Declarative Approaches for Constrained Clustering

Thi-Bich-Hanh DAO

LIFO - Université d'Orléans

GDR-IA GT Caviar
Montpellier November 30, 2021

Outline

Clustering

- Given n objects $\{o_1, \ldots, o_n\}$, find a partition of the objects into k groups (clusters) s.t.:
  - objects in a group are similar and/or
  - objects of different groups are dissimilar

- Different settings:
  - Conceptual clustering: with objects described by boolean features, find clusters and their descriptions (concepts)
  - Distance-based clustering: based on a dissimilarity measure between pairs of points
  - Spectral clustering, correlation clustering: based on a similarity between pairs of points defined by an weighted graph

Conceptual clustering

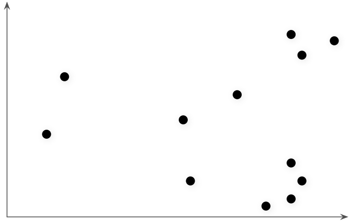
- n objects (transitions) $\mathcal{T}$ described by
- m boolean features (items) $\mathcal{I}$

	a	b	c
o_1	1	1	0
o_2	1	1	1
o_3	0	1	1
o_4	0	1	1
o_5	0	1	1

- Pattern: a set of items $I \subseteq \mathcal{I}$, closed if all the objects satisfying I have only I in common.
- Concept : (T, I), with $T \subseteq \mathcal{T}$, $I \subseteq \mathcal{I}$ closed pattern, such that the objects in T, and only them, satisfy I

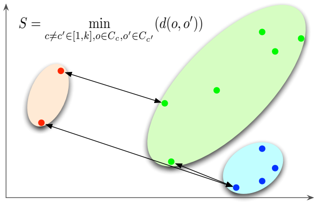
  $(\{o_2\}, \{a, b, c\})$, $(\{o_1, o_2\}, \{a, b\})$, $(\{o_2, o_3, o_4, o_5\}, \{b, c\})$
- Conceptual clustering: finding k non overlapping clusters covering all data and corresponding to concepts

Dissimilarity-based clustering

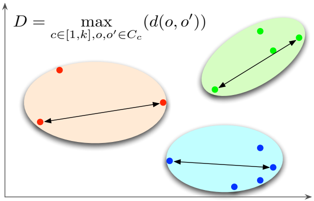


- Given $\mathcal{O} = \{x_i \in \mathbb{R}^m\}_1^n$, a dissimilarity measure $d : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}^+$
- Find a partition of $\mathcal{O}$ into K homogeneous clusters
- The homogeneity usually characterized by an optimization criterion

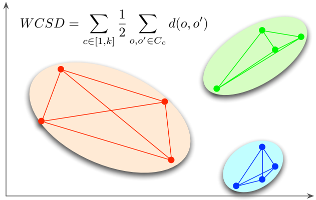
Clustering optimization criteria



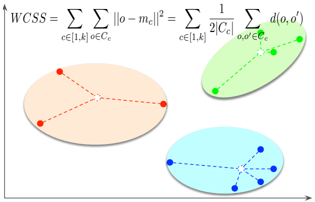
Maximizing S: minimal split between clusters



Minimizing D: maximal cluster diameter

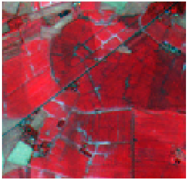


Minimizing WCSD: within-cluster sum of dissimilarities



Minimizing WCSS: within-cluster sum of squares

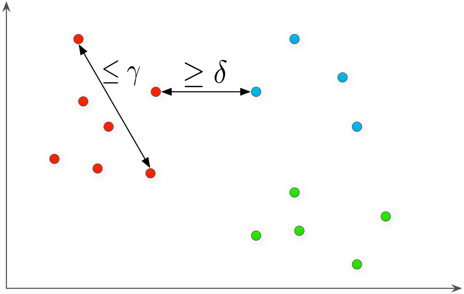
Constrained clustering



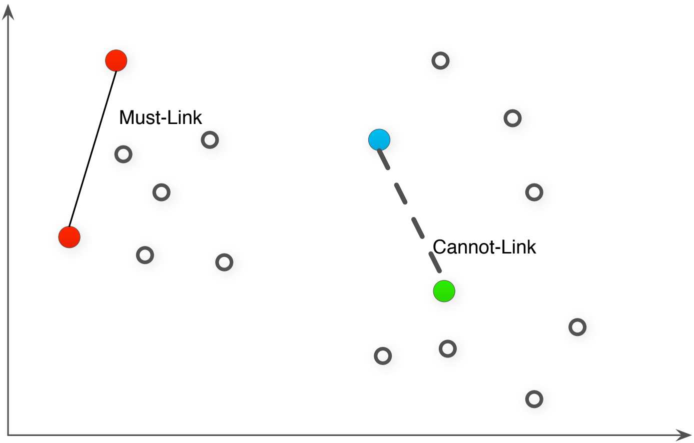
- Clustering is in general NP-hard
- Classic methods are usually heuristic and search for a local optimum, e.g. k-means for WCSS
  $\Longrightarrow$ Different local optima may exist
- The clustering solution must be coherent with the prior knowledge
  $\Longrightarrow$ Knowledge integrated into the clustering process by means of user-constraints
- Constrained clustering: clustering under
  - constraints on clusters
  - constraints on pairs of points
- With user-constraints, polynomial criterion (split) becomes NP-Hard

Constraints on clusters

- Capacity constraint: cluster size
  - lower-bounded by α
  - upper-bounded by β
- Maximal diameter constraint: cluster diameter upper-bounded by γ
- Minimal margin constraint: separation between clusters lower-bounded by δ
- Density constraint
- etc.



Constraints on pairs of points



Classic approaches for constrained clustering

- Classic clustering methods designed for one optimization criterion
  - K-means: $\arg\min_C \sum_{c\in[1,k]} \sum_{o_i\in C_c} d(o_i,\mu_c)^2$
  - FPF (K-centers): $\arg\min_C \max_{c\in[1,k],o,o'\in C_c} d(o,o')$

- Their extension integrates a certain type of user-constraints
  - ML/CL constraints:
    - COP-Kmeans [Wagstaff *et al.* 2001],
    - PCK-means [Basu *et al.* 2004], MPCK-means [Bilenko *et al.* 2004],
    - ...
  - cluster size constraint [Ng 2000, Bradley *et al.* 2000, Ge *et al.* 2007, Demiriz *et al.* 2008, ...]

Declarative approaches for constrained clustering

- Formulation of constrained clustering as a problem in
  - SAT
  - Constraint Programming (CP)
  - Integer Linear Programming (ILP)
- Use of SAT/CP/ILP solvers

Various works

- Conceptual clustering:
  - CP: de Raedt *et al.* 2008, Khiari *et al.* 2010, Guns *et al.* 2011, Chabert *et al.* 2017
  - SAT: Métivier *et al.* 2012
  - ILP: Mueller *et al.* 2010, Ouali *et al.* 2016

- Correlation clustering:
  - MIP and MAXSAT: Berg *et al.*, 2013, 2017

- Dissimilarity based clustering:
  - SAT: Davidson *et al.*, 2010
  - CP: Dao *et al.*, 2013, 2017, Guns *et al.*, 2016
  - ILP: Babaki *et al.*, 2014

Outline

Constrained clustering as a 2-SAT problem

Davidson & al., 2010

Find a partition of n points into 2 clusters 0/1.

- Partition represented by n boolean variables x_i:
  - $x_i = 0(1)$: point i belongs to cluster 0 (resp. 1)
- Constraints formulated into 2-SAT
  - $Must\_Link(x_i, x_j)$

$$(x_i \wedge x_j) \vee (\overline{x_i} \wedge \overline{x_j}) \Longleftrightarrow (x_i \vee \overline{x_j}) \wedge (\overline{x_i} \vee x_j)$$

  - $Cannot\_Link(x_i, x_j)$

$$(x_i \wedge \overline{x_j}) \vee (\overline{x_i} \wedge x_j) \Longleftrightarrow (x_i \vee x_j) \wedge (\overline{x_i} \vee \overline{x_j})$$

  - Diameter constraints $D \leq \alpha$:
    for all (i, j) such that $d_{ij} > \alpha$, add $Cannot\_Link(x_i, x_j)$
  - Margin constraints $S \geq \beta$:
    for all (i, j) such that $d_{ij} < \beta$, add $Must\_Link(x_i, x_j)$

Minimizing the maximal diameter

- Observation: the maximal diameter D is one of the values d_{ij}
- Optimization by dichotomic search:
  - sort all the distinct values d_{ij} in increasing order, set upper/lower bounds
  - repeat
    - choose D the middle value
    - solve 2-SAT problem P with D
    - if P is satisfiable then revise upper bound, else revise lower bound
- Complexity:
  - solving 2-SAT problem P: $O(n^2)$
  - optimization in the worst case: $O(n^2 log(n))$

Conceptual clustering using SAT

Métivier *et al.*, IDA 2012

- Constraint-based language

$$\texttt{isClustering}([X_1,...,X_k]) \equiv \begin{cases} \bigwedge_{1\le i\le k} \texttt{isNotEmpty}(X_i) \wedge \\ \texttt{coverTransactions}([X_1,...,X_k]) \wedge \\ \texttt{noOverlapTransactions}([X_1,...,X_k]) \wedge \\ \texttt{canonical}([X_1,...,X_k]) \end{cases}$$

- Queries to focus on more interesting clustering solutions
- Several problems formulated by queries, ex. balanced clustering:

$$q_3([X_1,...,X_k]) \equiv \begin{cases} \texttt{isClustering}([X_1,...,X_k]) \wedge \\ \bigwedge_{1\le i<j\le m, d(t_i,t_j)<\beta} \texttt{mustLink}(t_i,t_j) \wedge \\ \bigwedge_{1\le i<j\le m, d(t_i,t_j)>\alpha} \texttt{cannotLink}(t_i,t_j) \wedge \\ \bigwedge_{1\le i<j\le k} \mid \texttt{size}(X_i) - \texttt{size}(X_j) \mid \le \Delta \times m \end{cases}$$

SAT encoding

- Variables: $T_{ij} = 1$ iff transaction t belongs to cluster j
- Constraints in language encoded into SAT

$$\texttt{coverTransactions}([X_1, \ldots, X_k]) \equiv \bigwedge_{t \in \mathcal{T}} \bigvee_{j} T_{tj}$$

$$\texttt{mustLink}(t_1, t_2) \equiv \bigwedge_{j} (\neg T_{t_1 j} \vee T_{t_2 j}) \wedge (T_{t_1 j} \vee \neg T_{t_2 j})$$

$$\texttt{cannotLink}(t_1, t_2) \equiv \bigwedge_{j} (\neg T_{t_1 j} \vee \neg T_{t_2 j}) \wedge (T_{t_1 j} \vee T_{t_2 j})$$

- Ensuring completeness: having a solution s, add $\neg s$ to the CNF and restart to find another solution, until failure.

Outline

Clustering with ILP: modeling

Conceptual model:

$$\begin{aligned} \text{minimize} \quad & quality(\mathcal{C}), \\ \text{subject to} \quad & C_1 \cap C_2 = \emptyset \quad \forall C_1, C_2 \in \mathcal{C} \\ & |\bigcup_{C \in \mathcal{C}} C| = n \\ & |\mathcal{C}| = k \end{aligned}$$

Here, **Boolean encoding**: $x_{ik} = [o_i \in C_k]$

Clustering with ILP: modeling

Boolean encoding: $x_{ik} = [o_i \in C_k]$

$$\begin{aligned} \text{minimize} \quad & \mathit{quality}(\mathcal{C}), \\ \text{subject to} \quad & C_1 \cap C_2 = \emptyset \quad \forall C_1, C_2 \\ & |\bigcup_{C \in \mathcal{C}} C| = n \\ & |\mathcal{C}| = k \end{aligned}$$

$$\begin{aligned} \text{minimize} \quad & \mathit{quality}(x), \\ \text{subject to} \quad & \sum_k x_{ik} = 1 \quad \forall i \\ & ,, \\ & \sum_i x_{ik} \geq 1 \quad \forall k \end{aligned}$$

Clustering with ILP: constraints

Boolean encoding: $x_{ik} = [o_i \in C_k]$

Additional constraints:

- Must-Link(i,j) $\equiv x_{ik} = x_{jk} \quad \forall k$
- Cannot-Link(i,j) $\equiv x_{ik} + x_{jk} \leq 1 \quad \forall k$
- Margin-min(β) $\equiv d(o_i, o_j) < D \rightarrow$ Must-Link(i, j)
- Diameter-max(β) $\equiv d(o_i, o_j) > D \rightarrow$ Cannot-Link(i, j)
- Capacity-max(k,β) $\equiv \sum_i x_{ic} \leq \beta$

All these constraints are linear.

Minimizing maximal diameter

Objective: minimizing the maximal diameter

$$
\begin{aligned}
\text{minimize} \quad & Z \\
\text{subject to} \quad & \sum_k x_{ik} = 1 & \forall i \\
& \sum_i x_{ik} \geq 1 & \forall k \\
& Z = max_{c \in [1,k], o_i, o_j \in C_c}(d(o_i, o_j)) \\
& \leftrightarrow \ d(o_i, o_j) * x_{ik} * x_{jk} \leq Z \qquad \forall ijk \quad (quadratic) \\
& \leftrightarrow \ d_{ij} * x_{ik} + d_{ij} * x_{jk} - d_{ij} \leq Z \qquad \forall ijk \quad (linear)
\end{aligned}
$$

Requires $O(n^2 k)$ constraints to encode the diameter.

[Rao, 1979]

Clustering with ILP, other objectives

- WCSD criterion $W = \sum_{k \in [1,K]} \sum_{o_i, o_j \in C_k} d(i,j)^2$

$$\leftrightarrow W = \sum_k \sum_{i,j} d(o_i, o_j)^2 * x_{ik} * x_{jk}$$

Linearization requires $O(n^2)$ variables and $O(n^2 k)$ constraints of: $y_{ij} >= x_{ik} + x_{jk} - 1$

- WCSS criterion $V = 1/2 \sum_{k \in [1,K]} \frac{\sum_{o_i, o_j \in C_k} d(o_i, o_j)^2}{|C_k|}$

$$\leftrightarrow V = 1/2 \sum_k \frac{\sum_{i,j} d(o_i, o_j)^2 * x_{ik} * x_{jk}}{\sum_i x_{ik}}$$

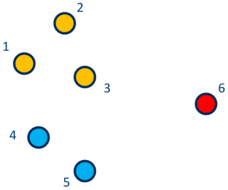
Linearization...?

Not suitable for ILP?

Dual view of clustering

- Primal view: every variable a point-in-cluster, constraint per cluster
- Dual view: every variable a possible cluster, constraint per point

Example ($k = 3$, subset of clusters):



x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	
0	0	1	0	1	0	0	0	1	1	>=1
0	0	1	1	0	0	0	1	1	0	>=1
0	0	1	0	1	0	1	0	0	1	>=1
1	1	0	1	0	0	0	0	1	1	>=1
1	1	0	0	0	0	1	1	0	0	>=1
1	0	0	1	0	1	1	0	0	1	>=1
1	1	1	1	1	1	1	1	1	1	=3

Dual view of clustering

Dual view: every variable a possible cluster

Advantage:

- Weight of each cluster can be precomputed
- Constraints ML/CL/Capacity/Diameter/Margin also precomputed
  $\rightarrow$ can preprocess constraints on every cluster individually
  $\rightarrow$ remove cluster from formulation if it violates a constraint

Disadvantage:

- Requires $O(2^n)$ variables

Can we overcome the exponential blow-up?

Clustering using ILP: 2 approaches

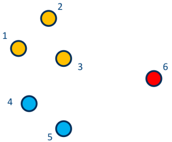
- Restricted problem: consider only a subset of all possible clusters
- Using column generation: consider each time only a subset of clusters, generate a new one if needed

Approach 1, restricted problem

Mueller and Kramer, DS 2010

- Conceptual clustering: each cluster represents a given concept
- In 2-step approach: mine possible patterns/clusters, then compose a clustering

$\rightarrow$ given set of candidate clusters, problem is standard ILP



x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	
0	0	1	0	1	0	0	0	1	1	>=1
0	0	1	1	0	0	0	1	1	0	>=1
0	0	1	0	1	0	1	0	0	1	>=1
1	1	0	1	0	0	0	0	1	1	>=1
1	1	0	0	0	0	1	1	0	0	>=1
1	0	0	1	0	1	1	0	0	1	>=1
1	1	1	1	1	1	1	1	1	1	=3

$$\text{minimize} \quad \sum_s w_s * c_s$$

$$\text{subject to} \quad \sum_s m_{is} * c_s = 1 \qquad \forall i$$

$$\sum_s c_s = k$$

(m_{is} = membership of point i)

Objective function

$$\begin{aligned} \text{minimize} \quad & \sum_s w_s * c_s \\ \text{subject to} \quad & \sum_s m_{is} * c_s = 1 && \forall i \\ & \sum_s c_s = k \end{aligned}$$

Objective aggregations:

- minSumQuality: $\sum_s w_s * c_s$
- minMeanQuality: $\frac{\sum_s w_s * c_s}{\sum_s 1}$
- minMaxQuality: $\mathrm{M}, \mathrm{M} \geq w_s * c_s \quad \forall s$

Constraints

$$\begin{aligned} \text{minimize} \quad & \sum_s w_s * c_s \\ \text{subject to} \quad & \sum_s m_{is} * c_s = 1 \qquad \forall i \\ & \sum_s c_s = k \end{aligned}$$

Constraints:

- completeness: $\sum_s m_{is} * c_s = 1$
- overlap: $\alpha \leq \sum_s m_{is} * c_s \leq \beta$
- numberClusters: $\alpha \leq \sum_s c_s \leq \beta$
- conditional cluster groups (clausal): $\sum_t c_t \leq 1$

Combining with existing algorithms

Ouali *et al.*, IJCAI 2016

Conceptual clustering: transactions $\mathcal{T}$, items $\mathcal{I}$

- Step 1: computed closed itemsets (candidate clusters) $\mathcal{C}$ using LCM
- Step 2: compose a clustering from candidate clusters

$$
\begin{aligned}
\text{optimize} \quad & \sum_{c \in \mathcal{C}} v_c * x_c \\
\text{subject to} \quad & (1) \quad \sum_{c \in \mathcal{C}} a_{t,c} * x_c = 1 \qquad \forall t \in \mathcal{T} \\
& (2) \quad \sum_{c \in \mathcal{C}} x_c = k \\
& x_c \in \{0, 1\}, c \in \mathcal{C}
\end{aligned}
$$

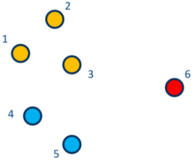
Combining with existing algorithms

Ouali *et al.*, IJCAI 2016

Co-clustering extension: k clusters covering both $\mathcal{T}$ and $\mathcal{I}$ without overlap

$$
\begin{array}{lll}
\text{optimize} & \displaystyle\sum_{c \in \mathcal{C}} v_c * x_c & \\
\text{subject to} & (1) \quad \displaystyle\sum_{c \in \mathcal{C}} a_{t,c} * x_c = 1 & \forall t \in \mathcal{T} \\
& (2) \quad \displaystyle\sum_{c \in \mathcal{C}} x_c = k & \\
& (2') \quad k_{min} \leq k \leq k_{max} & \\
& (3) \quad \displaystyle\sum_{c \in \mathcal{C}} w_{i,c} * x_c = 1 & \forall t \in \mathcal{I} \\
& k \in \mathbb{N}, x_c \in \{0,1\}, c \in \mathcal{C} &
\end{array}
$$

Approach 2, column generation



x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	
0	0	1	0	1	0	0	0	1	1	>=1
0	0	1	1	0	0	0	1	1	0	>=1
0	0	1	0	1	0	1	0	0	1	>=1
1	1	0	1	0	0	0	0	1	1	>=1
1	1	0	0	0	0	1	1	0	0	>=1
1	0	0	1	0	1	1	0	0	1	>=1
1	1	1	1	1	1	1	1	1	1	=3

Observe: only k of 2^n cluster will have $c_s = 1$.

→ Let's generate the clusters as needed: *column generation*

Column generation for dissimilarity-based constrained clustering

- Basic idea:
  - Start with a few initial clusters
  - Find optimal LP solution to this *restricted* problem
  - Find the *most violated* cluster for this solution
  - Add this cluster and repeat.
- WCSS criterion without constraints: [du Merle *et al.* 1999, Aloise *et al.* 2009]
- WCSS criterion with constraints: [Babaki *et al.* 2014]
  - adding constraints to subproblem
  - solve set enumeration problem using constrained branch-and-bound

Outline

Constrained clustering using CP

Two approaches:

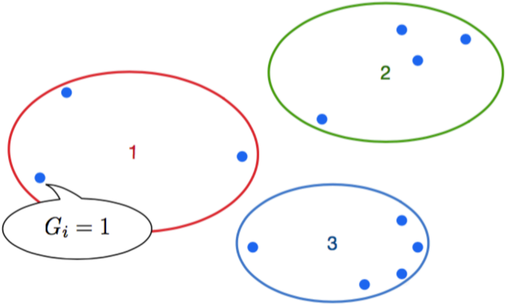
- 1-step: finding a clustering under constraints from data
  - conceptual clustering: Khiari *et al.* CP 2010, Guns *et al.* TKDE 2013
  - dissimilarity-based clustering: Dao *et al.* ECML/PKDD 2013, AIJ 2017
- 2-step: combining with an algorithm that generates cluster candidates then composing a clustering
  - conceptual clustering: Chabert *et al.* CP 2017

CP Framework for Constrained Clustering

Dao & al., ECML/PKDD 2013, AIJ 2017

Modeling a partition:

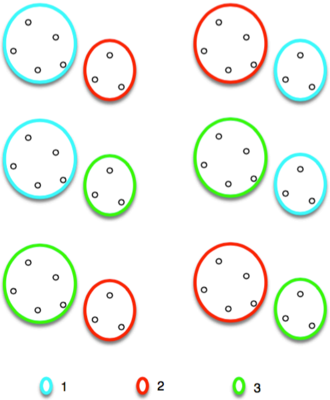
- Clusters identified by their index $1, \ldots, K$, $K_{min} \leq K \leq K_{max}$
- Decision variables $G_1, ..., G_N \in \{1, ..., K_{max}\}$
  $G_i = k$: point i is grouped in the cluster k



Partitioning: breaking symmetries

- Symmetries: one partition corresponds to different assignments
- Breaking symmetries:
  - First point in cluster 1
  - A cluster number k is created only if the number $k-1$ has been used
- Expressed by the CP constraint:

$$Precede([G_1, .., G_N], [1, K_{max}])$$



Partitioning: number of clusters

- At most K_{max} clusters: $Dom(G_i) \in [1, K_{max}]$
- At least K_{min} clusters: cardinality constraint

$$\#\{i \in [1, N] \mid G_i = K_{min}\} \geq 1$$

User-constraints

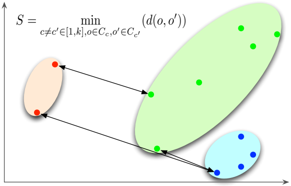
- Instance-level constraints
  - Must-link constraint $ML(i, j)$: $G_i = G_j$
  - Cannot-link constraint $CL(i, j)$: $G_i \neq G_j$

- All popular cluster-level constraints can be expressed by CP constraints
- Minimal size α of clusters

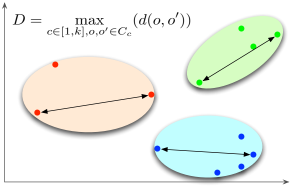
$$\forall i \in [1, N], \quad \#\{j \in [1, N] \mid G_i = G_j\} \geq \alpha$$

Optimization criteria

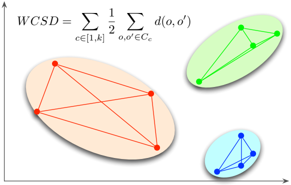
Each of the criteria can be modeled directly using CP constraints.



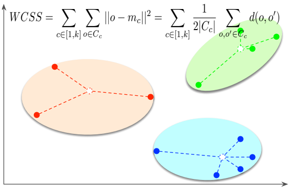
Maximizing the minimal split



Minimizing the maximal diameter



Minimizing the WCSD



Minimizing the WCSS

Diameter criterion

- Minimizing the maximal diameter
  - D represents the maximal diameter: minimize D
  - Any two points i, j with $d(i, j) > D$ must be in different clusters:

$$d(i, j) > D \;\rightarrow\; G_i \neq G_j \tag{1}$$

- Direct modeling
  - Modeling (1) using logical variables and constraints
  - Needs $O(N^2)$ of variables and constraints
  - Many of them do not have useful propagation

A global constraint for the diameter criterion

$$diameter(D, [G_1, .., G_N], d) \stackrel{def}{=} \forall i < j \in [1, N], d(i, j) > D \rightarrow G_i \neq G_j$$

Filtering algorithm

$Dom(D) = [\underline{D}, \overline{D})$

if $\overline{D}$ *has been changed* **then**
  $stack \leftarrow \{i \in [1, N] \mid G_i$ is instantiated$\}$
else
  $stack \leftarrow \{i \in [1, N] \mid$
    G_i has just been instantiated$\}$
foreach $i \in stack$ **do**
  for $j \leftarrow 1$ **to** n **do**
    if $d(i, j) \geq \overline{D}$ **then**
      remove $val(G_i)$ from $Dom(G_j)$
    if G_j *is instantiated* $\wedge\ G_i = G_j$ **then**
      $\underline{D} \leftarrow \max(\underline{D}, d(i, j))$;

Ensure the same consistency but better computation time:

- Consider only potential cases
- Avoid examining unuseful candidates

Global constraints for other criteria

- Split criterion
  $split(S, [G_1, ..., G_N], d) \stackrel{def}{=} \forall i < j \in [1, N], d(i,j) < S \; \rightarrow \; G_i = G_j$

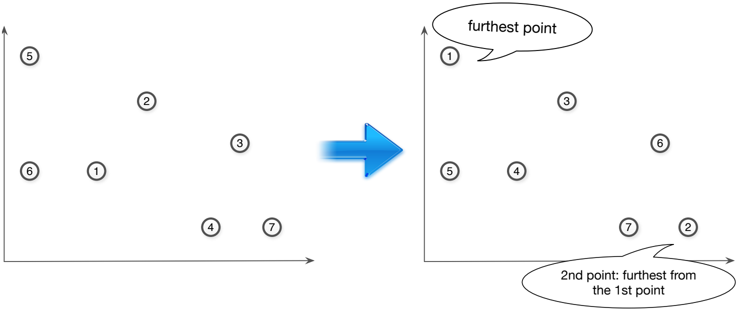
- WCSD criterion (ICTAI 2013)
  $wcsd(W, [G_1, ..., G_N], d) \stackrel{def}{=} W = \sum_{k \in [1,K]} \sum_{o_i, o_j \in C_k} d(i,j)^2$

- WCSS criterion (CP 2015)
  $wcss(V, [G_1, ..., G_N], d) \stackrel{def}{=} V = \sum_{k \in [1,K]} \sum_{o_i \in C_k} ||o_i - m_k||^2$
  where m_k is the centroid of the cluster C_k

- Better computation time (split)
- Better propagation and computation time (WCSD, WCSS)

Search strategies

- Search strategies depend on the optimization criterion
- Partition symmetry breaking is based on the indices of points ⇒ points are reordered using FPF (Furthest Point First) algorithm (Gonzales, 1985): points that are far from each other have a small index



2-step constrained clustering using CP

Method:

- Step 1: extract all formal concepts $\mathcal{F}$ with a dedicated tool (LCM)
- Step 2: use CP to select a subset of $\mathcal{F}$ forming the clustering

CP model for step 2 using set variable P: the set of selected concepts [Chabert *et al.*, CP 2017]

- partition: each $t \in \mathcal{T}$ is covered by one concept

$$\forall t \in \mathcal{T},\ |CF(t) \cap P| = 1$$

- k selected concepts

$$|P| = k$$

Conceptual clustering as an exact cover problem

In the selected concepts P, each object is covered exactly once

$$\forall t \in \mathcal{T}, \#\{C \in P \mid t \in C\} = 1$$

$\longrightarrow$ a conceptual clustering problem can be seen as an exact cover problem

Global constraint $exactCoverQ_{\mathcal{T},P,q}(selected, MinQ, MaxQ)$ [Chabert *et al.*, 2020]:

- the *selected* variables assigned to *true* correspond to an exact cover of $(\mathcal{T}, P)$
- *MinQ* and *MaxQ* variables are assigned to the minimum and maximum quality associated with the selected subsets

Outline

Making clustering useful using constraints

Cluster friend network in groups for different diner parties

- the difference in age is minimized
- equal number of males and females
- each person should have at least 5 other persons in the same group sharing the same hobby

More meaningful constraints

- Objects can be described by different types of information
- Constraints not only generated from ground truth label
- Constraints can be provided by expert and capture what makes the clustering useful in the domain

Actionable clustering

Dao *et al.*, ECAI 2016

Data: each instance $x \in \mathcal{X}$ is described by:

- a set of features: to compute distances between instances and the clustering objective function
- a set of properties: on which constraints are stated

Actionable clustering

- Constraints making clustering actionable
  - cardinality constraints
  - geometric constraints
  - density constraints
  - complex logic constraints

CP offers a natural modeling of these constraints

Minimal clustering modification

Cluster friend network in groups for different diner parties

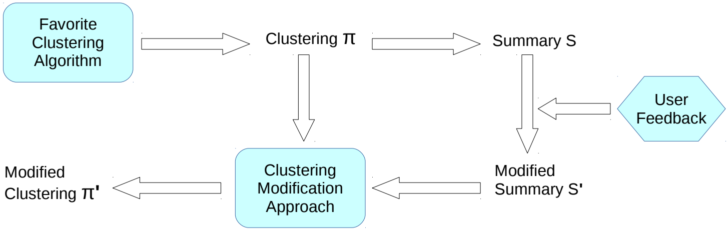
- A very cohesive clustering already obtained, but
  - the range of ages for some clusters is too large
  - one cluster has too many males compared to females
- Simply removing data points to get desirable clusters undermines the intended use
- Applying a constrained clustering algorithm does not guarantee to find a similar clustering

Minimal clustering modification

- Finding a similar clustering by minimal modifications
- Removing the undesirable properties

Minimal clustering modification problem

Kuo *et al.*, AAAI 2017



Minimally modify Π to obtain Π' to satisfy S'

$$\begin{aligned} \text{minimize}_{\Pi'} \quad & d(\Pi', \Pi) \\ \text{subject to} \quad & \Pi' \text{ satisfies } S' \end{aligned}$$

Minimal clustering modification with restriction on diameters

- Problem: minimally modify Π such that along l dimensions the maximum diameter is reduced.
- Theorems:
  - The problem with $l = 2$ is NP-Complete
  - Suppose the number of dimensions along which the maximum diameter must be reduce is a variable l. The reclustering problem is NP-Complete for $k \geq 3$.
- Formulation for diameter constraints:

$$\forall c \in [1, k], \forall t \in [1, l], \max_{i,j \in [1,n]} (C[c, i] C[c, j] D_{tij}) \leq D'_{ct}$$

$O(n^2 k)$ constraints, not efficient

ILP formulation

Data $X \subset \mathbb{R}^{n \times f}$, $\forall t \in [1, l]$ let:

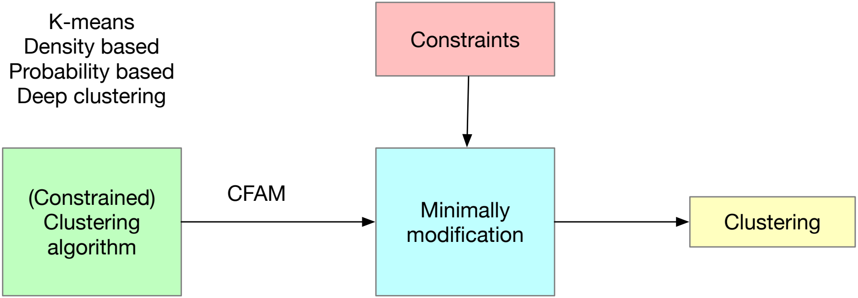
$$M_l[t] \leftarrow \min_{i=1,\dots,n} \{X[i,t]\} \; \forall t = 1, \dots, f$$
$$M_u[t] \leftarrow \max_{i=1,\dots,n} \{X[i,t]\} \; \forall t = 1, \dots, f$$

More efficient ILP formulation:

$$\begin{aligned}
&\underset{z,C,L,H}{\text{minimize}} \quad \sum_{i=1}^{n} z[i] \\
&\text{subject to} \\
&\quad \forall c = 1, \dots, k,\ \forall i = 1, \dots, n,\ C[c,i] = \mathbb{I}[\Pi'[i] = c] \\
&\quad \forall i = 1, \dots, n,\ z[i] = \mathbb{I}[\Pi'[i] \neq \Pi[i]] \\
&\quad \forall c = 1, \dots, k,\ \ \forall t = 1, \dots, f, \\
&\qquad L[c,t] = \min_{i=1,\dots,n} \{C[c,i](X[i,t] - M_u[t])\} + M_u[t] \\
&\qquad H[c,t] = \max_{i=1,\dots,n} \{C[c,i](X[i,t] - M_l[t])\} + M_l[t] \\
&\qquad H[c,t] - L[c,t] \leq \mathcal{D}'[c,t]
\end{aligned}$$

Distance between two partitions measured by number of changes

Post-process clustering algorithms with constraint

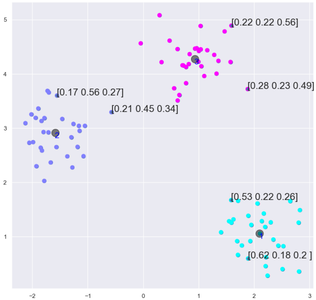


Exploiting current partition

Nghiem *et al.*, DS 2020

- Cluster Fractional Allocation Matrix $S \in \mathbb{R}^{n \times k}$, S_{ic} score of point i belonging to cluster c
  - Distance-based clustering: $S_{ic} = ||x_i - \mu_c||$
  - Deep/probability-based clustering: S_{ij} is the soft-assignment
- Minimally modification subject to constraints:

$$\text{optimize}_{\Pi'} \sum_i S_{i\Pi'[i]}$$



Take home messages

- Strong points:
  - declarative approaches offer frameworks modeling various constrained clustering settings
  - numerous constraints and objective functions can be integrated
- Weak points:
  - scalability
- Needs:
  - considering several views of the problem
  - appropriate choice of variables and/or constraint expressions
  - constraint propagation designs and heuristics
- Open issues:
  - scalability
  - interactive/incremental clustering
  - if not satisfying all constraints
  - if constraints are noisy

Acknowledgments

Some parts of this talk were taken from the tutorial at eEGC 2016 prepared with:

- Christel Vrain (LIFO)
- Tias Guns (KU Leuven)

Our work in this talk are in collaboration with:

- Khanh-Chuong Duong, Nguyen-Viet-Dung Nghiem, Christel Vrain (LIFO)
- Ian Davidson, Chia-Tung Kuo (UC Davis)
- S. S. Ravi (Univ. at Albany)