogeneity within clusters, heterogeneity between clusters.
- $\Rightarrow$ Compactness.
- $\Rightarrow$ Density.
- $\Rightarrow$ Integrity.
- Membership in several clusters - fuzzy clustering.

Similarity

Distance.

- Euclidean distance.
- Manhattan distance.
- tf-idf.
- Edit distance.
- etc.

Similarity.

- Cosine similarity.
- Jaccard index, Jaccard similarity coefficient.

Adjacency-based measures.

- Overlap of neighbourhoods.
- Pearson correlation.

Connectivity measures.

- Number of paths between each pair of vertices.

Fitness

Density measures.

- Decision problem: find the subset $S \subseteq V$ such that $|S| = k$ and the density $\delta(S) \geq \xi$.
- Additional problems: determine the k and the ξ.
- Often **NP**-hard or **NP**-complete.

Fitness

Cut-based measures - the independence of a subgraph.

- The cut size $c(e, V \backslash e)$ - the smaller, the better.
- Problem with cuts: important (large weighted) edges vs. unimportant (small weighted) edges.
- The relative cut size - the expansion:

> *"The expansion of a graph is the minimum ratio over all cuts of the graph of the total weight of edges of the cut to the number of vertices in the smaller part created by the cut."*

$\Rightarrow$ The larger the expansion of the clustering, the higher its quality.

- Weighted cuts - the conductance.

> *"Give greater importance to vertices that have many similar neighbours and lesser importance to vertices that have few similar neighbours."*

Fitness

Local fitness.

- The Cheeger ratio: the ratio of the cut size of the cluster to the minimum of the sums of degrees either inside the cluster or outside it.
- Combinations of local and relative density measures.
- The higher the internal degree of a vertex, the better it fits to the given cluster.

Some examples

Clustering with respect to a distance function.

- The minimum k-clustering problem: partition data set into k clusters such that the maximum intercluster distance is minimized. $\Rightarrow$ Approximation factor is **two**.
- The minimum k-centre problem: construct a set of centres such that the maximum distance from a vertex to the nearest centre is minimized. $\Rightarrow$ Approximation factor is **two**.
- The minimum k-median problem: the sum of the distances from a vertex to the nearest centre is minimized while keeping the order of the centre set fixed. $\Rightarrow$ Approximation factor is **near two**, unless (in given data set) **P=NP**.
- Perhaps the best known method is the k-means algorithm (in appendix). $\Rightarrow$ **NP**-hard.

Some examples

Easy to fool methods: e.g. minimum diameter, k-center, k-median, minimum sum.

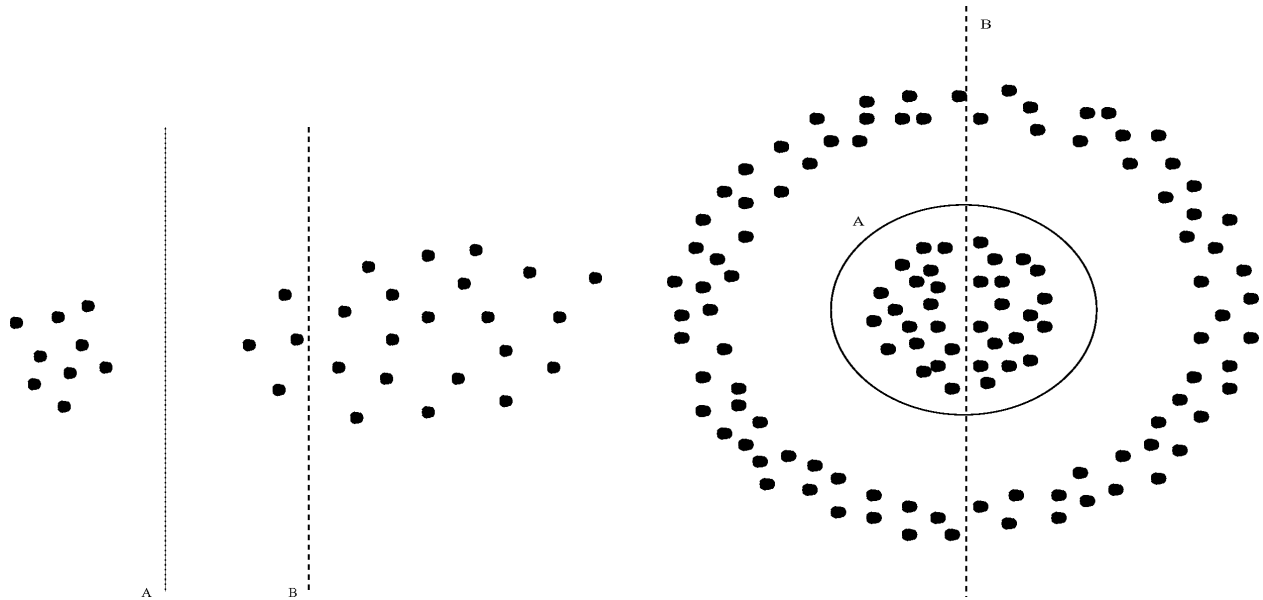

Figure: Diameter optimization (left) and 2-median optimization.

Iterative or online clustering

- Cluster all of the data elements at once vs. iteratively one element at the time.
  $\Rightarrow$ Obviously "all at once" -algorithms are not well for large data sets.
- Online clustering = clustering algorithm operates one datum at the time.
  $\Rightarrow$ Partial clustering.

Hierarchical clustering

- Top-level cluster is composed of subclusters, and so on.
- The root cluster contains at most all of the data, and each of the leaf clusters contains at least one data element.
  $\Rightarrow$ A dendrogram.

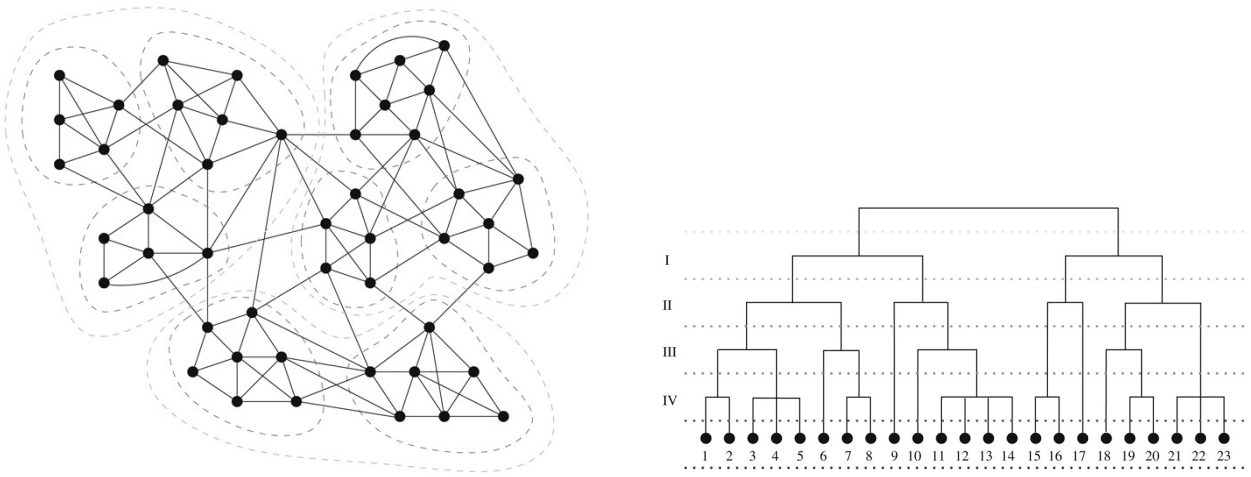

Figure: Examples of hierarchical structure and a dendrogram

Hierarchical clustering

- Top-down hierarchical algorithms: split the dataset iteratively or recursively into (two or more) smaller clusters, divisive clustering algorithms.
- Bottom-up hierarchical algorithms: start with each data element in its own singleton cluster or another set of small initial clusters, iteratively merging smaller clusters into larger ones, agglomerative clustering algorithms.
- The stopping condition.

$\Rightarrow$ **NP**-hard.

Cuts

Split graph into two by removing a cut.

- The relative order of subgraphs.
- When to stop splitting.
- Minimum bisection.
- ℓ-partition, ℓ-bisection.
- e.g. minimum cuts, low conductance, the Fiedler vector to find smallest normalized cut, SVD, etc.

Maximum flow

Connection between maximum-flow and minimum-cut problems:

> *"The maximum amount of flow is equal to the capacity of a minimal cut, i.e. between any two vertices, the quantity of flow from one to the other cannot be greater than the weakest set of edges somewhere between the two vertices."*

$\Rightarrow$ A minimum-cut tree.

- e.g. Ford-Fulkerson algorithm.

Spectral methods

Quite close relation to many other methods, as seen before (e.g. to cuts).

- Zhi-rong will present spectral clustering next time.

Betweenness

- Betweenness: the number of shortest paths connecting any pair of vertices that pass through the edge.
- Node-betweenness: for each vertex node-betweenness is the number of shortest paths in the graph that pass through that vertex.
- Quite often algoritms to compute betweenness operate in $\mathcal{O}(n \cdot m)$ time.
- The concept of modularity, many definitions: e.g. number of edges within groups minus expected number of edges within groups (Newman).
- e.g. Girvan-Newman algorithm (in appendix).

Voltage and potential

The graph as a circuit that has a unit resistor on each edge. (ref. the 3^{rd} lecture.)

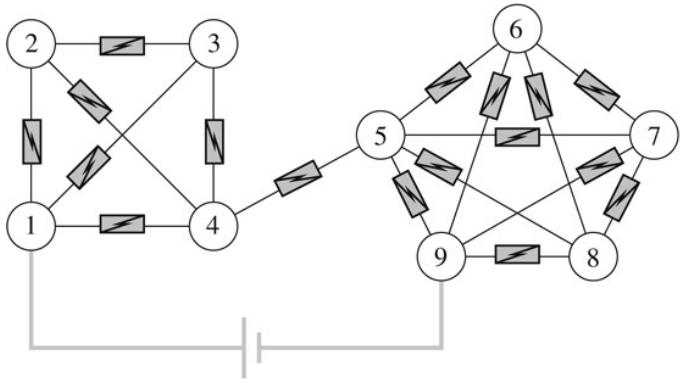

Figure: A graph as an electric circuit.

- Cluster the vertices based on the potential differences.
- Circuit analysis tools, e.g. SPICE (e.g. http://www.5spice.com/)

Voltage and potential

- Where to place battery when structure of a graph is not known a priori?
  $\Rightarrow$ Place it randomly and do the averaging.
  $\Rightarrow$ Pick a seed vertex and a "far away" sink vertex.
  $\Rightarrow$ The discrete Dirichlet problem $\Rightarrow$ The Fiedler vector.
- e.g. Wu–Huberman algorithm.

Markow chains and random walks

The components of the eigenvector corresponding to the second eigenvalue of the transition matrix of a random walk on a graph serve as "proximity" measures for how long it takes for the walk to reach each vertex. Two vertices in the same cluster should be "quickly reachable" from each other.

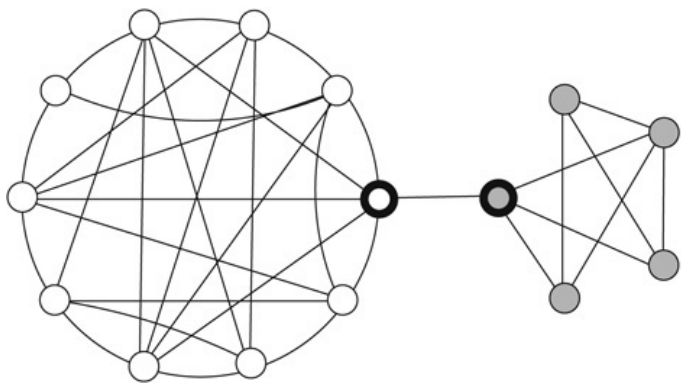

Figure: A two−cluster graph.

Other methods

One possible method as an example.

- Find the histogram of the density function.
- Filter the histogram by convolution.
- Use the output to split the graph in two.
- Continue iteratively.
- (http://www.tulip−software.org/)

Agglomerative global clustering

Iteratively merge clusters and continue until some threshold or a desired number of clusters is reached.

- The pairwise nearest neighbours method.
- Modularity optimization approach $\Rightarrow$ construct the full cluster hierarchy and then select a clustering from the resulting tree maximizing modularity.

Local clustering

Local clustering: clusters are computed one at a time based on only partial views of the graph topology.

- Symmetrical vs. asymmetrical clustering.
- Local availability: the adjacencies are known and there is direct access to the neighbouring vertices (in wider sense: also direct access to the second neighbours)
- Lower costs (e.g. time and storage requirements).

Local search

Find near-optimal solutions among large, complex sets of solution candidates.

- Heuristic or/and probabilistic algorithms.
- State space: the set of solution canditates (i.e. states).
- A neighbourhood relation: examine solution candidates one by one and then move to a neighbouring candidate.
- A fitness function: measures the quality of the solution of a state.
- A fitness function decides where the search will proceed - e.g. to the neighbour with the best fitness (greedy algorithms).
- e.g. hill-climbing, tabu search, simulated annealing.

Summary

- Many ways and methods to do clustering.
- Many application areas.

- Problems
  - Parameters.
  - Runtime and memory consumption.
  - Evaluation of clusterings.

The k-means algorithm I

1. Place k points into the space represented by the objects that are being clustered. These points represent initial group centroids.
2. Assign each object to the group that has the closest centroid.
3. When all objects have been assigned, recalculate the positions of the k centroids.
4. Repeat steps 2 and 3 until the centroids no longer move. This produces a separation of the objects into groups from which the metric to be minimized can be calculated.

Hierarchical clustering algorithm I

1. Start by assigning each object to a cluster each containing just one item. Let the distances (similarities) between the clusters be the same as the distances (similarities) between the objects they contain.
2. Find the closest (most similar) pair of clusters and merge them into a single cluster.
3. Compute distances (similarities) between the new cluster and each of the old clusters.
4. Repeat steps 2 and 3 until all objects are clustered into a single cluster.

Girvan-Newman algorithm I

1. The betweenness of all existing edges in the network is calculated first.
2. The edges with the highest betweenness are removed.
3. The betweenness of all edges affected by the removal is recalculated.
4. Steps 2 and 3 are repeated until no edges remain.