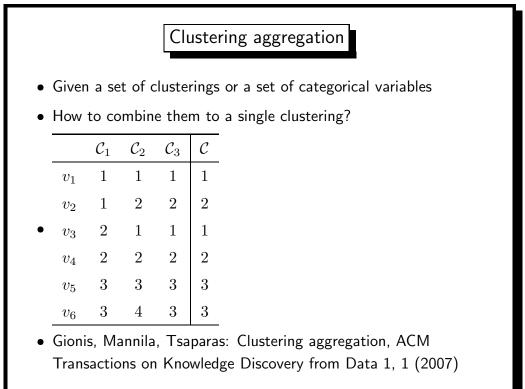
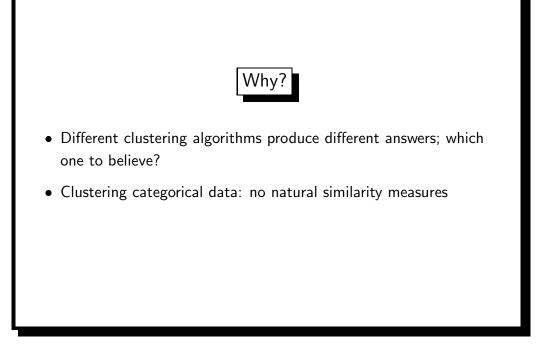
# Clustering aggregation

Algorithmic Methods of Data Mining, Fall 2005, Clustering aggregation





## Definitions

- $n \text{ objects } V = \{v_1, \ldots, v_n\}.$
- A clustering C of V is a *partition* of V into k disjoint sets  $C_1, \ldots, C_k$
- The k sets  $C_1, \ldots, C_k$  are the clusters of  $\mathcal{C}$ .
- C(v) the label of the cluster to which the object v belongs, i.e., C(v) = j if and only if  $v \in C_j$
- m clusterings:  $C_i$  to denote the *i*th clustering
- $k_i$  for the number of clusters of  $\mathcal{C}_i$

How to compare two clusterings? • u and v in V $d_{u,v}(\mathcal{C}_1, \mathcal{C}_2) = \begin{cases} 1 & \text{if } \mathcal{C}_1(u) = \mathcal{C}_1(v) \text{ and } \mathcal{C}_2(u) \neq \mathcal{C}_2(v), \\ & \text{or } \mathcal{C}_1(u) \neq \mathcal{C}_1(v) \text{ and } \mathcal{C}_2(u) = \mathcal{C}_2(v), \\ & 0 & \text{otherwise.} \end{cases}$   $d_V(\mathcal{C}_1, \mathcal{C}_2) = \sum_{(u,v) \in V \times V} d_{u,v}(\mathcal{C}_1, \mathcal{C}_2).$ 

Algorithmic Methods of Data Mining, Fall 2005, Clustering aggregation

5

Problem definition

**Problem 1 (Clustering aggregation)** Given a set of objects V and m clusterings  $C_1, \ldots, C_m$  on V, compute a new clustering C that minimizes the total number of disagreements with all the given clusterings, i.e., it minimizes

$$D(\mathcal{C}) = \sum_{i=1}^{m} d_V(\mathcal{C}_i, \mathcal{C}).$$

Example

Equivalent to finding the "center" of the clusterings  $\mathcal{C}_i$  with respect to the measure  $d_V$ 

The distance is a metric

**Observation** 1

 $d_V(\mathcal{C}_1, \mathcal{C}_3) \le d_V(\mathcal{C}_1, \mathcal{C}_2) + d_V(\mathcal{C}_2, \mathcal{C}_3)$ 

Why?

Show that for each pair (u, v) we have  $d_{u,v}(\mathcal{C}_1, \mathcal{C}_3) \leq d_{u,v}(\mathcal{C}_1, \mathcal{C}_2) + d_{u,v}(\mathcal{C}_2, \mathcal{C}_3).$ 

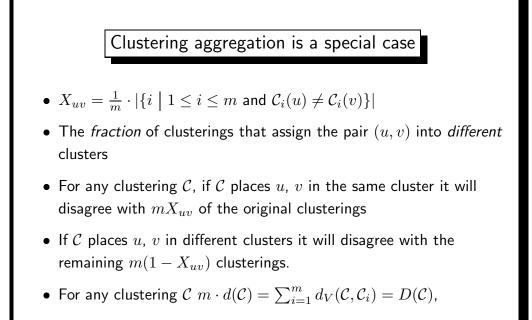
Algorithmic Methods of Data Mining, Fall 2005, Clustering aggregation

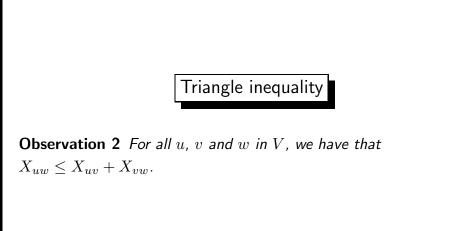
Correlation clustering

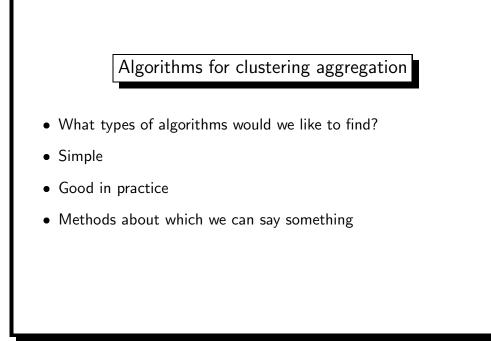
- A slightly more general problem
- Problem 2 (Correlation clustering) Given a set of objects V, and distances X<sub>uv</sub> ∈ [0, 1] for all pairs u, v ∈ V, find a partition C for the objects in V that minimizes the score function

$$d(\mathcal{C}) = \sum_{\substack{(u,v)\\\mathcal{C}(u) = \mathcal{C}(v)}} X_{uv} + \sum_{\substack{(u,v)\\\mathcal{C}(u) \neq \mathcal{C}(v)}} (1 - X_{uv}).$$
(1)

 $\overline{7}$ 



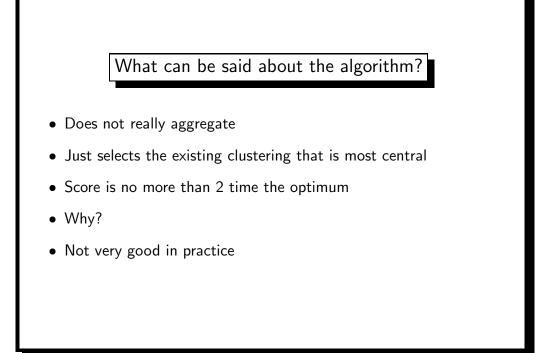




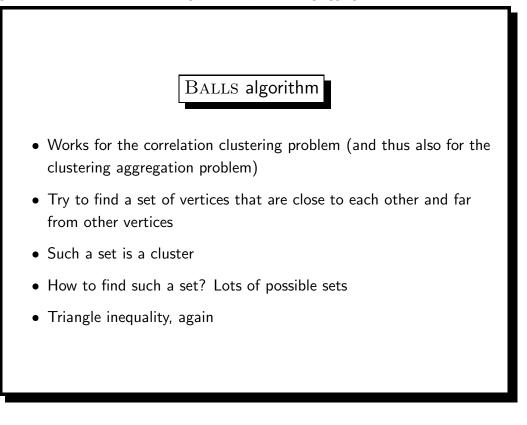
11

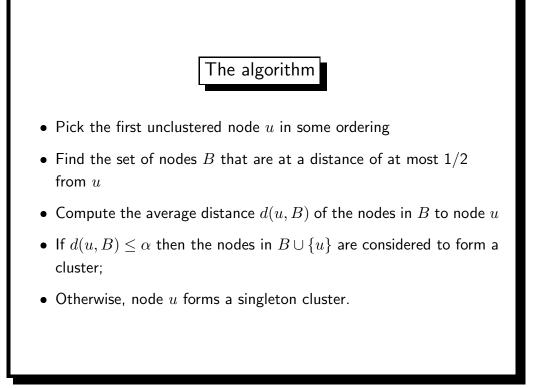
## $BestClustering \ algorithm$

- Given m clusterings  $\mathcal{C}_1, \ldots, \mathcal{C}_m$
- BESTCLUSTERING finds the input clustering  $C_i$  that minimizes the total number of disagreements  $D(C_i)$
- Can be implemented to work in time  $O(m^2n)$









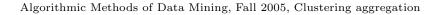
Algorithmic Methods of Data Mining, Fall 2005, Clustering aggregation

Properties of the algorithm

- If α = <sup>1</sup>/<sub>4</sub>, the cost of solution produced by the BALLS algorithm is guaranteed to be at most 3 times the cost of the optimal clustering.
- Not very good in practice
- $\alpha = 2/5$  seems better
- Complexity:  ${\cal O}(mn^2)$  for computing the distances  $X_{uv},\,{\cal O}(n^2)$  for running the algorithm

#### The AGGLOMERATIVE algorithm

- Correlation clustering problem
- Agglomerative clustering algorithm
- Start with all nodes in singleton clusters
- Merge the two clusters with the smallest cost
- Cost: the average weight of edges between clusters
- If this is less than 1/2, merge; otherwise, stop



The FURTHEST algorithm

- Correlation clustering
- In the beginning all nodes are in a single cluster
- Find the pair of clusters that are furthest apart
- Make them new cluster centers
- Reassign points to the closest cluster center
- Find the node that is furthest away from existing centers; repeat
- If the cost is lower that in the previous step, continue, otherwise, the result from the previous step is the answer

# Properties of the algorithm

- $O(mn^2)$  for creating the weights
- $O(k^2n)$  for running the algorithm; k is the number of clusters created

Algorithmic Methods of Data Mining, Fall 2005, Clustering aggregation

The LOCALSEARCH algorithm

- A heuristic that can be applied on top of any correlation clustering method
- Start with some clustering
- For each node: find if the cost would improve if the node were moved to another cluster or made into a singleton cluster
- Iterate

### Efficient implementation

• Cost  $d(v, C_i)$  of assigning a node v to a cluster  $C_i$ 

$$d(v, C_i) = \sum_{u \in C_i} X_{vu} + \sum_{u \in \overline{C_i}} (1 - X_{vu}).$$

- The first term is the cost of merging v in  $C_i$
- The second term is the cost of not merging node v with the nodes not in  $C_i$

Algorithmic Methods of Data Mining, Fall 2005, Clustering aggregation

- More efficient formulation: for every  $C_i$  we compute and store  $M(v, C_i) = \sum_{u \in C_i} X_{vu}$  and the size of the cluster  $|C_i|$ .
- ۰

$$d(v, C_i) = M(v, C_i) + \sum_{j \neq i} (|C_j| - M(v, C_j))$$

- The cost of assigning node v to a singleton cluster is  $\sum_{j} (|C_j| M(v, C_j)).$
- Given the distance matrix  $X_{uv}$  (takes time  $O(mn^2)$ )
- The running time of the LOCALSEARCH algorithm is  $O(Tn^2)$
- T is the number of local search iterations

${f Algorithm}$	k	<i>I</i> (%)	$E_D$
Class labels	2	0	34,184
Lower bound			28,805
BestClustering	3	15.1	31,211
Agglomerative	2	14.7	30,408
Furthest	2	13.3	30,259
$BALLS_{\alpha=0.4}$	2	13.3	30,181
LocalSearch	2	11.9	29,967
$ROCK_{k=2,\theta=0.73}$	2	11	32,486
$LIMBO_{k=2,\phi=0.0}$	2	11	30,147

Results on Votes dataset. k is the number of clusters, I is the impurity index, and  ${\it E}_{\it D}$  is the disagreement error. The lower bound on  ${\it E}_{\it D}$  is computed by considering an algorithm that merges all edges with weight less than  $\frac{1}{2}$ , and splits all edges with weight greater than  $\frac{1}{2}$ .

Algorithmic Methods of Data Mining, Fall 2005, Clustering aggregation

