Università di Pisa

Data Mining and Machine Learning
Bioinspired computational methods
Biological data mining

Clustering with Constraints

Francesco Marcelloni

Department of Information Engineering
University of Pisa
ITALY

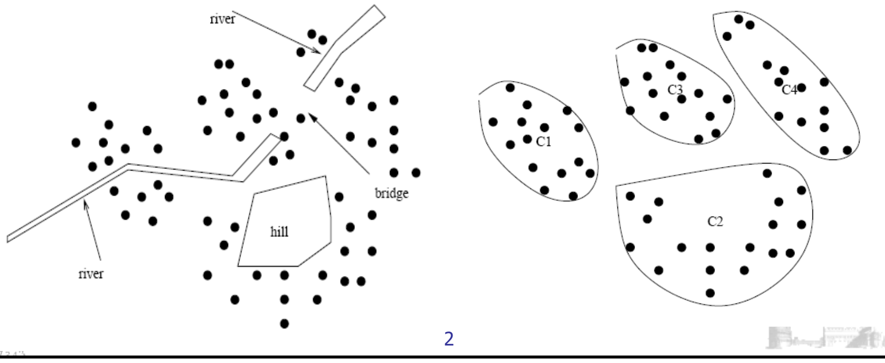
Some slides belong to the collection

Jiawei Han, Micheline Kamber, and Jian Pei
University of Illinois at Urbana-Champaign
Simon Fraser University

1

Why Constraint-Based Cluster Analysis?

- Need user feedback: Users know their applications the best
- Less parameters but more user-desired constraints:
  - A bank manager wishes to locate four ATMs in the area in the figure on the left: obstacle and desired clusters. Ignoring the obstacles will result in the clusters on the right



2

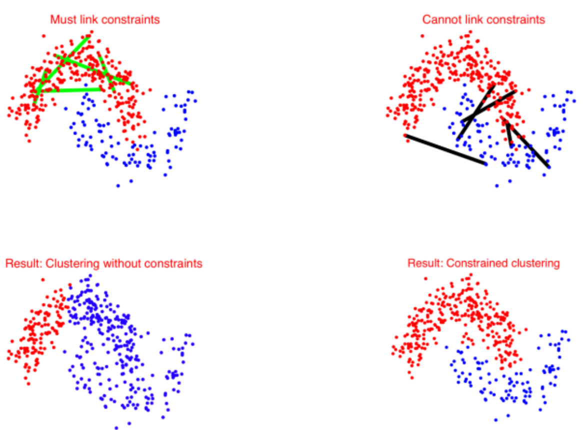
Categorization of Constraints

- Constraints on instances: specifies how a pair or a set of instances should be grouped in the cluster analysis
  - Must-link vs. cannot link constraints
    - must-link(x, y): x and y should be grouped into one cluster
  - Constraints can be defined using variables, e.g.,
    - cannot-link(x, y) if distance(x, y) > d

3

Categorization of Constraints

- Constraints on instances



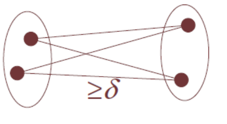
4

Categorization of Constraints

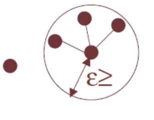
- Constraints on clusters: specify a requirement on the clusters
  - E.g., specify the min number of objects in a cluster, the max diameter of a cluster, the shape of a cluster (e.g., a convex), number of clusters (e.g., k)
  - δ-constraint (Minimum separation)
    - For any two clusters S_i, S_j, $\forall i, j$
    - For each two instances $s_p \in Si, s_q \in Sj, \forall p, q$
    - $D(s_p,s_q) >= \delta$
  - ε-constraint
    - For any cluster S_i, $|S_j| > 1$
    - $\forall p, s_p \in Si, \exists s_q \in Si$: $\varepsilon \geq D(sp,sq)$, $s_p <> s_q$

Categorization of Constraints

- Constraints on clusters can be converted to instance level constraints
  - δ-constraint (Minimum separation)
    - For every point x, must-link all points y such tha D(x,y) < δ, i.e., conjunction of must link (ML) constraints



  - ε-constraint
    - For every point x, must link to at least one point y such that D(x,y) <= ε, i.e. disjunction of ML constraints



  - Will generate many instance level constraints

Categorization of Constraints

- Constraints on similarity measurements: specifies a requirement that the similarity calculation must respect
  - E.g., to cluster people as moving objects in a plaza, while Euclidean distance is used to give the walking distance between two points, a constraint on similarity measurement is that the trajectory implementing the shortest distance cannot cross a wall.

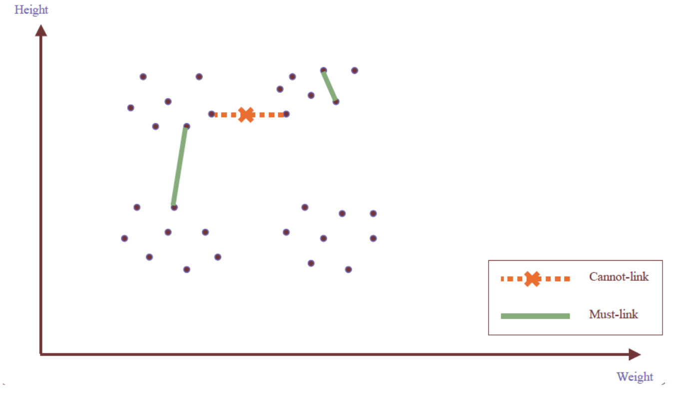
Categorization of Constraints

- Hard vs. soft constraints;
  - A constraint is hard if a clustering that violates the constraint is unacceptable.
  - A constraint is soft if a clustering that violates the constraint is not preferable but acceptable when no better solution can be found. Soft constraints are also called preferences.

Clustering with Constraints

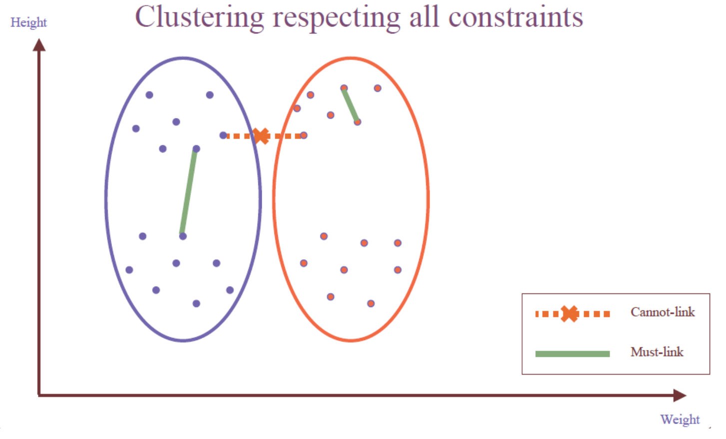
- Clustering with constraints:
  - Partition unlabeled data into clusters and use constraints to aid and bias clustering
- Goal
  - Examples in same cluster similar, separate clusters different and constraints are maximally respected
- Enforcing Constraints:
  - **Strict enforcement**: find best feasible clustering respecting all constraints
  - **Partial enforcement**: find best clustering maximally respecting constraints

Example: Enforcing Constraints



Example: Enforcing Constraints

Clustering respecting all constraints



Categorization of Constraints

- Conflicting or redundant constraints
  must-link(x, y) if dist(x, y) < 5
  cannot-link(x, y) if dist(x, y) > 3.
- If a data set has two objects, x, y, such that dist(x, y) = 4, then no clustering can satisfy both constraints simultaneously.
- How can we measure the quality and the usefulness of a set of constraints?
  - Informativeness: the amount of information carried by the constraints that is beyond the clustering model. Given a data set, D, a clustering method, A, and a set of constraints, C, the informativeness of C with respect to A on D can be measured by the fraction of constraints in C that are unsatisfied by the clustering computed by A on D.
  - Coherence of a set of constraints: the degree of agreement among the constraints themselves, which can be measured by the redundancy among the constraints

The Effects of Constraints on Clustering Solutions

- Constraints divide the set of all plausible solutions into two sets: feasible and infeasible: $S = S_F \cup S_I$
- Constraints effectively reduce the search space to S_F
- S_F all have a common property
- So it is not unexpected that we find solutions with a desired property and find them quickly.

Constraint-Based Clustering Methods (I): Handling Hard Constraints

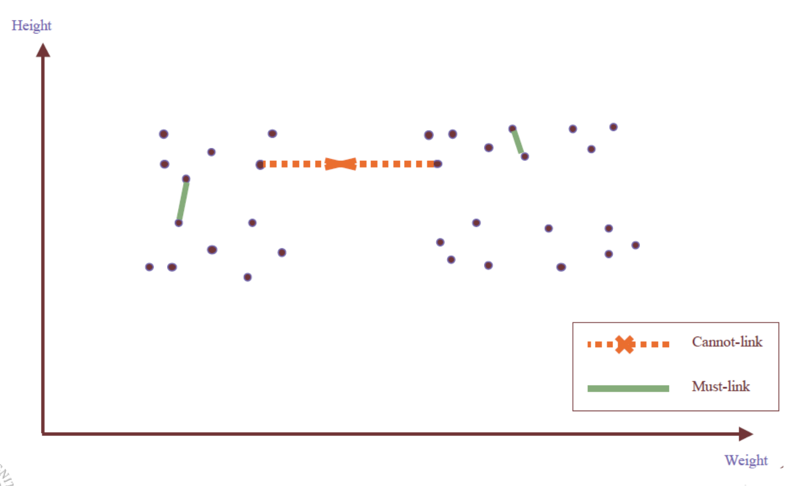
- Handling hard constraints: Strictly respect the constraints in cluster assignments
- The COP-k-means algorithm
  - Generate super-instances for must-link constraints
    - Compute the transitive closure of the must-link constraints
    - To represent such a subset, replace all those objects in the subset by the mean.
    - The super-instance also carries a weight, which is the number of objects it represents
  - Conduct modified k-means clustering to respect cannot-link constraints
    - Modify the center-assignment process in k-means to a nearest feasible center assignment
    - An object is assigned to the nearest center so that the assignment respects all cannot-link constraints

Constraint-Based Clustering Methods (I): Handling Hard Constraints

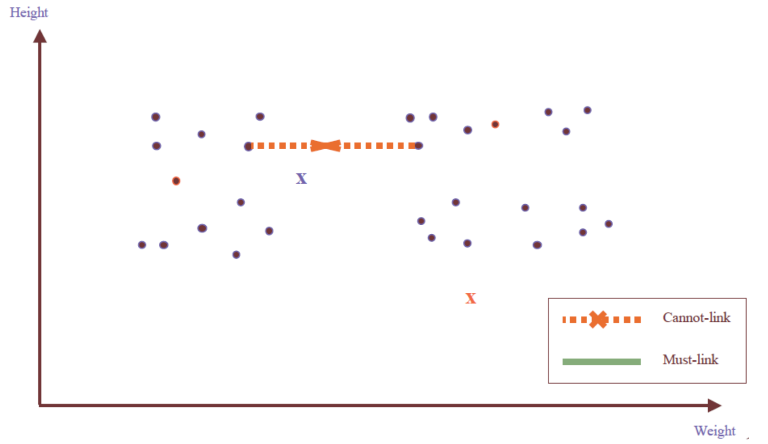
Input: S_u: unlabeled data, S_l: labeled data, k: the number of clusters to find, q: number of constraints to generate.
Output: A set partition of $S = S_u \cup S_l$ into k clusters so that all the constraints in $C = ML \cup CL$ are satisfied.

1. $ML = \emptyset, CL = \emptyset$
2. **loop** q times **do**
   (a) Randomly choose two distinct points x and y from S_l.
   (b) if(Label(x) = Label(y)) $ML = ML \cup \{x, y\}$ else $CL = CL \cup \{x, y\}$
3. Compute the transitive closure from ML to obtain the connected components $CC_1, ..., CC_r$.
4. For each i, $1 \leq i \leq r$, replace data points in CC_i with the average of the points in CC_i.
5. Randomly generate cluster centroids $C_1, \ldots, C_k$.
6. **loop** until convergence **do**
   (a) **for** i = 1 **to** $|S|$ **do**
      (a.1) Assign s_i to closest feasible cluster.
   (b) Recalculate $C_1, \ldots, C_k$.

Example: COP-K Means

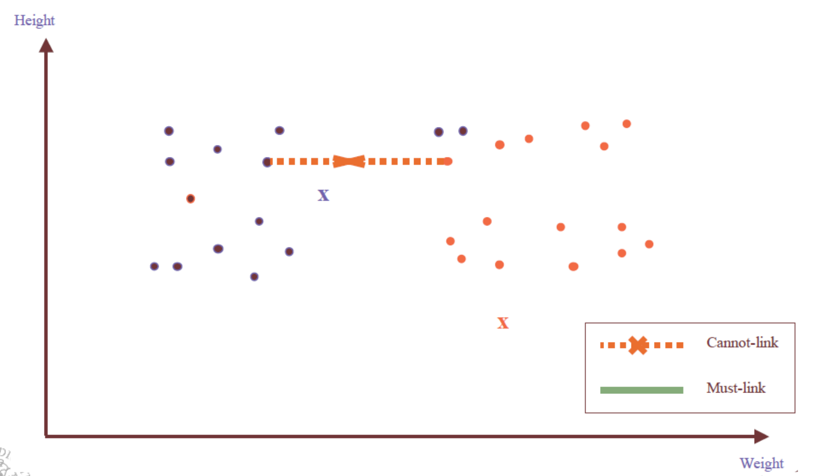


Example: COP-K Means – ML Points Averaged



17

Example: COP-K Means – 3 Nearest Feasible Assignment



18

Constraint-Based Clustering Methods (II): Handling Soft Constraints

- Treated as an optimization problem: When a clustering violates a soft constraint, a penalty is imposed on the clustering
- Overall objective: Optimizing the clustering quality, and minimizing the constraint violation penalty
- Ex. CVQE (Constrained Vector Quantization Error) algorithm: Conduct k-means clustering while enforcing constraint violation penalties

Constraint-Based Clustering Methods (III): Handling Soft Constraints

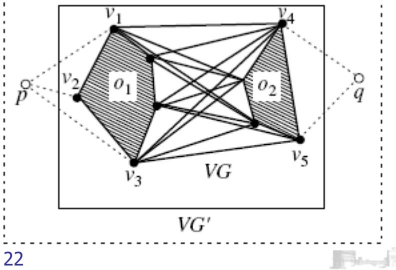
- Objective function: Sum of distance used in k-means, adjusted by the constraint violation penalties
  - Penalty of a must-link violation
    - If objects x and y must-be-linked but they are assigned to two different centers, c_1 and c_2, dist(c_1, c_2) is added to the objective function as the penalty
  - Penalty of a cannot-link violation
    - If objects x and y cannot-be-linked but they are assigned to a common center c, dist(c, c'), between c and c' is added to the objective function as the penalty, where c' is the closest cluster to c that can accommodate x or y

Speeding Up Constrained Clustering

- It is costly to compute some constrained clustering
- Ex. Clustering with obstacle objects: Tung, Hou, and Han. Spatial clustering in the presence of obstacles, ICDE'01
  - Cluster people as moving objects in a plaza.
  - Euclidean distance is used to measure the walking distance. However, constraint on similarity measurement is that the trajectory implementing the shortest distance cannot cross a wall.
  - Distance has to be derived by geometric computations: the computational cost is high if a large number of objects and obstacles are involved.
- A point p is visible from another point q if the straight line joining p and q does not intersect any obstacles.

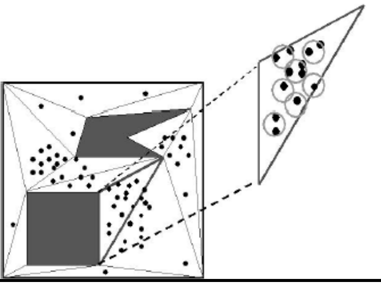
Speeding Up Constrained Clustering

- A visibility graph is the graph, VG = (V,E), such that each vertex of the obstacles has a corresponding node in V and two nodes, v_1 and v_2, in V are joined by an edge in E if and only if the corresponding vertices they represent are visible to each other.
- Let VG' = (V',E') be a visibility graph created from VG by adding two additional points, p and q, in V'. E' contains an edge joining two points in V' if the two points are mutually visible.
- The shortest path between two points, p and q, will be a subpath of VG'.



Speeding Up Constrained Clustering

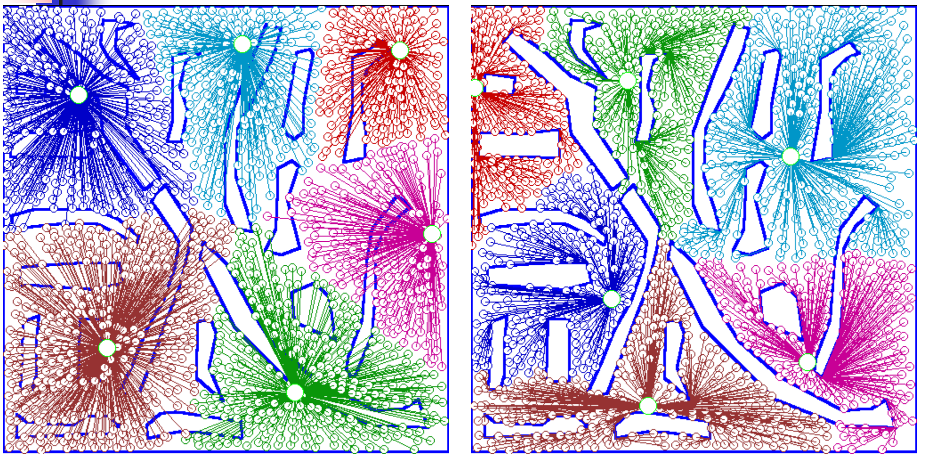
- To reduce the cost of distance computation between any two pairs of objects or points, several pre-processing and optimization techniques can be used.
- One method groups points that are close together into microclusters.
- This can be done by first triangulating the region R into triangles, and then grouping nearby points in the same triangle into microclusters, using a method similar to BIRCH or DBSCAN.



Speeding Up Constrained Clustering

- By processing microclusters rather than individual points, the overall computation is reduced.
- After that, precomputation can be performed to build two kinds of join indices based on the computation of the shortest paths:
  - (1) VV indices, for any pair of obstacle vertices, and
  - (2) MV indices, for any pair of microcluster and obstacle vertex.
- Use of the indices helps further optimize the overall performance.
- Using such precomputation and optimization strategies, the distance between any two points (at the granularity level of a microcluster) can be computed efficiently.
- Thus, the clustering process can be performed in a manner similar to a typical efficient k-medoids algorithm, such as CLARANS, and achieve good clustering quality for large data sets.

An Example: Clustering With Obstacle Objects



Not Taking obstacles into account   Taking obstacles into account