Investigating a three-way clustering approach for handling missing data

Project supervised by

Dr Zaineb CHELLY DAGDIA

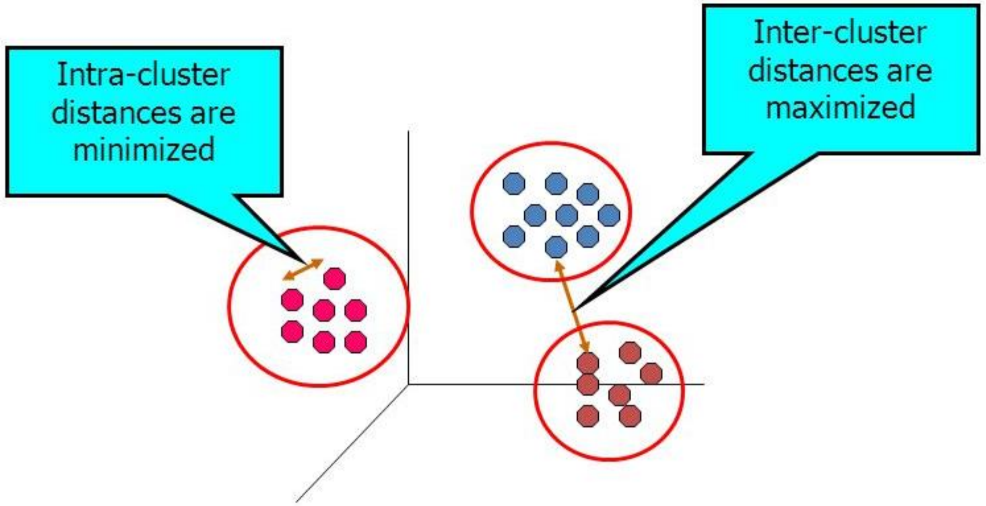
zaineb.chelly-dagdia@uvsq.fr

UVSQ — Paris Saclay University

Context (1/2)

Clustering is the process of grouping a set of homogeneous objects into subsets – named ***clusters*** – in such a way that objects in the same cluster are more similar to each other than to those in other clusters.

CustomerID	Genre	Age	Annual Income (k$)	Spending Score (1-100)
1	Male	19	15	39
2	Male	21	15	81
3	Female	20	16	6
4	Female	23	16	77
5	Female	31	17	40
…	………..	…	…	…



Context (2/2): key challenge

Clustering of data containing **missing values!**

CustomerID	Genre	Age	Annual Income (k$)	Spending Score (1-100)
1	Male	19	15	39
2	Male	21	15	81
3	Female	20	16	6
4	Female	23	16	77
5	Female	31	17	40
…	………..	…	…	…
54	**Male**	**?**	**14**	**?**
…	………..	…	…	…
9874	**Female**	**22**	**?**	**45**

Problem statement (1/2)

Jeopardizes the quality and reliability of the clustering results

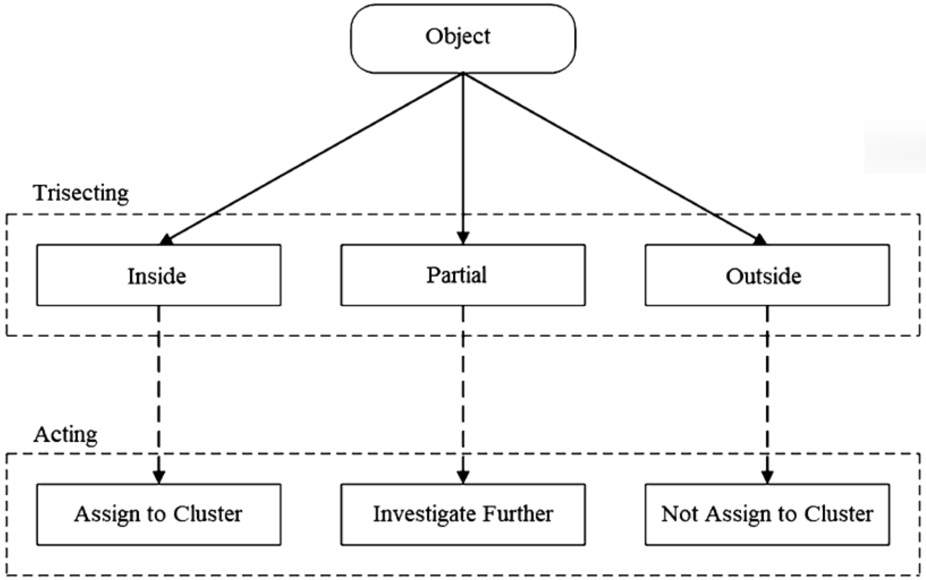
Common strategies to deal with missing values

Handle the data with missing values at the model level instead of modifying the data itself.

- Pre-processing
  - Deletion
  - Imputations
- Incorporate additional mechanisms in the clustering model
  - Assigning objects with missing values to a cluster having a high number of missing values
  - Creating a separate cluster for containing all objects with missing values
  - ...
  - A three-way approach for clustering of data containing missing values

Problem statement (2/2)

A three-way approach for clustering



$$Inside(c_k) = \{o_i \in U \mid e(c_k, o_i) \geq \alpha\},$$

$$Partial(c_k) = \{o_i \in U \mid \beta < e(c_k, o_i) < \alpha\},$$

$$Outside(c_k) = \{o_i \in U \mid e(c_k, o_i) \leq \beta\}.$$

(*where* ***e*** is an evaluation function)

The three-way decisions and the quality of the resulting three regions <u>are critically controlled</u> and defined based on a pair of thresholds (α,β).

An <u>automatic determination of thresholds</u> is needed to achieve significant results in terms of the three-way clustering main criteria *(accuracy, generality)*!

Approach to investigate

- **Main idea: Formulize the problem as a game between *accuracy* and *generality* using simulated missing values (from the non-missing data set) & search for an effective trade-off based solution.**
- **Used technique: Game-Theoretic Rough Set (GTRS)** - provides a game-theoretic environment for reading a trade-off solution between multiple criteria that are realized as game players.

Main process:

1) Divide the dataset into a set of objects with no missing values (called *C*) and a set of objects with missing values (called *M*).
2) Apply a clustering algorithm on *C*.
3) Randomly remove values from *C* by following the percentage of missing values in *M*.
4) Divide *C* into U_c and U_m; where U_c is the set of objects with no missing values and U_m is the set of objects with simulated missing values.
5) Formulize the problem as a game.
6) Apply GTRS to determine selected game strategies and corresponding new thresholds ($\boldsymbol{\alpha}'$, $\boldsymbol{\beta}'$) using U_c and U_m
7) Evaluate objects in *M*.
8) Use the new final values of the thresholds ($\boldsymbol{\alpha}'$, $\boldsymbol{\beta}'$) for assigning objects to different regions of a cluster.

Project tasks (1/2)

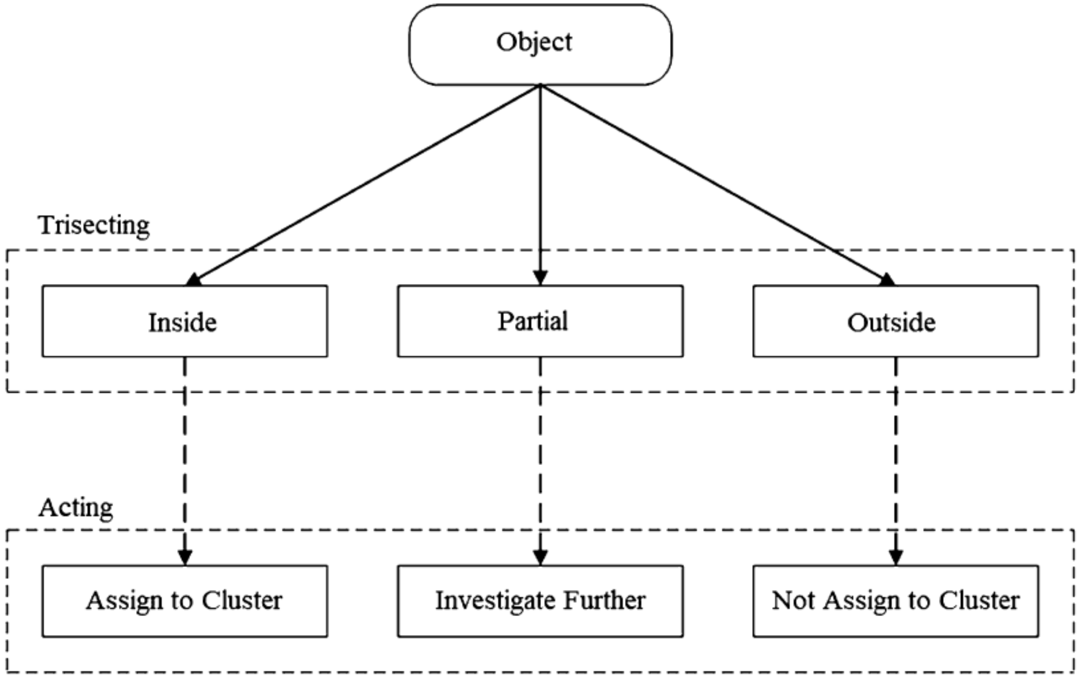
1. Understand the in-depth functioning of the algorithm as well as the technical details of GTRS;
2. Implement the GTRS three-way clustering algorithm;
3. Test the code using a sample dataset *(given in the corresponding research paper)* to validate the algorithm's implementation;
4. Test using some UCI machine learning datasets *(with respect to the clustering algorithm that you will select and apply);*
5. Investigate different settings of the algorithm;
6. Present visually the three-way clustering results;
7. Identify some limitations of the used algorithm;
8. Propose some improvements of the algorithm;
9. Write the technical report;
10. Present the conducted work.

Project tasks (2/2): increase your score!

The following different options can be considered to increase your project's score. You can either select one or combine several options:

- Apply the algorithm using a real world application (dataset will be given)
- Investigate the Nash equilibrium aspect coupled with GTRS
- Investigate other game strategies with GTRS
- Investigate more uncertainty aspects within the GTRS based algorithm (e.g., fuzzy clustering)

A three-way clustering approach for handling missing data using GTRS



$$Inside(c_k) = \{o_i \in U \mid e(c_k, o_i) \geq \alpha\},$$

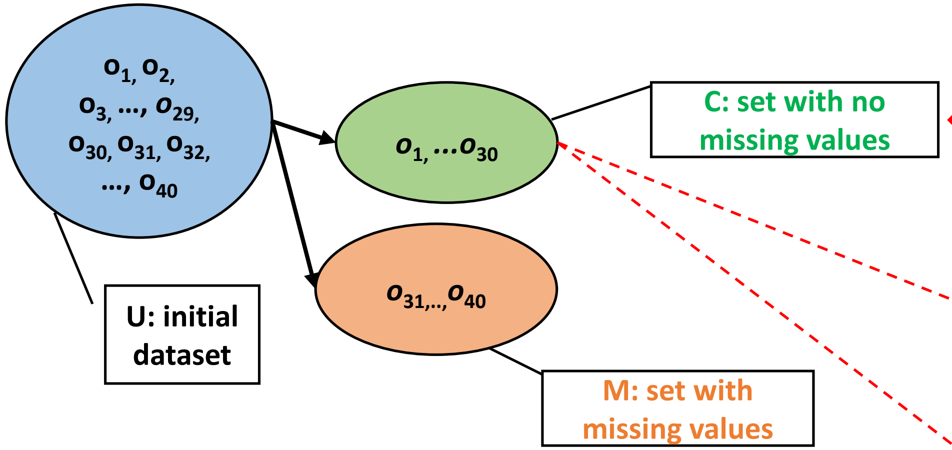
$$Partial(c_k) = \{o_i \in U \mid \beta < e(c_k, o_i) < \alpha\},$$

$$Outside(c_k) = \{o_i \in U \mid e(c_k, o_i) \leq \beta\}.$$

- ***$e(c_k, o_i)$:*** an evaluation function representing the relationship or association between a certain cluster c_k and a particular object o_i
- ***(α, β):*** some thresholds

- An object is included in the ***Inside(c_k)*** when its evaluation is above or equal to threshold ***α***.
- An object is included in the ***Outside(c_k)*** when its evaluation is below or equal to threshold ***β***.
- An object is included in the ***Partial(c_k)*** when its evaluation is between the two thresholds.

The thresholds *(α, β)* control the inclusion in different regions and its different settings lead to different regions.
→ How to determine the thresholds automatically is an important research issue in this context.

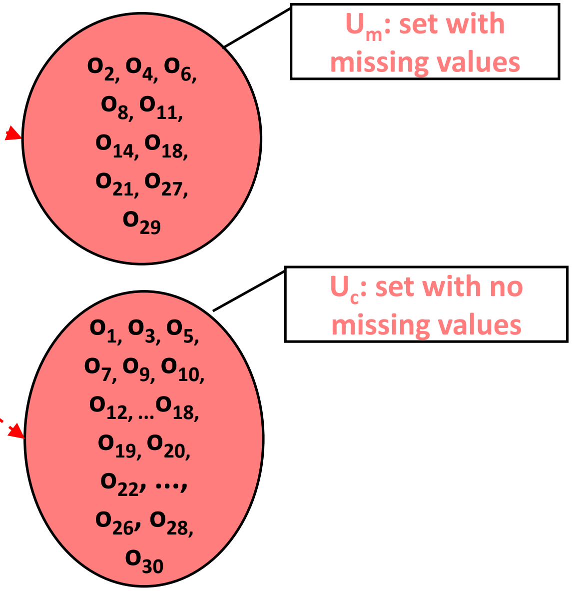


The objects in set *C* are clustered using one of the conventional algorithms such as K-means. (k=2)

Set C

Table 1
Sample dataset with missing data, assumed missing values are marked with a *.

	A_1	A_2	A_3	A_4		A_1	A_2	A_3	A_4		A_1	A_2	A_3	A_4
o_1	5.9	3.2	4.8	2	o_{11}	5.6*	2.9	4.1*	1.5	o_{21}	6.3	2.7*	4.9	1.8*
o_2	6.1	2.8*	4.2	1.5*	o_{12}	5.5	2.5	4	1.5	o_{22}	6.2	2.8	4.8	1.8
o_3	6.4	2.8	4.6	1.3	o_{13}	5.5	2.6	4.4	1.4	o_{23}	5.9	3	5.1	1.8
o_4	6.4*	2.5	4.3*	1.4	o_{14}	6.1*	2.7	4.6*	1.4	o_{24}	6.4	2.8	5.6	2.1
o_5	6.3	2.3	4.4	1.5	o_{15}	5.8	2.6	4	1.4	o_{25}	6.5	3	5.5	1.8
o_6	6.3	2.8*	4.9	1.6*	o_{16}	5.8	2.7	5.1	1.9	o_{26}	6.3	2.8	5.1	1.5
o_7	5.5	2.4	3.8	1.3	o_{17}	5.7	2.5	5	2	o_{27}	6.1*	2.7	5.6*	1.5
o_8	5.8	2.7*	4	1.4*	o_{18}	6.1	2.8*	5.6	2.2*	o_{28}	6.4	3.1	5.5	1.8
o_9	5.5	2.4	3.7	1.2	o_{19}	6	2.2	5	1.5	o_{29}	6*	2.9	4.8*	1.6
o_{10}	6	2.8	4.5	1.4	o_{20}	5.6	2.8	4.9	2	o_{30}	5.9	3	5.1	1.8



The rate of missing values is kept similar to the rate of missing values in dataset *U*.
→ This means that if the original dataset contains 30% of objects with missing values, than approximately 30% of objects will be randomly selected from *C* for induced missing values.

This step will help in selecting suitable values for *(α, β)* thresholds

Step 1: divide U into C and M, and cluster C

Step 2: construct an incomplete dataset from C and determine *(α, β)*

Table 1
Sample dataset with missing data, assumed missing values are marked with a *.

	A_1	A_2	A_3	A_4		A_1	A_2	A_3	A_4		A_1	A_2	A_3	A_4
o_1	5.9	3.2	4.8	2	o_{11}	5.6*	2.9	4.1*	1.5	o_{21}	6.3	2.7*	4.9	1.8*
o_2	6.1	2.8*	4.2	1.5*	o_{12}	5.5	2.5	4	1.5	o_{22}	6.2	2.8	4.8	1.8
o_3	6.4	2.8	4.6	1.3	o_{13}	5.5	2.6	4.4	1.4	o_{23}	5.9	3	5.1	1.8
o_4	6.4*	2.5	4.3*	1.4	o_{14}	6.1*	2.7	4.6*	1.4	o_{24}	6.4	2.8	5.6	2.1
o_5	6.3	2.3	4.4	1.5	o_{15}	5.8	2.6	4	1.4	o_{25}	6.5	3	5.5	1.8
o_6	6.3	2.8*	4.9	1.6*	o_{16}	5.8	2.7	5.1	1.9	o_{26}	6.3	2.8	5.1	1.5
o_7	5.5	2.4	3.8	1.3	o_{17}	5.7	2.5	5	2	o_{27}	6.1*	2.7	5.6*	1.5
o_8	5.8	2.7*	4	1.4*	o_{18}	6.1	2.8*	5.6	2.2*	o_{28}	6.4	3.1	5.5	1.8
o_9	5.5	2.4	3.7	1.2	o_{19}	6	2.2	5	1.5	o_{29}	6*	2.9	4.8*	1.6
o_{10}	6	2.8	4.5	1.4	o_{20}	5.6	2.8	4.9	2	o_{30}	5.9	3	5.1	1.8

U_m: set with missing values

$o_2, o_4, o_6, o_8, o_{11}, o_{14}, o_{18}, o_{21}, o_{27}, o_{29}$

U_c: set with no missing values

$o_1, o_3, o_5, o_7, o_9, o_{10}, o_{12}, \ldots o_{18}, o_{19}, o_{20}, o_{22}, \ldots, o_{26}, o_{28}, o_{30}$

The rate of missing values is kept similar to the rate of missing values in dataset *U*.
→ This means that if the original dataset contains 30% of objects with missing values, than approximately 30% of objects will be randomly selected from *C* for induced missing values.

This step will help in selecting suitable values for *(α, β)* thresholds

Step 2: construct an incomplete dataset from C and determine *(α, β)*

$o_{31,..,}o_{40}$

M: set with missing values

$$Inside(c_k) = \{o_i \in U \mid e(c_k, o_i) \geq \alpha\},$$
$$Partial(c_k) = \{o_i \in U \mid \beta < e(c_k, o_i) < \alpha\},$$
$$Outside(c_k) = \{o_i \in U \mid e(c_k, o_i) \leq \beta\}.$$

Step 3: decide M via the three-way framework using the determined *(α, β)*

Table 1
Sample dataset with missing data, assumed missing values are marked with a *.

	A_1	A_2	A_3	A_4
o_1	5.9	3.2	4.8	2
o_2	6.1	2.8*	4.2	1.5*
o_3	6.4	2.8	4.6	1.3
o_4	6.4*	2.5	4.3*	1.4
o_5	6.3	2.3	4.4	1.5
o_6	6.3	2.8*	4.9	1.6*
o_7	5.5	2.4	3.8	1.3
o_8	5.8	2.7*	4	1.4*
o_9	5.5	2.4	3.7	1.2
o_{10}	6	2.8	4.5	1.4
o_{11}	5.6*	2.9	4.1*	1.5
o_{12}	5.5	2.5	4	1.5
o_{13}	5.5	2.6	4.4	1.4
o_{14}	6.1*	2.7	4.6*	1.4
o_{15}	5.8	2.6	4	1.4
o_{16}	5.8	2.7	5.1	1.9
o_{17}	5.7	2.5	5	2
o_{18}	6.1	2.8*	5.6	2.2*
o_{19}	6	2.2	5	1.5
o_{20}	5.6	2.8	4.9	2
o_{21}	6.3	2.7*	4.9	1.8*
o_{22}	6.2	2.8	4.8	1.8
o_{23}	5.9	3	5.1	1.8
o_{24}	6.4	2.8	5.6	2.1
o_{25}	6.5	3	5.5	1.8
o_{26}	6.3	2.8	5.1	1.5
o_{27}	6.1*	2.7	5.6*	1.5
o_{28}	6.4	3.1	5.5	1.8
o_{29}	6*	2.9	4.8*	1.6
o_{30}	5.9	3	5.1	1.8

- Table 1 = set C
- C contains information about 30 objects.
- Rows = objects {*o1, o2, o3, ..., o30*}
- Columns = 4 attributes {*A1, A2, A3, A4*}
- We assume that the missing rate in the original dataset (*U*) was 30%. Therefore, we also randomly considered 30% of the objects having missing values from *C*.
- The missing values are assumed to be the values with a * on top of them.
- The induced missing values will be used to compute the (α, β) thresholds which may be later on applied on the objects in M to determine three-way clustering for those objects

Step 1 : we apply K-means clustering on *C* with K = 2, which leads to the formation of two clusters:

- *c1 = {o1, ..., o15}*
- *c2 = {o16, ..., o30}.*

Step 2 : based on the objects with induced missing values, we aim to determine suitable thresholds that will do a good job of clustering these objects.

→ Application of the three-way clustering approach

1) compute the evaluation function $e(c_k, o_i)$

$$e(c_k, o_i) = \frac{\text{Number of } o_i \text{ neighbors belonging to } c_k}{\text{Total neighbors of } o_i}.$$

The evaluation function quantifies the relationship between an object o_i and cluster c_k and may be defined in different ways. We consider the evaluation function based on the relative number of nearest neighbours for object o_i belonging to cluster c_k.

To compute the neighbours, we need a certain distance metric:

$$d_{(i,\,j)} = \sqrt{\sum_{a=1}^{A} (o_i^a - o_j^a)^2},$$

where o^a_i is the value of the a^{th} attribute of the i^{th} object.

! We ignore the attributes with missing values while computing the distance. For instance, the distance between object *o2* and *o1* is determined as,

$$d_{(2,\,1)} = \sqrt{\sum_{a=1}^{A} (o_2^a - o_1^a)^2}$$
$$= \sqrt{(6.1 - 5.9)^2 + (* - 3.2)^2 + (4.2 - 4.8)^2 + (* - 2)^2}$$
$$= \sqrt{(6.1 - 5.9)^2 + (4.2 - 4.8)^2} = 0.63$$

➢ compute the distances of each o_i with missing values, from all the objects in U_c

➢ Sort these distances and compute the nearest neighbours for each o_i.

For instance, the distances of o_2 from all objects in U_c are $d(o_2, o_1) = 0.63$, $d(o_2, o_3) = 0.5$, ..., $d(o_2, o_{30}) = 0.92$. By sorting these distances, we find that the nearest neighbours, say 7 nearest neighbours (k=7) of o_2 are o_5, o_{10}, o_{15}, o_3, o_{22}, o_1 and o_{12}.

Once the neighbours are determined, we can compute the evaluation function $e(c_k, o_i)$. For instance, considering cluster c_1, based on the 7 neighbours of o_2, the evaluation function

$$e(c_1, o_2) = \frac{\text{Number of } o_2 \text{ neighbors belong to } c_1}{\text{Total neighbors of } o_2} = 6/7 = 0.86,$$

This means that 86% neighbours of o2 belongs to cluster c1.

$$e(c_2, o_2) = \frac{\text{Number of } o_2 \text{ neighbors belong to } c_2}{\text{Total number of } o_2 \text{ neighbors}} = 1/7 = 0.14.$$

The evaluation functions corresponding to the two clusters for all objects in U_m having missing values are given in Table 2.

Table 2
Evaluation function $e(c_k, o_i)$ values for objects in U_m.

	o_2	o_4	o_6	o_8	o_{11}	o_{14}	o_{18}	o_{21}	o_{27}	o_{29}
c_1	0.86	1	0.43	1	0.57	0.86	0	0.29	0.71	0
c_2	0.14	0	0.57	0	0.43	0.14	1	0.71	0.29	1

Once the evaluation functions are computed, we may use the three-way approach for inclusion of objects into one of the three regions.

$$Inside(c_k) = \{o_i \in U \mid e(c_k, o_i) \geq \alpha\},$$

$$Partial(c_k) = \{o_i \in U \mid \beta < e(c_k, o_i) < \alpha\},$$

$$Outside(c_k) = \{o_i \in U \mid e(c_k, o_i) \leq \beta\}.$$

For instance, if we assume thresholds *(α, β)* = (1, 0), the object *o*2 will be in the *Partial(c*1) and *Partial(c*2). → This will mean that object *o*2 is not being clustered. However, if we set thresholds *(α, β)* = (0.7, 0.25), than the object *o*2 will belong to cluster *c*1 and it will be in the outside region of the cluster *c*2.

→ different threshold settings will lead to different regions.

For instance, if we set thresholds *(α, β)* = (1, 0), than only objects *o*4, *o*8, *o*18 and *o*29 will be clustered. In particular, *o*4 and *o*8 will be in the *Inside(c*1) and *o*18 and *o*29 will be in *Inside(c*2). Since *o*4 and *o*8 belong to cluster *c*1 and *o*18 and *o*29 belong to *c*2, this means that we have accurately clustered these objects. However, we were only able to cluster 4 out of 10 objects. This suggests that with the thresholds setting of *(α, β)* = (1, 0) we are able to cluster only 4 out of 10 or 40% objects with all of these 4 objects being correctly placed in their appropriate clusters thereby leading to 100% accuracy.

On the other hand, if we set *(α, β)* = (0.5, 0.5), we will be able to cluster all the objects, however, 8 out of these 10 objects will be appropriately placed in their respective clusters thereby leading to 80% accuracy. Let us consider the formal definitions for the accuracy and generality of clustered objects.

$$Accuracy(\alpha, \beta) = \frac{\text{Correctly clustered objects}}{\text{Total clustered objects}},$$

$$Generality(\alpha, \beta) = \frac{\text{Total clustered objects}}{\text{Total objects in } U}.$$

Accuracy means how much accurately we cluster the objects with missing values and generality refers to percentage of objects that were actually being clustered.

(α, β) = (1, 0) -- Class = c1		
Inside	**Outside**	**Partial**
o4, o8	o18, o29	o2, o11, o6, o14, o21, o27

Correctly clustered

Acc = 4/4 = 100%
Gen = 4/10 = 40%

Step 1 : we apply K-means clustering on *C* with K = 2, which leads to the formation of two clusters:
- *c*1 = {*o*1, ..., *o*15}
- *c*2 = {*o*16, ..., *o*30}.

The accuracy and generality for different thresholds setting is summarized in Table 3.

Table 3
(Accuracy, Generality) values for different threshold values.

		α			
		1	0.85	0.7	0.5
β	0	$\boxed{(1.00, 0.40)}$	(1.00, 0.50)	(0.86, 0.60)	(0.70, 0.69)
	0.15	(1.00, 0.50)	(1.00, 0.60)	(0.86, 0.70)	(0.71, 0.80)
	0.3	(0.92, 0.60)	$\boxed{(0.93, 0.61)}$	(0.85, 0.80)	(0.83, 0.90)
	0.5	(0.86, 0.70)	(0.86, 0.80)	(0.84, 0.91)	(0.80, 1.00)

In general, modifying the thresholds to improve the generality or the number of clustered points may affect the accuracy and improving the accuracy may affect the generality.

→ How to determine the thresholds in order to achieve a balance between accuracy and generality is a critical issue in this context

Game theoretic rough sets (GTRS)

- Provides a game-theoretic environment for reading a **tradeoff solution between multiple criteria that are realized as game players.**
- It formulates strategies for players in the form of changes in thresholds in order to improve the overall quality of three-way decisions.
- Each player participates in the game by configuring the thresholds with the aim to maximize its benefits and utilities.

→ **The overall objective of a game in GTRS is to select suitable thresholds for three-way decisions, based on the available criteria.**

❖ A typical game in GTRS is defined as a tuple **{P, S, u}**, where:

- **P is a finite set of *n* players,**
- **$S = S_1 \times \ldots \times S_n$, where S_i is a finite set of strategies available to each player *i*. Each vector $s = (s_1, \ldots, s_n) \in S$ is called a strategy profile where player *i* plays strategy s_i ,**
- **$u = (u_1, \ldots, u_n)$ where $u_i : S \rightarrow \Re$ is a real-valued utility or payoff function for player *i*.**

→ **Nash equilibrium** is generally used to determine game solution or game outcome in GTRS

- A strategy profile **$(s_1, \ldots, s_n)$** is a Nash equilibrium, when, $u_i(s_i, s_{-i}) \geq u_i(s'_i, s_{-i})$, where $(s'_i \neq s_i)$

Game theoretic rough sets (GTRS)

- The players = **different criteria** that highlight various quality related aspects of three-way decisions such as **accuracy, generality, precision recall, uncertainty or cost**
- The strategies = different level of changes in the thresholds
- Suitable measures are defined for evaluating each criterion. **The values of these measures reflect the payoffs of different players or criteria.**

Table 4
A typical two-player game in GTRS.

		P_2		
		s_1	s_2	...
P_1	s_1	$u_1(s_1, s_1), u_2(s_1, s_1)$	$u_1(s_1, s_2), u_2(s_1, s_2)$	...
	s_2	$u_1(s_2, s_1), u_2(s_2, s_1)$	$u_1(s_2, s_2), u_2(s_2, s_2)$	...
	...	...	...	...

- **The players in the game are denoted by P1 and P2.**
- **Cells in the table correspond to strategy profiles.**
- **Each cell contains a pair of payoff functions based on their strategy profile. For example the top right cell corresponds to a strategy profile (s1, s2) which contains payoff functions u1(s1, s2) and u2(s1, s2).**

→ Playing the game results in the selection of Nash equilibrium which is utilized in determining a possible strategy profile and the associated thresholds.

Nash equilibrium (example)

A Simple Game

- Game Theory:
  Concern with the analysis of optimal decision making in competitive situations.
- Strategy:
  A plan for the action that a player in a game will take under.
- There are:
  - players
  - strategies
  - outcomes

Example:

		Toyota	
		Build a new plant	Do not build a new plant
Honda	Build a new plant	16, 16	20, 15
	Do not build a new plant	15, 20	18, 18

Example (continued)

- Outcome:
  1. Honda build a new plant and Toyota build a new plant: 16 for Honda, and 16 for Toyota (16, 16).
  2. Honda build a new plant and Toyota do not build a new plant: 20 for Honda, and 15 for Toyota (20, 15).

The Nash Equilibrium

- Nash Equilibrium:
  a situation in which each player in a game chooses the strategy that yields the highest payoff, given strategies chosen by the other players.

Example (continued)

- Player:
  1. Honda.
  2. Toyota.
- Strategies:
  1. Build a new plant.
  2. Do not build a new plant.

Example (continued)

- Outcome:
  3. Honda do not build a new plant and Toyota build a new plant: 15 for Honda, and 20 for Toyota (15, 20).
  4. Honda do not build a new plant and Toyota do not build a new plant: 18 for Honda, and 18 for Toyota (18, 18).

Nash Equilibrium: for each firm the strategy "build a new plant" was better than "do not build," no matter what strategy the other firm chose.

Three-way clustering with GTRS

- ❖ **An approach based on GTRS, which considers the tradeoff between accuracy and generality and automatically determines the thresholds.**
- ❖ **Game formulization:**

- **Objective = improve the quality of clustering data with missing values = find a tradeoff between accuracy and generality of the clustering**
- **Game: Accuracy VS Generality**
- **2 players: P = {*A*, *G*}. The player accuracy = *A* and player generality = *G***
- **Strategies = modification in thresholds**
- **3 strategies: (1) decrease in threshold *α* (denoted as *α*↓), (2) increase in threshold *β* (denoted as *β*↑), and (3) decrease *α* and increase *β* simultaneously (denoted as *α*↓*β*↑).**
- **Each player chooses a strategy in order to maximize his benefits.**
- **A payoff function is used to measure the results of selecting a certai*(α, β)*n strategy**
- **For a particular strategy profile, say *(s_m, s_n)* that leads to thresholds, the associated payoffs of the players are defined as,**

$$u_A(s_m, s_n) = Accuracy(\alpha, \beta), \qquad Accuracy(\alpha, \beta) = \frac{\text{Correctly clustered objects}}{\text{Total clustered objects}},$$

u_A and u_G are the payoff functions of players *A* and *G*,

$$u_G(s_m, s_n) = Generality(\alpha, \beta), \qquad Generality(\alpha, \beta) = \frac{\text{Total clustered objects}}{\text{Total objects in } U}.$$

For both the players, a value of 1 means maximum payoff and a value of 0 means minimum payoff.

Three-way clustering with GTRS

Algorithm 1 GTRS based threshold learning algorithm.

Input: K as number of clusters, U as a dataset and initial values of $\alpha-$, $\alpha--$, $\beta+$ and $\beta++$.
Output: Three-way clustering of objects.
1: Initialize $\alpha = 1.0$, $\beta = 0.0$.
2: Divide U into C and M. # C is the set of objects with no missing values and M is the set of objects with missing values.
3: Apply K-mean clustering on C.
4: Randomly remove values from C by following the percentage of missing values in M.
5: Divide C into U_c and U_m. # U_c is the set of objects with no missing values and U_m is the set of objects with simulated missing values.
6: **Repeat**
7: Calculate the utilities of players by using Equations (14) and (15).
8: Populate the payoff table with calculated values.
9: Calculate equilibrium in a payoff table by using Equations (19) and (20).
10: Determine selected strategies and corresponding thresholds (α', β').
11: $(\alpha, \beta) = (\alpha', \beta')$.
12: **Until** $Accuracy(\alpha, \beta) \leq Generality(\alpha, \beta)$ or $\alpha \leq 0.5$ or $\beta \geq 0.5$ or Maximum iterations reached.
13: Evaluate objects in M using Equation (9).
14: Use (α, β) determined in Line 11, with three-way framework of Equations (6)–(8) for assigning objects to different regions of a clusters.

List of participants

- houssam.ali@ens.uvsq.fr
- farah.bourrar@ens.uvsq.fr
- imane.ghouzali@ens.uvsq.fr
- amine.laiou@ens.uvsq.fr
- sakhite.mboup@ens.uvsq.fr

Extra slides

Méthode k-moyenne (k-means)

K-moyenne (Forgy 1965, MacQueen 1967)

- Choisir le nombre de clusters et une mesure de distance.
- Construire une partition aléatoire comportant k clusters non vides.
- Répéter
  - Calculer le centre de chaque cluster de la partition.
  - Assigner chaque objet au cluster dont le centre est plus proche (distance).

Jusqu'à ce que la partition soit stable.

Exemple: k-moyenne

T={2, 4, 6, 7, 8, 11, 13}

D = distance Euclidienne

- Choisir k = 3 clusters au hasard à partir de T:

  C_1={2}, M_1=2,

  C_2={4}, M_2=4,

  C_3={6}, M_3=6

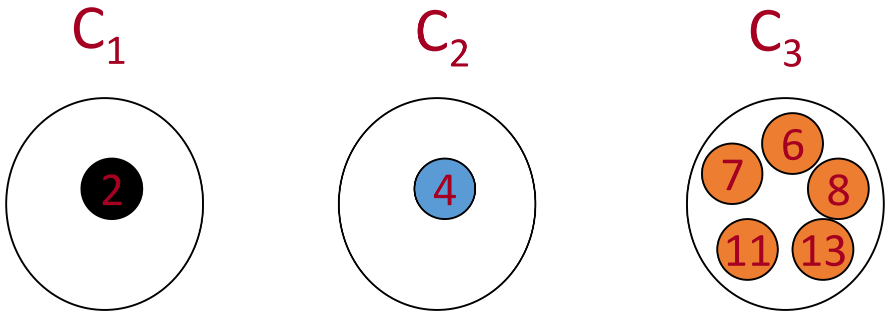
- Les autres objets de T sont affectés au cluster C_3 puisque $D(O,M_3)$ est minimale.

- On aura:

  C_1={2}, M_1=2,

  C_2={4}, M_2=4,

  C_3={6, 7, 8, 11, 13}, M_3= 45/5 = 9



Exemple: k-moyenne

- D(6,M2=4)<D(6,M3=9)

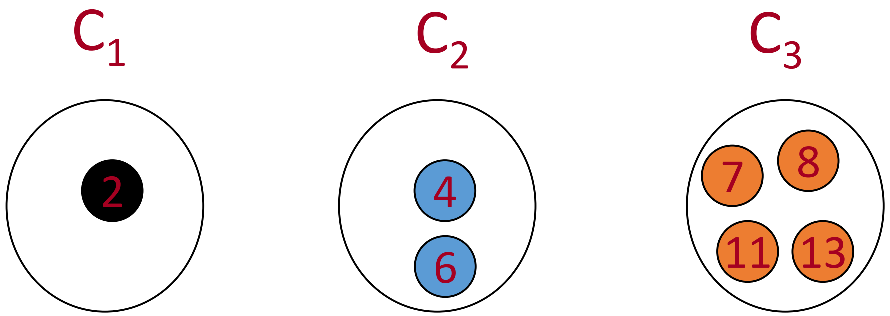
⇨ 6 passe au cluster C_2: les autres objets restent dans leurs clusters.

- On aura:

  C_1={2}, M_1=2,

  C_2={4, 6}, M_2= 10/2 = 5

  C_3={7, 8, 11, 13}, M_3= 39/4 = 9.75



Exemple: k-moyenne

- $D(7,M2)<D(7,M3)$

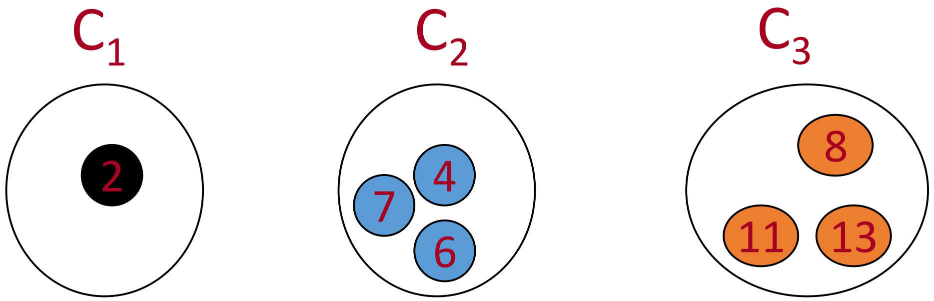
⇨ 7 passe au cluster C_2: les autres objets restent dans leurs clusters.

- 0n aura:

  $C_1=\{2\}$, $M_1=2$,

  $C_2=\{4, 6, 7\}$, $M_2=17/3 = 5.66$,

  $C_3=\{8, 11, 13\}$, $M_3= 32/3 = 10.66$



Exemple: k-moyenne

- D(8,M2)<D(8,M3)

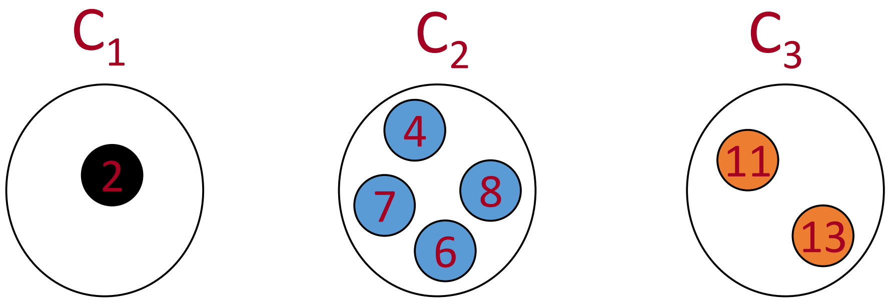
➡ 8 passe au cluster C_2: les autres objets restent dans leurs clusters.

- On aura:

C_1={2}, M_1=2,

C_2={4, 6, 7, 8}, M_2=25/4 = 6.25,

C_3={11, 13}, M_3= 24/2 = 12



Exemple: k-moyenne

- $D(4,M1)<D(4,M2)$

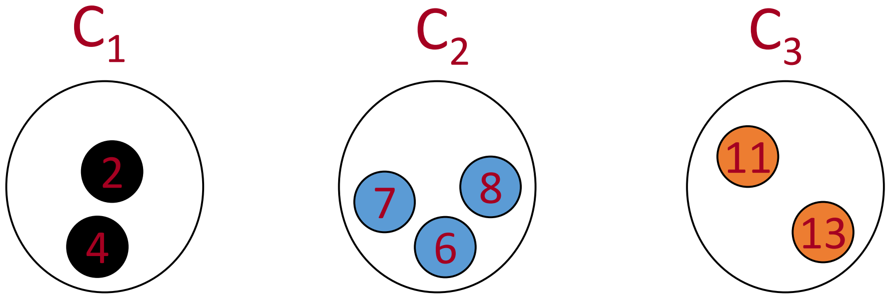
4 passe au cluster C_1: les autres objets restent dans leurs clusters.

- 0n aura:

$C_1=\{2, 4\}$, $M_1=3$,

$C_2=\{6, 7, 8\}$, $M_2=21/3 = 7$,

$C_3=\{11, 13\}$, $M_3= 24/2 = 12$



La partition est stable.

Attention !!

- K-moyenne est appelée aussi méthode des centres mobiles.
- Si on a plusieurs attributs

  ⇨ Nécessité de normaliser les échelles des différents attributs.

- Choix du nombre de classes k dépend de l'utilisateur.
- Résultat dépendant des clusters initiaux choisis.

  ⇨ Faire plusieurs expérimentations avec différents clusters initiaux et choisir la meilleure configuration.

- Plusieurs variantes de K-moyenne:
  - Sélection des k clusters initiaux.
  - Mesure de la distance utilisée.
  - Calcul de la moyenne des clusters.