Clustering and community detection

Social and Technological Networks

Rik Sarkar

University of Edinburgh, 2016.

Clustering

- A core problem of machine learning:
  - Which items are in the same group?
- Identifies items that are similar relative to rest of data
- Simplifies information by grouping similar items
  - Helps in all types of other problems

Clustering

- Outline approach:
- Given a set of items
  - Define a distance between them
    - E.g. Euclidean distance between points in a plane; Euclidean distance between other attributes; path lengths in a network; tie strengths in a network…
  - Determine a grouping that optimises some function (prefers ‘close’ items in same group).

- Reference for clustering:
  - Charu Aggarwal: The Data Mining Textbook, Springer
  - Free on Springer site (from university network)

K-means clustering

- There are n items
- Select k ‘centers’
  - May be random k locations in space
  - May be location of k of the items selected randomly
  - May be chosen according to some method
- Iterate till convergence:
  - Assign each item to the cluster for its closest center
  - Recompute location of center as the mean location of all elements in the cluster
  - Repeat

K means: discussion

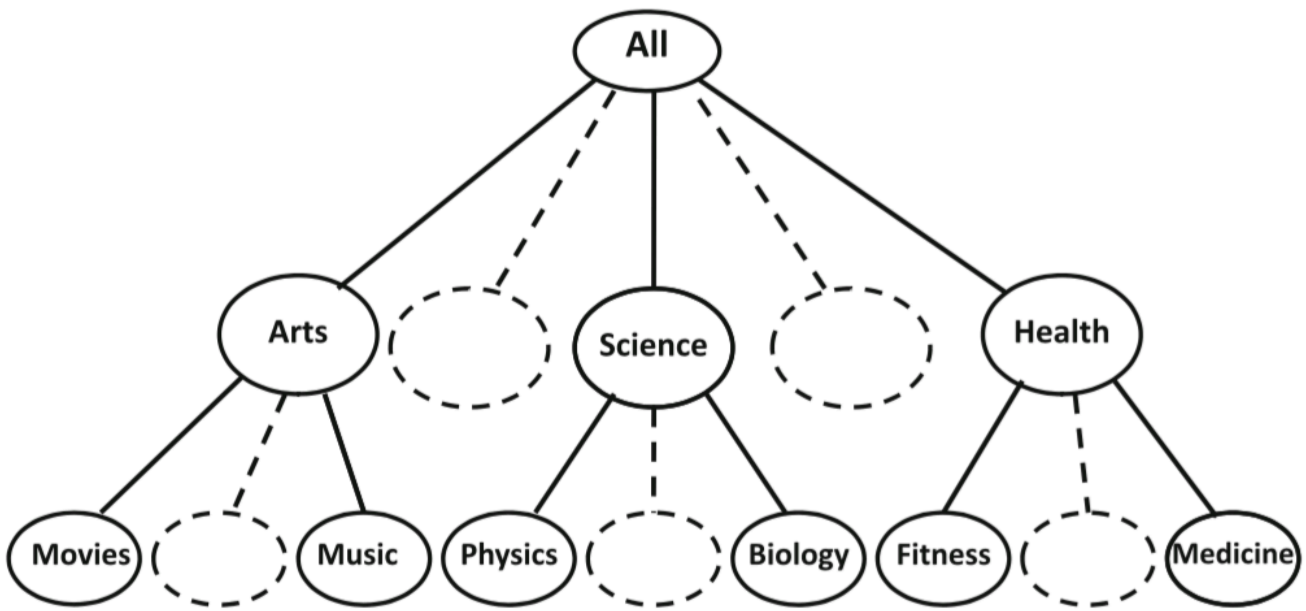
- Tries to minimise sum of distances of items to cluster centers
  - Computationally hard problem
  - Algorithm gives local optimum
- Depends on initialisation (starting set of centers)
  - Can give poor results
  - Slow speed
- The right 'k' may be unknown
  - Possible strategy: try different possibilities and take the best
- Can be improved by heuristics like choosing centers carefully
  - E.g. choosing centers to be as far apart as possible: choose one, choose point farthest to it, choose point farthest to both (maximise min distance to existing set etc)…
  - Try multiple times and take best result..

K-medoids

- Similar, but now each center must be one of the given items
  - In each cluster, find the item that is the best ‘center’ and repeat
- Useful when there is no ambient space
  - E.g. A distance between items can be computed, but they are not in any particular Euclidean space, so the ‘center’ is not a meaningful point

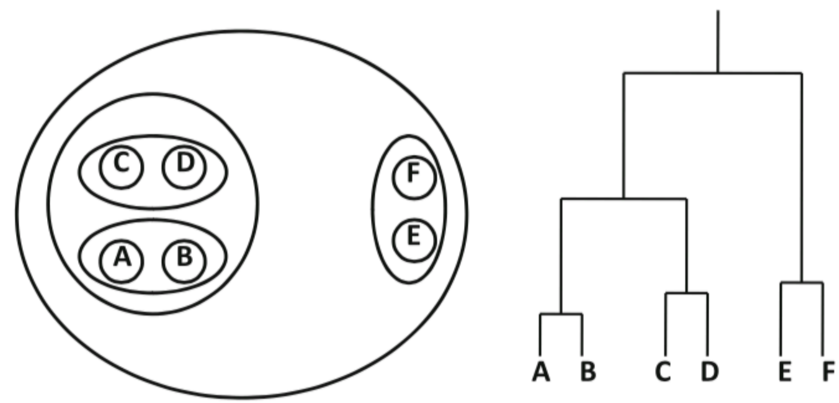
Hierarchical clustering

- Hierarchically group items



Hierarchical clustering

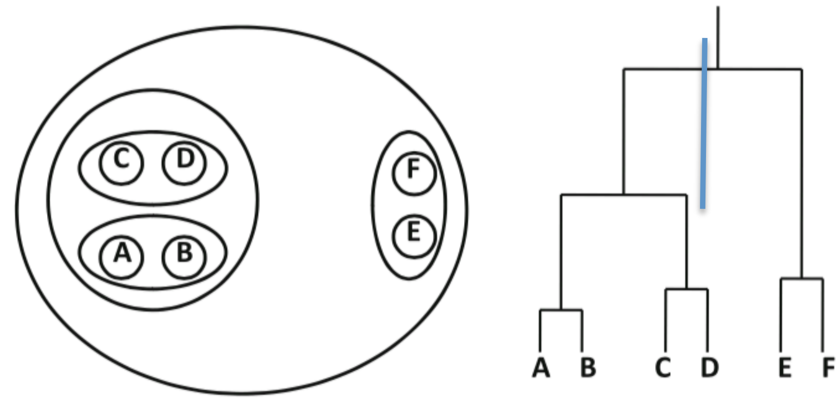
- Top down (divisive):
  - Start with everything in 1 cluster
  - Make the best division, and repeat in each subcluster
- Bottom up (agglomerative):
  - Start with n different clusters
  - Merge two at a time by finding pairs that give the best improvement



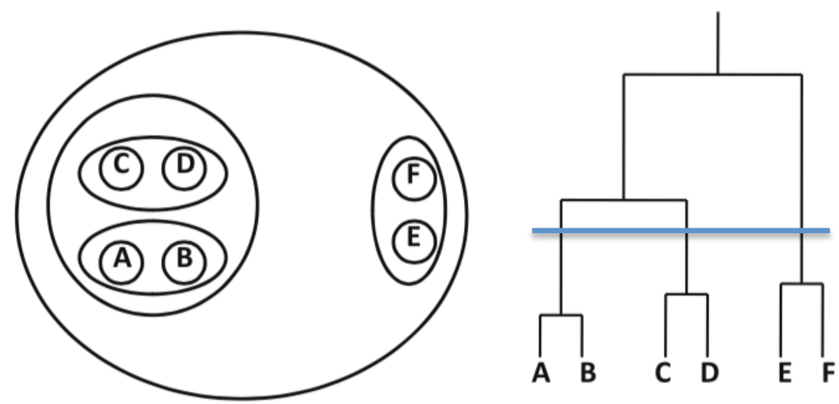
(a) Dendrogram

Hierarchical clustering

- Gives many options for a flat clustering
- Problem: what is the right place to ‘cut’ the dendogram?



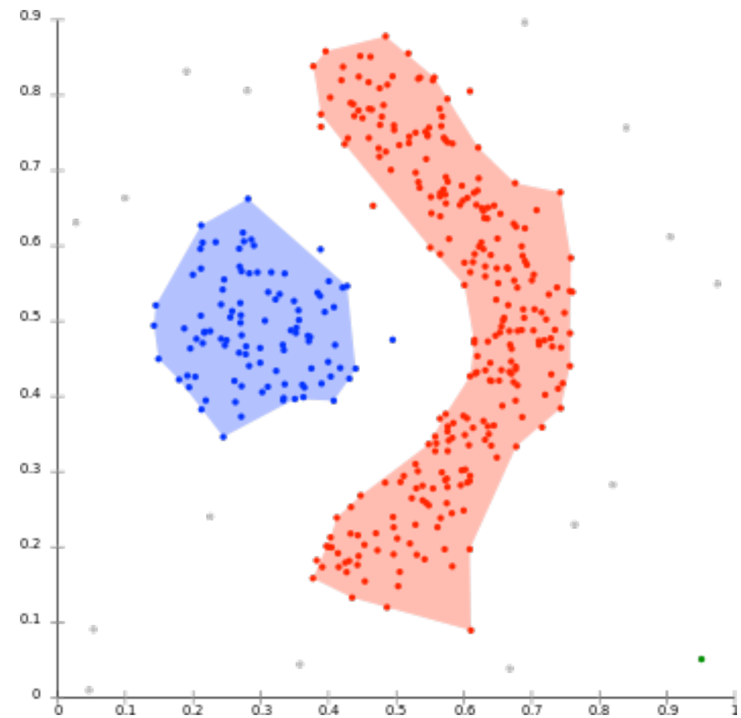
(a) Dendrogram



(a) Dendrogram

Density based clustering

- Group dense regions together
- Less dependent on distance configurations
- Better at non-linear separations
- Works with unknown number of clusters



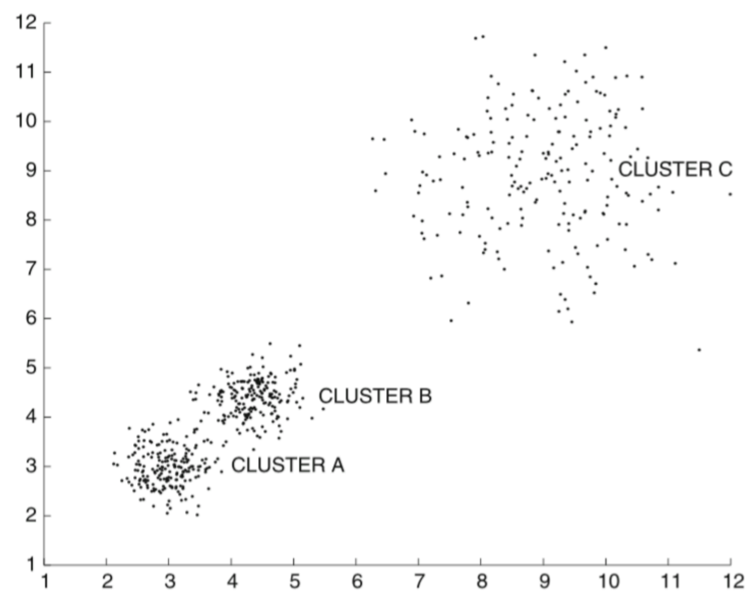
Density based clustering

- Density at a data point:
  - Number of data points within radius Eps
- A core point:
  - Point with density at least τ

Algorithm $DBSCAN$(Data: $\mathcal{D}$, Radius: Eps, Density: τ)
begin
  Determine core, border and noise points of $\mathcal{D}$ at level (Eps, τ);
  Create graph in which core points are connected
    if they are within Eps of one another;
  Determine connected components in graph;
  Assign each border point to connected component
    with which it is best connected;
  return points in each connected component as a cluster;
end

DBSCAN: Discussions

- Requires knowledge of suitable radius and density parameters (Eps and τ)
- Does not allow for possibility that different clusters may have different densities



Course and projects

- Office hours
  - Wednesdays as usual
  - This week, also:
    - Tuesday & Thursday 2pm – 3pm
- Report writing
  - Highlight whatever is important/interesting
    - Interesting result, interesting technique, anything unusual..
  - State it right at the beginning. Clear and concise.
  - Make it easy to find a reason to give you marks!

Communities

- Groups of friends
- Colleagues/collaborators
- Web pages on similar topics
- Biological reaction groups
- Similar customers/users …

Other applications

- A coarser representation of networks
- One or more meta-node for each community
- Identify bridges/weak-links
- Structural holes

Community detection in networks

- A simple strategy:
  - Choose a suitable distance measure based on available data
    - E.g. Path lengths; distance based on inverse tie strengths; size of largest enclosing group or common attribute; distance in a spectral (eigenvector) embedding; etc..
  - Apply a standard clustering algorithm

Clustering is not always suitable in networks

- Small world networks have small diameter
  - And sometime integer distances
  - A distance based method does not have a lot of option to represent similarities/dissimilarities
- High degree nodes are common
  - Connect different communities
  - Hard to separate communities
- Edge densities vary across the network
  - Same threshold does not work well everywhere

Definitions of communities

- Varies. Depending on application
- General idea: **Dense subgraphs:** More links within community, few links outside
- Some types and considerations:
  - Partitions: Each node in exactly one community
  - Overlapping: Each node can be in multiple communities

Finding dense subgraphs is hard in general

- Finding largest clique
  - NP-hard
  - Computationally intractable
  - Polynomial time (efficient) algorithms unlikely to exist
- Decision version: Does a clique of size k exist?
  - NP-complete
  - Computationally intractable
  - Polynomial time (efficient) algorithms unlikely to exist

Dense subgraphs: Few preliminary definitions

- For S, T subgraphs of V
- e(S,T): Set of edges from S to T
  - e(S) = e(S,S): Edges within S
- $d_S(v)$: number of edges from v to S
- Edge density of S : |e(S)|/|S|
  - Largest for complete graphs or cliques

Dense subgraph

- The subgraph with largest edge density
- There also exists a decision version:
  - Is there a subgraph with edge density > α
- Can be solved using Max Flow algorithms
  - $O(n^2m)$: inefficient in large datasets
  - Finds the one densest subgraph
- Variant: Find densest S containing given subset X
- Other versions: Find subgraphs size k or less
- NP-hard

Efficient approximation for finding dense S containing X

Let $G_n \leftarrow G$.
for $k = n$ downto $|X| + 1$ **do**
  Let $v \notin X$ be the lowest degree node in $G_k \setminus X$.
  Let $G_{k-1} \leftarrow G_k \setminus \{v\}$.
Output the densest subgraph among $G_n, \ldots, G_{|X|}$.

- Gives a 1/2 approximation
- Edge density of output S set is at least half of optimal set S*
- (Proof in Kempe 2011).

Modularity

- We want to find the many communities, not just one
- Clustering a graph
- Problem: What is the right clustering?
- Idea: Maximize a quantity called *modularity*

Modularity of subset S

- Given graph G
- Consider a random G' graph with same node degrees (remember configuration model)
  - Number of edges in S in G: $|e(S)|_G$
  - Expected number of edges in S in G': $E[|e(S)|_{G'}]$
  - Modularity of S: $|e(S)| - E[|e(S)|_{G'}]$
  - More coherent communities have more edges inside than would be expected in a random graph with same degrees
  - Note: modularity can be negative

Modularity of a clustering

- Take a partition (clustering) of V: $\mathcal{P} = \{S_1, \ldots, S_k\}$
- Write $d(S_i)$ for sum of degrees of all nodes in S_i
- Can be shown that $E[|e(S)|_{G'}] \sim d(S_i)^2$
- Definition: Sum over the partition:

$$q(\mathcal{P}) \quad = \quad \frac{1}{m} \sum_i |e(S_i)| - \frac{1}{4m} d(S_i)^2$$

Modularity based clustering

- Modularity is meant for use more as a measure of quality, not so much as a clustering method
- Finding clustering with highest modularity is NP-hard
- Heuristic:
  - Use modularity matrix
  - Take its first eigen vector
- Note: Modularity is a relative measure for comparing community structure.
- Not entirely clear in which cases it may or may not give good results
- A threshold of 0.3 or more is sometimes considered to give good clustering

- Can be used as a stopping criterion (or finding right level of partitioning) in other methods
  - Eg. Girvan-newman

Karate club hierarchic clustering

- Shape of nodes gives actual split in the club due to internal conflicts
  - Newman 2003