An Efficient and Effective Generic Agglomerative Hierarchical Clustering Approach [1]

Julien Ah-Pine
Université Lyon 2 - Laboratoire ERIC
(Actuellement en délégation CNRS au LMBP)
julien.ah-pine@univ-lyon2.fr

Séminaire LISC
Clermont-Ferrand
7 juin 2019

1. http://www.jmlr.org/papers/v19/18-117.html

Sommaire

Rappel du Sommaire

Plusieurs façons de classifier

- Problème : étant donné une collection de n objets décrits par p variables, il s'agit de regrouper automatiquement ces objets dans des **classes (ou clusters) homogènes** càd : les objets doivent être similaires s'ils appartiennent à une même classe et dissimilaires s'ils appartiennent à des classes distinctes.

Plusieurs façons de classifier

- Problème : étant donné une collection de n objets décrits par p variables, il s'agit de regrouper automatiquement ces objets dans des **classes (ou clusters) homogènes** càd : les objets doivent être similaires s'ils appartiennent à une même classe et dissimilaires s'ils appartiennent à des classes distinctes.
- Structures de classification :
  - Partition en k classes,
  - Classification hiérarchique (ensemble de partitions emboîtées).

Plusieurs façons de classifier

- Problème : étant donné une collection de n objets décrits par p variables, il s'agit de regrouper automatiquement ces objets dans des **classes (ou clusters) homogènes** càd : les objets doivent être similaires s'ils appartiennent à une même classe et dissimilaires s'ils appartiennent à des classes distinctes.
- Structures de classification :
  - Partition en k classes,
  - Classification hiérarchique (ensemble de partitions emboîtées).
- Plusieurs types d'appartenance d'un objet à un cluster :
  - Fonction caractéristique binaire,
  - Fonction caractéristique floue,
  - Classes recouvrantes,
  - Distribution de probabilité.

Le partitionnement : un problème combinatoire

- Approche énumérative (naïve) :
  - Enumération de toutes les partitions possibles de n objets en k classes,
  - Mesure de la qualité de chaque partition,
  - Sélection de la partition de meilleure qualité.

Le partitionnement : un problème combinatoire

- Approche énumérative (naïve) :
  - Enumération de toutes les partitions possibles de n objets en k classes,
  - Mesure de la qualité de chaque partition,
  - Sélection de la partition de meilleure qualité.
- Problème combinatoire :
  - Nombre de partitions d'un ensemble de n objets en k classes, **nombre de Stirling** de 2nde espèce :

$$S(n,k) = \frac{1}{k!}\sum_{j=0}^{k}(-1)^j \binom{n}{k} j^n$$

  - Nombre total de partitions de n objets, **nombre de Bell** :

$$B(n) = \sum_{k=0}^{n} S(n,k)$$

Un problème combinatoire (suite)

- Quelques valeurs de $S(n,k)$ et $B(n)$:

$n\backslash k$	0	1	2	3	4	5	6	$B(n)$
0	1	0	0	0	0	0	0	1
1	0	1	0	0	0	0	0	1
2	0	1	1	0	0	0	0	2
3	0	1	3	1	0	0	0	5
4	0	1	7	6	1	0	0	15
5	0	1	15	25	10	1	0	52
6	0	1	31	90	65	15	1	203

Un problème combinatoire (suite)

- Quelques valeurs de $S(n,k)$ et $B(n)$:

$n\backslash k$	0	1	2	3	4	5	6	$B(n)$
0	1	0	0	0	0	0	0	1
1	0	1	0	0	0	0	0	1
2	0	1	1	0	0	0	0	2
3	0	1	3	1	0	0	0	5
4	0	1	7	6	1	0	0	15
5	0	1	15	25	10	1	0	52
6	0	1	31	90	65	15	1	203

- Par exemple : $B(71) \simeq 4 \times 10^{74}$.

Un problème combinatoire (suite)

- Quelques valeurs de $S(n,k)$ et $B(n)$:

$n\backslash k$	0	1	2	3	4	5	6	$B(n)$
0	1	0	0	0	0	0	0	1
1	0	1	0	0	0	0	0	1
2	0	1	1	0	0	0	0	2
3	0	1	3	1	0	0	0	5
4	0	1	7	6	1	0	0	15
5	0	1	15	25	10	1	0	52
6	0	1	31	90	65	15	1	203

- Par exemple : $B(71) \simeq 4 \times 10^{74}$.
- Le partitionnement en k classes est un problème NP-dur.

Un problème combinatoire (suite)

- Quelques valeurs de $S(n,k)$ et $B(n)$:

$n\backslash k$	0	1	2	3	4	5	6	$B(n)$
0	1	0	0	0	0	0	0	1
1	0	1	0	0	0	0	0	1
2	0	1	1	0	0	0	0	2
3	0	1	3	1	0	0	0	5
4	0	1	7	6	1	0	0	15
5	0	1	15	25	10	1	0	52
6	0	1	31	90	65	15	1	203

- Par exemple : $B(71) \simeq 4 \times 10^{74}$.
- Le partitionnement en k classes est un problème NP-dur.
- La recherche d'une hiérarchie optimale est encore plus coûteuse.

Quelques méthodes classiques de partitionnement

- On utilise donc des heuristiques (solutions approchées).

Quelques méthodes classiques de partitionnement

- On utilise donc des heuristiques (solutions approchées).
- Pour le “partitionnement dur”, la méthode des k-moyennes :
  - Critère basé sur l'inertie (variance intra-classe).
  - Procédure itérative de ré-allocation des objets vers les classes.

Quelques méthodes classiques de partitionnement

- On utilise donc des heuristiques (solutions approchées).
- Pour le "partitionnement dur", la méthode des k-moyennes :
  - Critère basé sur l'inertie (variance intra-classe).
  - Procédure itérative de ré-allocation des objets vers les classes.
- Pour le "partitionnement probabiliste", le modèle de mélange :
  - Modèle paramétrique : $f(\mathbf{x}|\boldsymbol{\pi}, \boldsymbol{\theta}) = \sum_{l=1}^{k} \pi_l g_l(\mathbf{x}|\theta_l)$.
  - Estimation des paramètres par l'algorithme EM.

Quelques méthodes classiques de partitionnement

- On utilise donc des heuristiques (solutions approchées).
- Pour le "partitionnement dur", la méthode des k-moyennes :
  - Critère basé sur l'inertie (variance intra-classe).
  - Procédure itérative de ré-allocation des objets vers les classes.
- Pour le "partitionnement probabiliste", le modèle de mélange :
  - Modèle paramétrique : $f(\mathbf{x}|\boldsymbol{\pi}, \boldsymbol{\theta}) = \sum_{l=1}^{k} \pi_l g_l(\mathbf{x}|\theta_l)$.
  - Estimation des paramètres par l'algorithme EM.
- Pour le "partitionnement flou", la méthode des fuzzy c-means :
  - Extension du modèle des k-moyennes à des matrices d'affectation à valeurs dans $[0, 1]$ au lieu de $\{0, 1\}$.
  - Procédure itérative d'optimisation alternée avec solutions analytiques.

Quelques méthodes classiques de partitionnement

- On utilise donc des heuristiques (solutions approchées).
- Pour le "partitionnement dur", la méthode des k-moyennes :
  - Critère basé sur l'inertie (variance intra-classe).
  - Procédure itérative de ré-allocation des objets vers les classes.
- Pour le "partitionnement probabiliste", le modèle de mélange :
  - Modèle paramétrique : $f(\mathbf{x}|\boldsymbol{\pi}, \boldsymbol{\theta}) = \sum_{l=1}^{k} \pi_l g_l(\mathbf{x}|\theta_l)$.
  - Estimation des paramètres par l'algorithme EM.
- Pour le "partitionnement flou", la méthode des fuzzy c-means :
  - Extension du modèle des k-moyennes à des matrices d'affectation à valeurs dans $[0, 1]$ au lieu de $\{0, 1\}$.
  - Procédure itérative d'optimisation alternée avec solutions analytiques.

▷ Avantages : complexité en $O(n)$, convergence.

Quelques méthodes classiques de partitionnement

- On utilise donc des heuristiques (solutions approchées).
- Pour le "partitionnement dur", la méthode des k-moyennes :
  - Critère basé sur l'inertie (variance intra-classe).
  - Procédure itérative de ré-allocation des objets vers les classes.
- Pour le "partitionnement probabiliste", le modèle de mélange :
  - Modèle paramétrique : $f(\mathbf{x}|\boldsymbol{\pi}, \boldsymbol{\theta}) = \sum_{l=1}^{k} \pi_l g_l(\mathbf{x}|\theta_l)$.
  - Estimation des paramètres par l'algorithme EM.
- Pour le "partitionnement flou", la méthode des fuzzy c-means :
  - Extension du modèle des k-moyennes à des matrices d'affectation à valeurs dans $[0, 1]$ au lieu de $\{0, 1\}$.
  - Procédure itérative d'optimisation alternée avec solutions analytiques.

▷ Avantages : complexité en $O(n)$, convergence.

▷ Inconvénients : classes de forme ellipsoïdale, données numériques principalement, nombre de classes à fixer, sensibles à l'initialisation.

Rappel du Sommaire

Ascendante *versus* Descendante

- Dans ce cas, on construit un **arbre binaire de classification**.

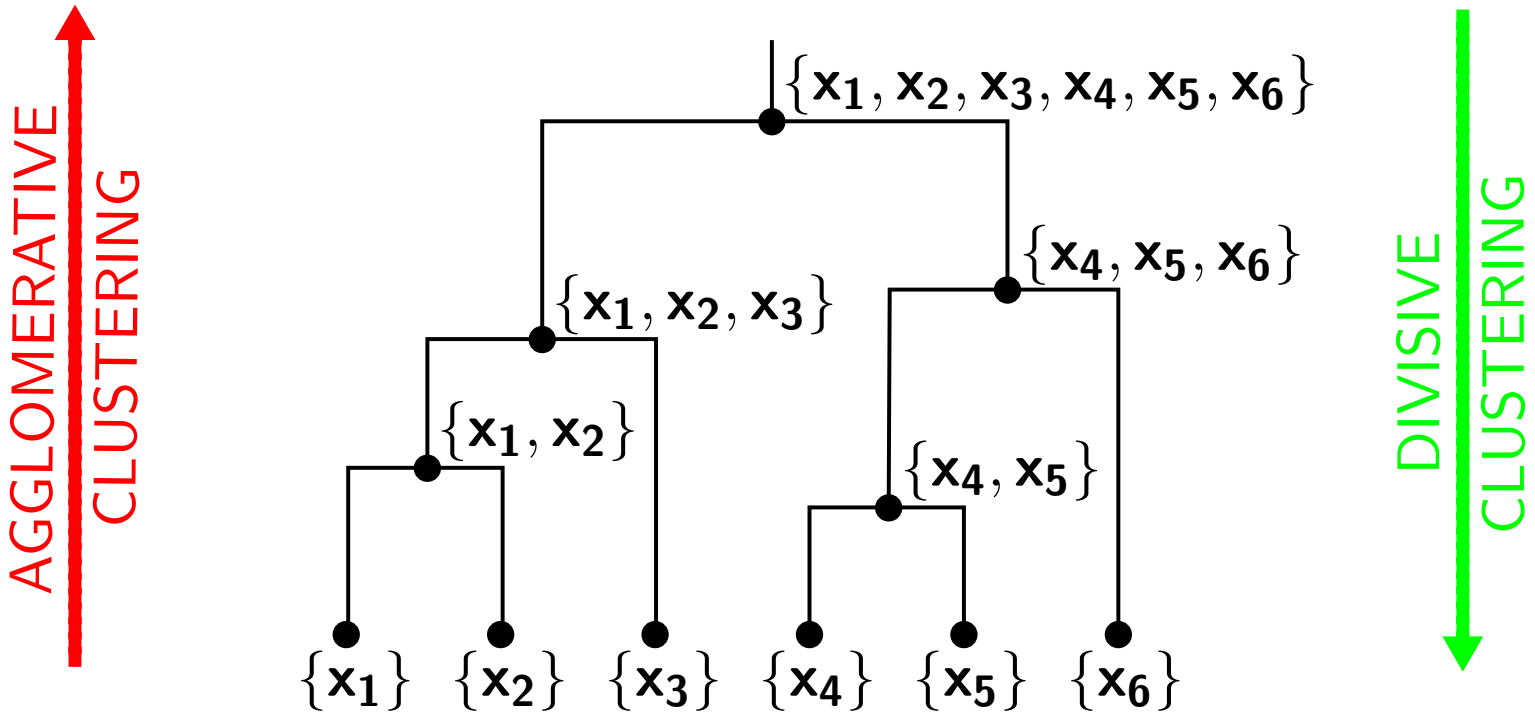
Ascendante *versus* Descendante

- Dans ce cas, on construit un **arbre binaire de classification**.
- Deux procédures :
  - **Ascendante** : on part de n singletons (feuilles de l'arbre) et on regroupe itérativement deux classes à la fois jusqu'à ce que tout les objets soient dans la même classe (racine de l'arbre).
  - **Descendante** : on part de la partition à une classe et on scinde (en deux généralement) les classes jusqu'à obtenir des singletons.

Ascendante *versus* Descendante

- Dans ce cas, on construit un **arbre binaire de classification**.
- Deux procédures :
  - **Ascendante** : on part de n singletons (feuilles de l'arbre) et on regroupe itérativement deux classes à la fois jusqu'à ce que tout les objets soient dans la même classe (racine de l'arbre).
  - **Descendante** : on part de la partition à une classe et on scinde (en deux généralement) les classes jusqu'à obtenir des singletons.
- Dans les deux cas il s'agit d'une **stratégie gloutonne** : à chaque itération on ne remet pas en cause les décisions précédentes.
- Algorithmes optimaux localement mais sous-optimaux globalement.

Ascendante *versus* Descendante (suite)

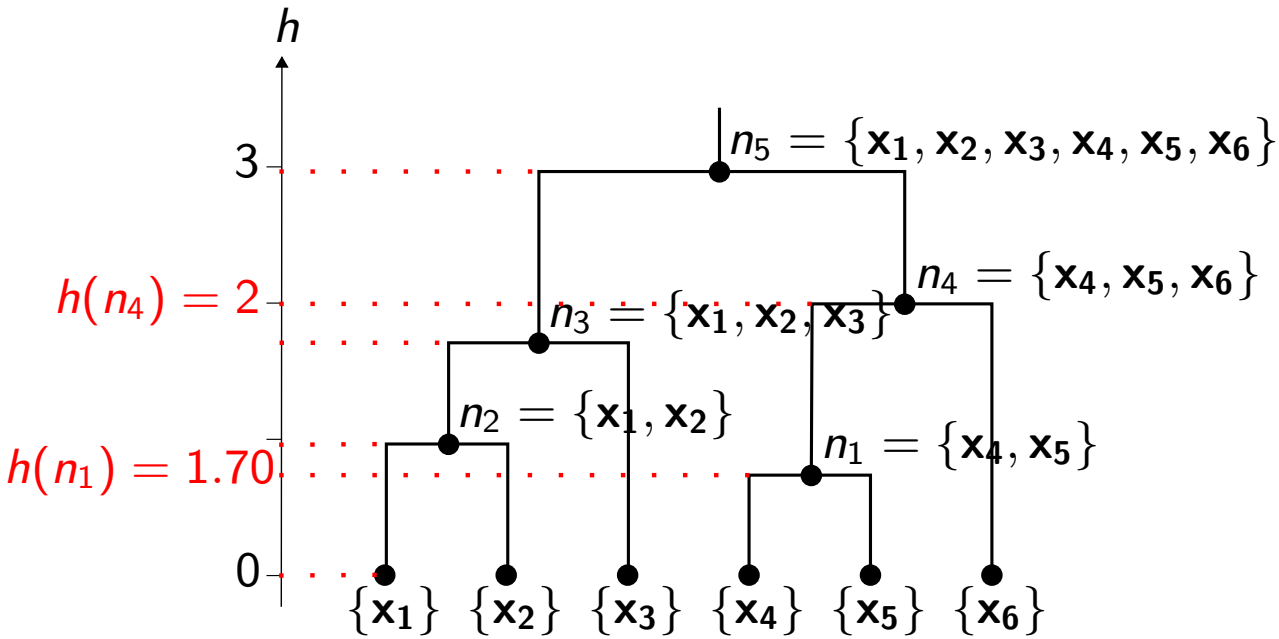


Dendrogramme

- Le résultat d'une classification hiérarchique est appelé **dendrogramme** : **arbre binaire** auquel on ajoute à chaque noeud n un indice $h(n)$ appelé la **hauteur** et un sous-ensemle d'objets (**cluster**). Dans l'approche classique, $h(n)$ est égale à la valeur de la dissimilarité lorsque le cluster de n est créé.

Dendrogramme

- Le résultat d'une classification hiérarchique est appelé **dendrogramme** : **arbre binaire** auquel on ajoute à chaque noeud n un indice $h(n)$ appelé la **hauteur** et un sous-ensemle d'objets (**cluster**). Dans l'approche classique, $h(n)$ est égale à la valeur de la dissimilarité lorsque le cluster de n est créé.
- Illustration :



AHC basée sur la matrice de dissimilarités

- Nous étudions la classification **ascendante** hiérarchique (AHC).

AHC basée sur la matrice de dissimilarités

- Nous étudions la classification **ascendante** hiérarchique (AHC).
- L'approche classique est basée sur la matrice de **dissimilarités** entre objets, **D** : Dissimilarity based AHC (**D-AHC**).

AHC basée sur la matrice de dissimilarités

- Nous étudions la classification **ascendante** hiérarchique (AHC).
- L'approche classique est basée sur la matrice de **dissimilarités** entre objets, **D** : Dissimilarity based AHC (**D-AHC**).
- **D** vérifie les axiomes suivantes : $\forall a, b \in \mathbb{O}$,

$$\begin{cases} \mathbf{D}_{ab} \geq 0 \text{ (non négativité)} \\ \mathbf{D}_{ab} = \mathbf{D}_{ba} \text{ (symétrie)} \\ \mathbf{D}_{aa} = 0 \text{ (réflexivité)} \end{cases}$$

AHC basée sur la matrice de dissimilarités

- Nous étudions la classification **ascendante** hiérarchique (AHC).
- L'approche classique est basée sur la matrice de **dissimilarités** entre objets, **D** : Dissimilarity based AHC (**D-AHC**).
- **D** vérifie les axiomes suivantes : $\forall a, b \in \mathbb{O}$,

$$\begin{cases} \mathbf{D}_{ab} \geq 0 \text{ (non négativité)} \\ \mathbf{D}_{ab} = \mathbf{D}_{ba} \text{ (symétrie)} \\ \mathbf{D}_{aa} = 0 \text{ (réflexivité)} \end{cases}$$

- Le pseudo-code est le suivant :

Input: **D** (dissimiarity matrix), dissimilarity measure
Output: D a dendrogram

```
1 Initialize D with n leaves;
2 for t = 1, ..., n − 1 do
3 |   Find the pair of clusters (k, l) that are the closest;
4 |   Merge (k, l) into (kl) and update D;
5 |   Compute the dissimilarity measure between (kl) and other clusters.
```

Plusieurs mesures de dissimilarités

- 7 mesures de dissimilarité classiques.

Plusieurs mesures de dissimilarités

- 7 mesures de dissimilarité classiques.
- Méthodes dites basées sur les graphes :
  - Single linkage
  - Complete linkage
  - Group average
  - Mcquitty

Plusieurs mesures de dissimilarités

- 7 mesures de dissimilarité classiques.
- Méthodes dites basées sur les graphes :
  - Single linkage
  - Complete linkage
  - Group average
  - Mcquitty
- Méthodes dites géométriques :
  - Centroid
  - Median
  - Ward

La formule de Lance-Williams

- $\mathbb{O}$ est l'ensemble des objets à classifier, $|\mathbb{O}| = n$.

La formule de Lance-Williams

- $\mathbb{O}$ est l'ensemble des objets à classifier, $|\mathbb{O}| = n$.
- $t \in \mathbb{T}$ (où $\mathbb{T} = \{1, \ldots, n-1\}$) est l'indice spécifiant les itérations.

La formule de Lance-Williams

- $\mathbb{O}$ est l'ensemble des objets à classifier, $|\mathbb{O}| = n$.
- $t \in \mathbb{T}$ (où $\mathbb{T} = \{1, \ldots, n-1\}$) est l'indice spécifiant les itérations.
- $\mathbb{C}^t$ est l'ensemble des classes à l'itération t.

La formule de Lance-Williams

- $\mathbb{O}$ est l'ensemble des objets à classifier, $|\mathbb{O}| = n$.
- $t \in \mathbb{T}$ (où $\mathbb{T} = \{1, \ldots, n-1\}$) est l'indice spécifiant les itérations.
- $\mathbb{C}^t$ est l'ensemble des classes à l'itération t.
- a et b désignent deux singletons ; k, l, m des classes quelconques.

La formule de Lance-Williams

- $\mathbb{O}$ est l'ensemble des objets à classifier, $|\mathbb{O}| = n$.
- $t \in \mathbb{T}$ (où $\mathbb{T} = \{1, \ldots, n-1\}$) est l'indice spécifiant les itérations.
- $\mathbb{C}^t$ est l'ensemble des classes à l'itération t.
- a et b désignent deux singletons ; k, l, m des classes quelconques.
- AHC basée sur la matrice de dissimilarités et la formule LW :

La formule de Lance-Williams

- $\mathbb{O}$ est l'ensemble des objets à classifier, $|\mathbb{O}| = n$.
- $t \in \mathbb{T}$ (où $\mathbb{T} = \{1, \ldots, n-1\}$) est l'indice spécifiant les itérations.
- $\mathbb{C}^t$ est l'ensemble des classes à l'itération t.
- a et b désignent deux singletons ; k, l, m des classes quelconques.
- AHC basée sur la matrice de dissimilarités et la formule LW :
  - Initialisation $\mathbf{D}^1 = \mathbf{D}$, la matrice de dissimilarités entre les n objets.

La formule de Lance-Williams

- $\mathbb{O}$ est l'ensemble des objets à classifier, $|\mathbb{O}| = n$.
- $t \in \mathbb{T}$ (où $\mathbb{T} = \{1, \ldots, n-1\}$) est l'indice spécifiant les itérations.
- $\mathbb{C}^t$ est l'ensemble des classes à l'itération t.
- a et b désignent deux singletons ; k, l, m des classes quelconques.
- AHC basée sur la matrice de dissimilarités et la formule LW :
  - Initialisation $\mathbf{D}^1 = \mathbf{D}$, la matrice de dissimilarités entre les n objets.
  - A chaque t, on fusionne la paire (k, l) qui **minimise la dissimilarité** :

$$(k, l) = \underset{(i,j) \in \mathbb{C}^t \times \mathbb{C}^t, i \neq j}{\arg\min} \mathbf{D}^t_{ij}$$

La formule de Lance-Williams

- $\mathbb{O}$ est l'ensemble des objets à classifier, $|\mathbb{O}| = n$.
- $t \in \mathbb{T}$ (où $\mathbb{T} = \{1, \ldots, n-1\}$) est l'indice spécifiant les itérations.
- $\mathbb{C}^t$ est l'ensemble des classes à l'itération t.
- a et b désignent deux singletons ; k, l, m des classes quelconques.
- AHC basée sur la matrice de dissimilarités et la formule LW :
  - Initialisation $\mathbf{D}^1 = \mathbf{D}$, la matrice de dissimilarités entre les n objets.
  - A chaque t, on fusionne la paire (k, l) qui **minimise la dissimilarité** :

$$(k, l) = \underset{(i,j) \in \mathbb{C}^t \times \mathbb{C}^t, i \neq j}{\arg\min} \mathbf{D}^t_{ij}$$

  - k et l sont regroupées en (kl) et on calcule la dissimilarité entre (kl) et les autres classes, $m \in \mathbb{C}^{t+1}, m \neq (kl)$, comme suit :

La formule de Lance-Williams

- $\mathbb{O}$ est l'ensemble des objets à classifier, $|\mathbb{O}| = n$.
- $t \in \mathbb{T}$ (où $\mathbb{T} = \{1, \ldots, n-1\}$) est l'indice spécifiant les itérations.
- $\mathbb{C}^t$ est l'ensemble des classes à l'itération t.
- a et b désignent deux singletons ; k, l, m des classes quelconques.
- AHC basée sur la matrice de dissimilarités et la formule LW :
  - Initialisation $\mathbf{D}^1 = \mathbf{D}$, la matrice de dissimilarités entre les n objets.
  - A chaque t, on fusionne la paire (k, l) qui **minimise la dissimilarité** :

$$(k, l) = \underset{(i,j) \in \mathbb{C}^t \times \mathbb{C}^t, i \neq j}{\arg\min} \mathbf{D}^t_{ij}$$

  - k et l sont regroupées en (kl) et on calcule la dissimilarité entre (kl) et les autres classes, $m \in \mathbb{C}^{t+1}, m \neq (kl)$, comme suit :

$$\begin{aligned}\mathbf{D}^{t+1}_{(kl)m} =& \alpha(k, l, m)\mathbf{D}^t_{km} + \alpha(l, k, m)\mathbf{D}^t_{lm} + \beta(k, l, m)\mathbf{D}^t_{kl} \\ &+ \gamma|\mathbf{D}^t_{km} - \mathbf{D}^t_{lm}|\end{aligned}$$

où $\gamma \in \mathbb{R}$ et α, β sont des **fonctions d'ensemble** à valeurs dans $\mathbb{R}$.

Paramètre des méthodes dans le cadre de la formule de LW

Method	$\alpha(k,l,m)$	$\beta(k,l,m)$	γ
Single link.	$1/2$	0	$-1/2$
Complete link.	$1/2$	0	$1/2$
Group aver.	$\frac{\lvert k\rvert}{\lvert k\rvert+\lvert l\rvert}$	0	0
Mcquitty	$1/2$	0	0
Centroid	$\frac{\lvert k\rvert}{\lvert k\rvert+\lvert l\rvert}$	$-\frac{\lvert k\rvert\lvert l\rvert}{(\lvert k\rvert+\lvert l\rvert)^2}$	0
Median	$1/2$	$-1/4$	0
Ward	$\frac{\lvert k\rvert+\lvert m\rvert}{\lvert k\rvert+\lvert l\rvert+\lvert m\rvert}$	$-\frac{\lvert m\rvert}{\lvert k\rvert+\lvert l\rvert+\lvert m\rvert}$	0

Une version équivalente de la formule de LW

- Nous traitons que les cas $\gamma = 0$. On exclut donc single et complete linkage. Ces cas se réduisent aux opérations min et max et sont traités plus efficacement par des algo. de graphe (minimum spanning tree).

Une version équivalente de la formule de LW

- Nous traitons que les cas $\gamma = 0$. On exclut donc single et complete linkage. Ces cas se réduisent aux opérations min et max et sont traités plus efficacement par des algo. de graphe (minimum spanning tree).
- Nous utilisons une version équivalente de la formule de LW :

Une version équivalente de la formule de LW

- Nous traitons que les cas $\gamma = 0$. On exclut donc single et complete linkage. Ces cas se réduisent aux opérations min et max et sont traités plus efficacement par des algo. de graphe (minimum spanning tree).
- Nous utilisons une version équivalente de la formule de LW :
  - A chaque itération t, on fusionne la paire (k, l) qui **minimise la dissimilarité pondérée** suivante :

$$\boxed{(k,l) = \underset{(i,j)\in\mathbb{C}^t\times\mathbb{C}^t, i\neq j}{\arg\min} p(i,j)\mathbf{D}_{ij}^t} \quad (1)$$

où p est une fonction d'ensemble (cf tableau qui suit).

Une version équivalente de la formule de LW

- Nous traitons que les cas $\gamma = 0$. On exclut donc single et complete linkage. Ces cas se réduisent aux opérations min et max et sont traités plus efficacement par des algo. de graphe (minimum spanning tree).
- Nous utilisons une version équivalente de la formule de LW :
  - A chaque itération t, on fusionne la paire (k, l) qui **minimise la dissimilarité pondérée** suivante :

$$\boxed{(k,l) = \underset{(i,j)\in\mathbb{C}^t\times\mathbb{C}^t, i\neq j}{\arg\min} p(i,j)\mathbf{D}^t_{ij}} \quad (1)$$

où p est une fonction d'ensemble (cf tableau qui suit).

  - Pour calculer la dissimilarité entre (kl) et $m \in \mathbb{C}^{t+1}, m \neq (kl)$:

$$\boxed{\mathbf{D}^{t+1}_{(kl)m} = \alpha(k,l)\mathbf{D}^t_{km} + \alpha(l,k)\mathbf{D}^t_{lm} + \beta(k,l)\mathbf{D}^t_{kl}} \quad (2)$$

où α et β sont ici des fonctions d'ensemble de **deux arguments**.

Paramètre des méthodes dans le cadre de la 2ème formule de LW

Method	$\alpha(k,l)$	$\beta(k,l)$	$p(i,j)$
Group aver.	$\frac{\lvert k\rvert}{\lvert k\rvert+\lvert l\rvert}$	0	1
Mcquitty	$1/2$	0	1
Centroid	$\frac{\lvert k\rvert}{\lvert k\rvert+\lvert l\rvert}$	$-\frac{\lvert k\rvert\lvert l\rvert}{(\lvert k\rvert+\lvert l\rvert)^2}$	1
Median	$1/2$	$-1/4$	1
Ward	$\frac{\lvert k\rvert}{\lvert k\rvert+\lvert l\rvert}$	$-\frac{\lvert k\rvert\lvert l\rvert}{(\lvert k\rvert+\lvert l\rvert)^2}$	$\frac{\lvert i\rvert\lvert j\rvert}{\lvert i\rvert+\lvert j\rvert}$
W-Median	$1/2$	$-1/4$	$\frac{\lvert i\rvert\lvert j\rvert}{\lvert i\rvert+\lvert j\rvert}$

Paramètre des méthodes dans le cadre de la 2ème formule de LW

Method	$\alpha(k,l)$	$\beta(k,l)$	$p(i,j)$
Group aver.	$\frac{\lvert k\rvert}{\lvert k\rvert+\lvert l\rvert}$	0	1
Mcquitty	$1/2$	0	1
Centroid	$\frac{\lvert k\rvert}{\lvert k\rvert+\lvert l\rvert}$	$-\frac{\lvert k\rvert\lvert l\rvert}{(\lvert k\rvert+\lvert l\rvert)^2}$	1
Median	$1/2$	$-1/4$	1
Ward	$\frac{\lvert k\rvert}{\lvert k\rvert+\lvert l\rvert}$	$-\frac{\lvert k\rvert\lvert l\rvert}{(\lvert k\rvert+\lvert l\rvert)^2}$	$\frac{\lvert i\rvert\lvert j\rvert}{\lvert i\rvert+\lvert j\rvert}$
W-Median	$1/2$	$-1/4$	$\frac{\lvert i\rvert\lvert j\rvert}{\lvert i\rvert+\lvert j\rvert}$

- Remarque : la méthode de Ward peut être vue telle une version pondérée de la méthode centroid.

Paramètre des méthodes dans le cadre de la 2ème formule de LW

Method	$\alpha(k,l)$	$\beta(k,l)$	$p(i,j)$
Group aver.	$\frac{\lvert k\rvert}{\lvert k\rvert+\lvert l\rvert}$	0	1
Mcquitty	$1/2$	0	1
Centroid	$\frac{\lvert k\rvert}{\lvert k\rvert+\lvert l\rvert}$	$-\frac{\lvert k\rvert\lvert l\rvert}{(\lvert k\rvert+\lvert l\rvert)^2}$	1
Median	$1/2$	$-1/4$	1
Ward	$\frac{\lvert k\rvert}{\lvert k\rvert+\lvert l\rvert}$	$-\frac{\lvert k\rvert\lvert l\rvert}{(\lvert k\rvert+\lvert l\rvert)^2}$	$\frac{\lvert i\rvert\lvert j\rvert}{\lvert i\rvert+\lvert j\rvert}$
W-Median	$1/2$	$-1/4$	$\frac{\lvert i\rvert\lvert j\rvert}{\lvert i\rvert+\lvert j\rvert}$

- Remarque : la méthode de Ward peut être vue telle une version pondérée de la méthode centroid.
- Remarque : la méthode w-median que nous proposons est une version pondérée de la méthode median.

Synthèse et perspectives

- La procédure “bottom-up” + formule de LW forment l’approche usuelle de la D-AHC.

Synthèse et perspectives

- La procédure "bottom-up" + formule de LW forment l'approche usuelle de la D-AHC.
- ▷ Avantages : plus informatif qu'une partition, plus flexible car plusieurs types de dissimilarité, on peut revenir à une partition en k classes a posteriori (pas de fixation du nombre de classes a priori).

Synthèse et perspectives

- La procédure "bottom-up" + formule de LW forment l'approche usuelle de la D-AHC.
- ▷ Avantages : plus informatif qu'une partition, plus flexible car plusieurs types de dissimilarité, on peut revenir à une partition en k classes a posteriori (pas de fixation du nombre de classes a priori).
- Donc approche très utilisée en pratique et implémentée dans de nombreux languages.

Synthèse et perspectives

- La procédure "bottom-up" + formule de LW forment l'approche usuelle de la D-AHC.
- ▷ Avantages : plus informatif qu'une partition, plus flexible car plusieurs types de dissimilarité, on peut revenir à une partition en k classes a posteriori (pas de fixation du nombre de classes a priori).
- Donc approche très utilisée en pratique et implémentée dans de nombreux languages.
- ▷ Inconvénients : complexité en mémoire en $O(n^2)$ et en temps de traitement en $O(n^3)$, approche gloutonne.

Synthèse et perspectives

- La procédure "bottom-up" + formule de LW forment l'approche usuelle de la D-AHC.

▷ Avantages : plus informatif qu'une partition, plus flexible car plusieurs types de dissimilarité, on peut revenir à une partition en k classes a posteriori (pas de fixation du nombre de classes a priori).

- Donc approche très utilisée en pratique et implémentée dans de nombreux languages.

▷ Inconvénients : complexité en mémoire en $O(n^2)$ et en temps de traitement en $O(n^3)$, approche gloutonne.

▷ On propose des extensions de la AHC basée sur les **noyaux** (K-AHC) et les matrices de **similarités creuses** (SNK-AHC). Cette dernière permet d'**améliorer la complexité** mais aussi **la qualité** du clustering en tenant compte de la **géométrie intrinsèque des données** (variétés non linéaires).

Rappel du Sommaire

Représentation géométrique des objets

- **Hypothèse géométrique 1** : les objets $(a, b, \ldots)$ sont représentés par des vecteurs $(\mathbf{x}^a, \mathbf{x}^b, \ldots)$ dans un espace de Hilbert $\mathcal{H}$ équipé du produit scalaire $\langle ., . \rangle$. On introduit $\mathbf{S}$, la **matrice de Gram**. On suppose alors : $\forall a, b \in \mathbb{O}$,

$$\begin{cases} \mathbf{S}_{ab} = \langle \mathbf{x}^a, \mathbf{x}^b \rangle \\ \mathbf{D}_{ab} = \mathbf{S}_{aa} + \mathbf{S}_{bb} - 2\mathbf{S}_{ab} \end{cases} \tag{C1}$$

  $\mathbf{D}$ est donc la **matrice de distances au carré**.

Représentation géométrique des objets

- **Hypothèse géométrique 1** : les objets $(a, b, \ldots)$ sont représentés par des vecteurs $(\mathbf{x}^a, \mathbf{x}^b, \ldots)$ dans un espace de Hilbert $\mathcal{H}$ équipé du produit scalaire $\langle ., . \rangle$. On introduit $\mathbf{S}$, la **matrice de Gram**. On suppose alors : $\forall a, b \in \mathbb{O}$,

$$\begin{cases} \mathbf{S}_{ab} = \langle \mathbf{x}^a, \mathbf{x}^b \rangle \\ \mathbf{D}_{ab} = \mathbf{S}_{aa} + \mathbf{S}_{bb} - 2\mathbf{S}_{ab} \end{cases} \tag{C1}$$

  $\mathbf{D}$ est donc la **matrice de distances au carré**.
- Ceci implique les propriétés sur $\mathbf{S}$ suivantes :

$$\begin{cases} \mathbf{S}_{ab} = \mathbf{S}_{ba}, \quad \forall a, b \in \mathbb{O} & \text{(symétrie)} \\ \mathbf{S} \succeq 0 & \text{(semi-définie positive)} \end{cases}$$

Représentation géométrique des objets

- **Hypothèse géométrique 1** : les objets $(a, b, \ldots)$ sont représentés par des vecteurs $(\mathbf{x}^a, \mathbf{x}^b, \ldots)$ dans un espace de Hilbert $\mathcal{H}$ équipé du produit scalaire $\langle ., . \rangle$. On introduit $\mathbf{S}$, la **matrice de Gram**. On suppose alors : $\forall a, b \in \mathbb{O}$,

$$\begin{cases} \mathbf{S}_{ab} = \langle \mathbf{x}^a, \mathbf{x}^b \rangle \\ \mathbf{D}_{ab} = \mathbf{S}_{aa} + \mathbf{S}_{bb} - 2\mathbf{S}_{ab} \end{cases} \tag{C1}$$

  $\mathbf{D}$ est donc la **matrice de distances au carré**.
- Ceci implique les propriétés sur $\mathbf{S}$ suivantes :

$$\begin{cases} \mathbf{S}_{ab} = \mathbf{S}_{ba}, \quad \forall a, b \in \mathbb{O} & \text{(symétrie)} \\ \mathbf{S} \succeq 0 & \text{(semi-définie positive)} \end{cases}$$

- En particulier, nous pouvons avoir $\mathbf{S}_{ab} = \langle \phi(\mathbf{x}^a), \phi(\mathbf{x}^b) \rangle = K(\mathbf{x}^a, \mathbf{x}^b)$ où $\phi : \mathcal{H} \to \mathcal{H}'$ et K est la **fonction noyau (RKHS)** correspondante.

Représentation géométrique des objets

- **Hypothèse géométrique 1** : les objets $(a, b, \ldots)$ sont représentés par des vecteurs $(\mathbf{x}^a, \mathbf{x}^b, \ldots)$ dans un espace de Hilbert $\mathcal{H}$ équipé du produit scalaire $\langle ., . \rangle$. On introduit $\mathbf{S}$, la **matrice de Gram**. On suppose alors : $\forall a, b \in \mathbb{O}$,

$$\begin{cases} \mathbf{S}_{ab} = \langle \mathbf{x}^a, \mathbf{x}^b \rangle \\ \mathbf{D}_{ab} = \mathbf{S}_{aa} + \mathbf{S}_{bb} - 2\mathbf{S}_{ab} \end{cases} \tag{C1}$$

  $\mathbf{D}$ est donc la **matrice de distances au carré**.
- Ceci implique les propriétés sur $\mathbf{S}$ suivantes :

$$\begin{cases} \mathbf{S}_{ab} = \mathbf{S}_{ba}, \quad \forall a, b \in \mathbb{O} & \text{(symétrie)} \\ \mathbf{S} \succeq 0 & \text{(semi-définie positive)} \end{cases}$$

- En particulier, nous pouvons avoir $\mathbf{S}_{ab} = \langle \phi(\mathbf{x}^a), \phi(\mathbf{x}^b) \rangle = K(\mathbf{x}^a, \mathbf{x}^b)$ où $\phi : \mathcal{H} \to \mathcal{H}'$ et K est la **fonction noyau (RKHS)** correspondante.
- ▷ On définit un système équivalent à la formule de LW, qui utilise et met à jour successivement $\mathbf{S}$ la **matrice des noyaux** au lieu de $\mathbf{D}$.

Expression de la formule de LW en termes de noyaux

- Etant donné $\mathbf{S}^t$, $\mathbf{\Lambda}^t$ est la matrice définie par : $\forall i, j \in \mathbb{C}^t$,

$$\boxed{\mathbf{\Lambda}^t_{ij} = \mathbf{S}^t_{ij} - \frac{1}{2}(\mathbf{S}^t_{ii} + \mathbf{S}^t_{jj})} \tag{3}$$

Expression de la formule de LW en termes de noyaux

- Etant donné $\mathbf{S}^t$, $\mathbf{\Lambda}^t$ est la matrice définie par : $\forall i, j \in \mathbb{C}^t$,

$$\boxed{\mathbf{\Lambda}^t_{ij} = \mathbf{S}^t_{ij} - \frac{1}{2}(\mathbf{S}^t_{ii} + \mathbf{S}^t_{jj})} \tag{3}$$

- Notre approche Kernel based AHC (**K-AHC**) est définie comme suit :

Expression de la formule de LW en termes de noyaux

- Etant donné $\mathbf{S}^t$, $\mathbf{\Lambda}^t$ est la matrice définie par : $\forall i, j \in \mathbb{C}^t$,

$$\boxed{\mathbf{\Lambda}^t_{ij} = \mathbf{S}^t_{ij} - \frac{1}{2}(\mathbf{S}^t_{ii} + \mathbf{S}^t_{jj})} \tag{3}$$

- Notre approche Kernel based AHC (**K-AHC**) est définie comme suit :
  - Initialisation $\mathbf{S}^1 = \mathbf{S}$, la matrice de noyaux entre les n objets.

Expression de la formule de LW en termes de noyaux

- Etant donné $\mathbf{S}^t$, $\mathbf{\Lambda}^t$ est la matrice définie par : $\forall i, j \in \mathbb{C}^t$,

$$\boxed{\mathbf{\Lambda}^t_{ij} = \mathbf{S}^t_{ij} - \frac{1}{2}(\mathbf{S}^t_{ii} + \mathbf{S}^t_{jj})} \tag{3}$$

- Notre approche Kernel based AHC (**K-AHC**) est définie comme suit :
  - Initialisation $\mathbf{S}^1 = \mathbf{S}$, la matrice de noyaux entre les n objets.
  - A chaque t, on fusionne la paire (k, l) qui **maximise** :

$$\boxed{(k, l) = \underset{(i,j)\in\mathbb{C}^t\times\mathbb{C}^t, i\neq j}{\arg\max} \mathfrak{p}(i, j)\mathbf{\Lambda}^t_{ij}} \tag{4}$$

Expression de la formule de LW en termes de noyaux

- Etant donné $\mathbf{S}^t$, $\mathbf{\Lambda}^t$ est la matrice définie par : $\forall i, j \in \mathbb{C}^t$,

$$\boxed{\mathbf{\Lambda}^t_{ij} = \mathbf{S}^t_{ij} - \frac{1}{2}(\mathbf{S}^t_{ii} + \mathbf{S}^t_{jj})} \tag{3}$$

- Notre approche Kernel based AHC (**K-AHC**) est définie comme suit :
  - Initialisation $\mathbf{S}^1 = \mathbf{S}$, la matrice de noyaux entre les n objets.
  - A chaque t, on fusionne la paire (k, l) qui **maximise** :

$$\boxed{(k, l) = \underset{(i,j)\in\mathbb{C}^t\times\mathbb{C}^t, i\neq j}{\arg\max} \mathfrak{p}(i, j)\mathbf{\Lambda}^t_{ij}} \tag{4}$$

  - k et l sont regroupées en (kl) et on calcule la similarité entre (kl) et les autres classes, $m \in \mathbb{C}^{t+1}$ et elle même par :

$$\boxed{\begin{aligned} \mathbf{S}^{t+1}_{(kl)m} &= \mathfrak{a}(k, l)\mathbf{S}^t_{km} + \mathfrak{a}(l, k)\mathbf{S}^t_{lm} && (5)\\ \mathbf{S}^{t+1}_{(kl)(kl)} &= \mathfrak{b}(k, l)\mathbf{S}^t_{kl} + \mathfrak{c}(k, l)\mathbf{S}^t_{kk} + \mathfrak{c}(l, k)\mathbf{S}^t_{ll} && (6)\end{aligned}}$$

où $\mathfrak{p}, \mathfrak{a}, \mathfrak{b}, \mathfrak{c}$ sont des fonctions d'ensemble.

Paramètres des methodes dans le cadre de notre approche

- 2 formules de mise à jour de **S** sont nécessaires.

Paramètres des methodes dans le cadre de notre approche

- 2 formules de mise à jour de **S** sont nécessaires.
- Les fonctions d'ensemble sont définies comme suit :

$$\mathfrak{a} = \alpha;\ \mathfrak{b} = -2\beta;\ \mathfrak{c} = \alpha + \beta;\ \mathfrak{p} = p;$$

α, β, p étant les fonctions d'ensemble intervenant dans (1) et (2).

Paramètres des methodes dans le cadre de notre approche

- 2 formules de mise à jour de **S** sont nécessaires.
- Les fonctions d'ensemble sont définies comme suit :

$$\mathfrak{a} = \alpha;\ \mathfrak{b} = -2\beta;\ \mathfrak{c} = \alpha + \beta;\ \mathfrak{p} = p;$$

  α, β, p étant les fonctions d'ensemble intervenant dans (1) et (2).
- Les méthodes sont alors définies dans notre approche par :

Method	$\mathfrak{a}(k,l)$	$\mathfrak{b}(k,l)$	$\mathfrak{c}(k,l)$	$\mathfrak{p}(i,j)$
Group average	$\frac{\lvert k\rvert}{\lvert k\rvert+\lvert l\rvert}$	0	$\frac{\lvert k\rvert}{\lvert k\rvert+\lvert l\rvert}$	1
Mcquitty	1/2	0	1/2	1
Centroid	$\frac{\lvert k\rvert}{\lvert k\rvert+\lvert l\rvert}$	$\frac{2\lvert k\rvert\lvert l\rvert}{(\lvert k\rvert+\lvert l\rvert)^2}$	$\frac{\lvert k\rvert^2}{(\lvert k\rvert+\lvert l\rvert)^2}$	1
Median	1/2	1/2	1/4	1
Ward	$\frac{\lvert k\rvert}{\lvert k\rvert+\lvert l\rvert}$	$\frac{2\lvert k\rvert\lvert l\rvert}{(\lvert k\rvert+\lvert l\rvert)^2}$	$\frac{\lvert k\rvert^2}{(\lvert k\rvert+\lvert l\rvert)^2}$	$\frac{\lvert i\rvert\lvert j\rvert}{\lvert i\rvert+\lvert j\rvert}$
W-Median	1/2	1/2	1/4	$\frac{\lvert i\rvert\lvert j\rvert}{\lvert i\rvert+\lvert j\rvert}$

Equivalence entre D-AHC et K-AHC

Proposition

Soient $\{\mathbf{D}^t\}_{t\in\mathbb{T}}$ *et* $\{\mathbf{\Lambda}^t\}_{t\in\mathbb{T}}$ *les séquences de matrices carrées avec pour éléments initiaux* $\mathbf{D}$ *et* $\mathbf{S}$ *; et éléments successifs definis par* (2)*, et* (3)-(5)-(6)*, respectivement.*
Alors, sous la condition (C1) *et si* $\mathfrak{a}(k,l)+\mathfrak{a}(l,k)=1, \forall k,l\in 2^{\mathbb{O}}$*, nous avons :* $\forall t\in\mathbb{T}, \forall i,j\in\mathbb{C}^t, i\neq j$*,*

$$\mathbf{\Lambda}_{ij}^t = -\frac{1}{2}\mathbf{D}_{ij}^t$$

Equivalence entre D-AHC et K-AHC

Proposition

Soient $\{\mathbf{D}^t\}_{t\in\mathbb{T}}$ *et* $\{\mathbf{\Lambda}^t\}_{t\in\mathbb{T}}$ *les séquences de matrices carrées avec pour éléments initiaux* $\mathbf{D}$ *et* $\mathbf{S}$ *; et éléments successifs definis par* (2)*, et* (3)-(5)-(6)*, respectivement.*
Alors, sous la condition (C1) *et si* $\mathfrak{a}(k,l)+\mathfrak{a}(l,k)=1, \forall k,l\in 2^{\mathbb{O}}$*, nous avons :* $\forall t\in\mathbb{T}, \forall i,j\in\mathbb{C}^t, i\neq j$*,*

$$\mathbf{\Lambda}^t_{ij} = -\frac{1}{2}\mathbf{D}^t_{ij}$$

- Ce résultat indique que l'approche classique (1)-(2) fondée sur les dissimilarités et notre approche (4)-(5)-(6) fondée sur les noyaux produisent des dendrogrammes equivalents.

Equivalence entre D-AHC et K-AHC

Proposition

Soient $\{\mathbf{D}^t\}_{t\in\mathbb{T}}$ *et* $\{\mathbf{\Lambda}^t\}_{t\in\mathbb{T}}$ *les séquences de matrices carrées avec pour éléments initiaux* $\mathbf{D}$ *et* $\mathbf{S}$ *; et éléments successifs definis par* (2)*, et* (3)-(5)-(6)*, respectivement.*
Alors, sous la condition (C1) *et si* $\mathfrak{a}(k,l)+\mathfrak{a}(l,k)=1, \forall k,l\in 2^{\mathbb{O}}$*, nous avons :* $\forall t\in\mathbb{T}, \forall i,j\in\mathbb{C}^t, i\neq j$*,*

$$\mathbf{\Lambda}^t_{ij} = -\frac{1}{2}\mathbf{D}^t_{ij}$$

- Ce résultat indique que l'approche classique (1)-(2) fondée sur les dissimilarités et notre approche (4)-(5)-(6) fondée sur les noyaux produisent des dendrogrammes equivalents.
- Par équivalence nous entendons le fait que les séquences de fusion de classes des deux dendrogrammes sont identiques (même si les valeurs de hauteur/profondeur sont différentes).

Rappel du Sommaire

Idée générale et propriétés de notre approche

- Si on raisonne avec des dissimilarités, les paires d'objets dont les valeurs de dissimilarité sont très grandes sont celles les moins pertinentes pour un regroupement dans une même classe.

Idée générale et propriétés de notre approche

- Si on raisonne avec des dissimilarités, les paires d'objets dont les valeurs de dissimilarité sont très grandes sont celles les moins pertinentes pour un regroupement dans une même classe.
- On pourrait alors alléger (**sparsifier**) **D** en écrêtant les valeurs très grandes en les remplaçant par 0. Mais ceci n'est pas cohérent.

Idée générale et propriétés de notre approche

- Si on raisonne avec des dissimilarités, les paires d'objets dont les valeurs de dissimilarité sont très grandes sont celles les moins pertinentes pour un regroupement dans une même classe.
- On pourrait alors alléger (**sparsifier**) **D** en écrêtant les valeurs très grandes en les remplaçant par 0. Mais ceci n'est pas cohérent.
- ▷ Si on avait des similarités non négatives, ce seraient les mesures très faibles et proches de 0 qui seraient les moins pertinentes. Dans ce cas, on pourrait alors les remplacer par 0 et raisonner avec une matrice creuse de façon cohérente.

Idée générale et propriétés de notre approche

- Si on raisonne avec des dissimilarités, les paires d'objets dont les valeurs de dissimilarité sont très grandes sont celles les moins pertinentes pour un regroupement dans une même classe.
- On pourrait alors alléger (**sparsifier**) **D** en écrêtant les valeurs très grandes en les remplaçant par 0. Mais ceci n'est pas cohérent.
- ▷ Si on avait des similarités non négatives, ce seraient les mesures très faibles et proches de 0 qui seraient les moins pertinentes. Dans ce cas, on pourrait alors les remplacer par 0 et raisonner avec une matrice creuse de façon cohérente.
- ▷ Une matrice creuse est **plus légère en mémoire** et permet également d'**améliorer le temps de traitement**.

Idée générale et propriétés de notre approche

- Si on raisonne avec des dissimilarités, les paires d'objets dont les valeurs de dissimilarité sont très grandes sont celles les moins pertinentes pour un regroupement dans une même classe.
- On pourrait alors alléger (**sparsifier**) **D** en écrêtant les valeurs très grandes en les remplaçant par 0. Mais ceci n'est pas cohérent.
- ▷ Si on avait des similarités non négatives, ce seraient les mesures très faibles et proches de 0 qui seraient les moins pertinentes. Dans ce cas, on pourrait alors les remplacer par 0 et raisonner avec une matrice creuse de façon cohérente.
- ▷ Une matrice creuse est **plus légère en mémoire** et permet également d'**améliorer le temps de traitement**.
- ▷ Restreindre les relations de proximité de chaque observation à son voisinage proche permet de mieux capter la **géométrie intrinsèque des données** et d'**améliorer ainsi la qualité des résultats de clustering** sur des données n'appartenant pas à un espace linéaire.

Normalisation et translation de la matrice des noyaux

- **Hypothèse géométrique 2** : les vecteurs $\mathbf{x}^a, \mathbf{x}^b, \ldots$ appartiennent à une hypersphère de $\mathcal{H}$. On a $\forall a, b \in \mathbb{O}$,

$$\mathbf{S}_{aa} = \mathbf{S}_{bb} \tag{C2}$$

Normalisation et translation de la matrice des noyaux

- **Hypothèse géométrique 2** : les vecteurs $\mathbf{x}^a, \mathbf{x}^b, \ldots$ appartiennent à une hypersphère de $\mathcal{H}$. On a $\forall a, b \in \mathbb{O}$,

$$\mathbf{S}_{aa} = \mathbf{S}_{bb} \tag{C2}$$

- Si $\mathbf{S}$ ne vérifie pas (C2), on peut projeter les objets sur une hypersphère par la **normalisation cosinus** : $\forall a, b \in \mathbb{O}$,

$$\mathbf{S}_{ab} \leftarrow \frac{\mathbf{S}_{ab}}{\sqrt{\mathbf{S}_{aa}\mathbf{S}_{bb}}}$$

Normalisation et translation de la matrice des noyaux

- **Hypothèse géométrique 2** : les vecteurs $\mathbf{x}^a, \mathbf{x}^b, \ldots$ appartiennent à une hypersphère de $\mathcal{H}$. On a $\forall a, b \in \mathbb{O}$,

$$\mathbf{S}_{aa} = \mathbf{S}_{bb} \tag{C2}$$

- Si $\mathbf{S}$ ne vérifie pas (C2), on peut projeter les objets sur une hypersphère par la **normalisation cosinus** : $\forall a, b \in \mathbb{O}$,

$$\mathbf{S}_{ab} \leftarrow \frac{\mathbf{S}_{ab}}{\sqrt{\mathbf{S}_{aa}\mathbf{S}_{bb}}}$$

- Ensuite on translate les valeurs de $\mathbf{S}$ de sorte à n'avoir que des valeurs non négatives : $\forall a, b \in \mathbb{O}$,

$$\mathbf{S}_{ab} \leftarrow \mathbf{S}_{ab} + |v|$$

où v est la plus petite valeur de $\mathbf{S}$.

Normalisation et translation de la matrice des noyaux

- **Hypothèse géométrique 2** : les vecteurs $\mathbf{x}^a, \mathbf{x}^b, \ldots$ appartiennent à une hypersphère de $\mathcal{H}$. On a $\forall a, b \in \mathbb{O}$,

$$\mathbf{S}_{aa} = \mathbf{S}_{bb} \tag{C2}$$

- Si $\mathbf{S}$ ne vérifie pas (C2), on peut projeter les objets sur une hypersphère par la **normalisation cosinus** : $\forall a, b \in \mathbb{O}$,

$$\mathbf{S}_{ab} \leftarrow \frac{\mathbf{S}_{ab}}{\sqrt{\mathbf{S}_{aa}\mathbf{S}_{bb}}}$$

- Ensuite on translate les valeurs de $\mathbf{S}$ de sorte à n'avoir que des valeurs non négatives : $\forall a, b \in \mathbb{O}$,

$$\mathbf{S}_{ab} \leftarrow \mathbf{S}_{ab} + |v|$$

où v est la plus petite valeur de $\mathbf{S}$.

▷ Par la suite on suppose que $\mathbf{S}$ est une **matrice de noyaux normalisés et de valeurs non négatives** (Normalized Kernels).

Propriétés de la matrice de noyaux normalisés

- **S** reste une matrice de noyaux (symétrique et sdp) qui peut, de plus, être interprétée telle une **matrice de similarités** sur $\mathbb{O}$ avec les axiomes suivants : $\forall a, b \in \mathbb{O}$,

$$\begin{cases} \mathbf{S}_{ab} \geq 0 \text{ (non négativité)} \\ \mathbf{S}_{ab} = \mathbf{S}_{ba} \text{ (symétrie)} \\ \mathbf{S}_{aa} \geq \mathbf{S}_{ab} \text{ (auto-similarité maximale)} \end{cases}$$

Propriétés de la matrice de noyaux normalisés

- **S** reste une matrice de noyaux (symétrique et sdp) qui peut, de plus, être interprétée telle une **matrice de similarités** sur $\mathbb{O}$ avec les axiomes suivants : $\forall a, b \in \mathbb{O}$,

$$\begin{cases} \mathbf{S}_{ab} \geq 0 \text{ (non négativité)} \\ \mathbf{S}_{ab} = \mathbf{S}_{ba} \text{ (symétrie)} \\ \mathbf{S}_{aa} \geq \mathbf{S}_{ab} \text{ (auto-similarité maximale)} \end{cases}$$

- L'inégalité de Cauchy-Schwartz implique l'auto-similarité maximale.

Propriétés de la matrice de noyaux normalisés

- **S** reste une matrice de noyaux (symétrique et sdp) qui peut, de plus, être interprétée telle une **matrice de similarités** sur $\mathbb{O}$ avec les axiomes suivants : $\forall a, b \in \mathbb{O}$,

$$\begin{cases} \mathbf{S}_{ab} \geq 0 \text{ (non négativité)} \\ \mathbf{S}_{ab} = \mathbf{S}_{ba} \text{ (symétrie)} \\ \mathbf{S}_{aa} \geq \mathbf{S}_{ab} \text{ (auto-similarité maximale)} \end{cases}$$

- L'inégalité de Cauchy-Schwartz implique l'auto-similarité maximale.
- La translation de **S** n'a pas d'incidence sur le résultat de la AHC.

Propriétés de la matrice de noyaux normalisés

- **S** reste une matrice de noyaux (symétrique et sdp) qui peut, de plus, être interprétée telle une **matrice de similarités** sur $\mathbb{O}$ avec les axiomes suivants : $\forall a, b \in \mathbb{O}$,

$$\begin{cases} \mathbf{S}_{ab} \geq 0 \text{ (non négativité)} \\ \mathbf{S}_{ab} = \mathbf{S}_{ba} \text{ (symétrie)} \\ \mathbf{S}_{aa} \geq \mathbf{S}_{ab} \text{ (auto-similarité maximale)} \end{cases}$$

- L'inégalité de Cauchy-Schwartz implique l'auto-similarité maximale.
- La translation de **S** n'a pas d'incidence sur le résultat de la AHC.
- Dans ce qui suit, nous interprétons les noyaux normalisés tels des indices de similarités et nous donnons une autre interprétation des méthodes AHC basée sur le concept de **similarités pénalisées** et sur l'utilisation implicite de différentes stratégies d'agrégation par **moyennes pondérées**.

Similarités pénalisées

- Rappelons la règle de fusion (4) :

$$(k, l) = \underset{(i,j)\in\mathbb{C}^t\times\mathbb{C}^t, i\neq j}{\arg\max} \overbrace{\mathfrak{p}(i,j)}^{\text{Poids}} \underbrace{\mathbf{\Lambda}_{ij}^t}_{\text{Sim. pénalisée}}$$

Similarités pénalisées

- Rappelons la règle de fusion (4) :

$$(k, l) = \underset{(i,j)\in\mathbb{C}^t\times\mathbb{C}^t, i\neq j}{\arg\max} \overbrace{\mathfrak{p}(i,j)}^{\text{Poids}} \underbrace{\mathbf{\Lambda}_{ij}^t}_{\text{Sim. pénalisée}}$$

- $\mathbf{\Lambda}^t$ est interprétée comme une **matrice de similarités pénalisées** :

$$\mathbf{\Lambda}_{ij}^t = \overbrace{\mathbf{S}_{ij}^t}^{\text{Inter-sim.}} - \underbrace{\frac{1}{2}(\overbrace{\mathbf{S}_{ii}^t + \mathbf{S}_{jj}^t}^{\text{Intra-sim.}})}_{\text{Pénalité}}$$

Similarités pénalisées

- Rappelons la règle de fusion (4) :

$$(k, l) = \underset{(i,j)\in\mathbb{C}^t\times\mathbb{C}^t, i\neq j}{\arg\max} \overbrace{\mathfrak{p}(i,j)}^{\text{Poids}} \underbrace{\mathbf{\Lambda}_{ij}^t}_{\text{Sim. pénalisée}}$$

- $\mathbf{\Lambda}^t$ est interprétée comme une **matrice de similarités pénalisées** :

$$\mathbf{\Lambda}_{ij}^t = \overbrace{\mathbf{S}_{ij}^t}^{\text{Inter-sim.}} - \underbrace{\frac{1}{2}(\overbrace{\mathbf{S}_{ii}^t + \mathbf{S}_{jj}^t}^{\text{Intra-sim.}})}_{\text{Pénalité}}$$

- **Pénalité** = moyenne arithmétique des intra-similarités.

Similarités pénalisées

- Rappelons la règle de fusion (4) :

$$(k,l) = \underset{(i,j)\in\mathbb{C}^t\times\mathbb{C}^t, i\neq j}{\arg\max} \overbrace{\mathfrak{p}(i,j)}^{\text{Poids}} \underbrace{\mathbf{\Lambda}_{ij}^t}_{\text{Sim. pénalisée}}$$

- $\mathbf{\Lambda}^t$ est interprétée comme une **matrice de similarités pénalisées** :

$$\mathbf{\Lambda}_{ij}^t = \overbrace{\mathbf{S}_{ij}^t}^{\text{Inter-sim.}} - \underbrace{\frac{1}{2}(\overbrace{\mathbf{S}_{ii}^t + \mathbf{S}_{jj}^t}^{\text{Intra-sim.}})}_{\text{Pénalité}}$$

- **Pénalité** = moyenne arithmétique des intra-similarités.
- A mesure d'inter-similarité constante, on fusionnera d'abord les deux classes dont la mesure de pénalité est la plus faible. Autrement dit, deux classes fortement homogènes et relativement peu similaire entre elles, se regrouperont tardivement dans la hiérarchie.

Différentes stratégies d'agrégation

- L'ensemble des méthodes à l'étude sont telles que : $\forall k, l \in 2^{\mathbb{O}}$,

$$\begin{cases} \mathfrak{a}(k,l), \mathfrak{b}(k,l), \mathfrak{c}(k,l) \geq 0 \\ \mathfrak{a}(k,l) + \mathfrak{a}(l,k) = 1 \\ \mathfrak{b}(k,l) + \mathfrak{c}(k,l) + \mathfrak{c}(l,k) = 1 \end{cases}$$

  Elles définissent donc des **moyennes pondérées**.

Différentes stratégies d'agrégation

- L'ensemble des méthodes à l'étude sont telles que : $\forall k, l \in 2^{\mathbb{O}}$,

$$\begin{cases} \mathfrak{a}(k,l), \mathfrak{b}(k,l), \mathfrak{c}(k,l) \geq 0 \\ \mathfrak{a}(k,l) + \mathfrak{a}(l,k) = 1 \\ \mathfrak{b}(k,l) + \mathfrak{c}(k,l) + \mathfrak{c}(l,k) = 1 \end{cases}$$

  Elles définissent donc des **moyennes pondérées**.

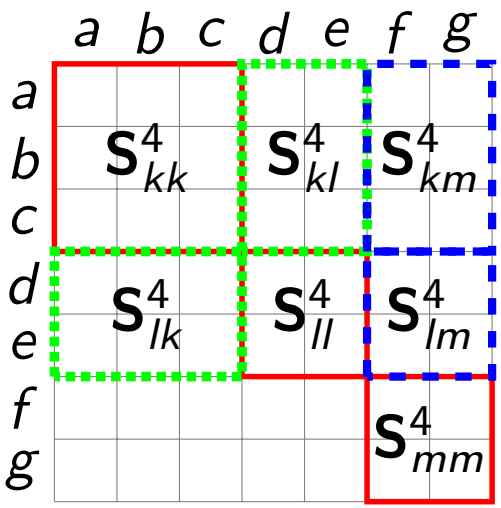
▷ La différence entre les méthodes peut s'interpréter comme **différentes stratégies d'agrégation** d'inter-similarités et d'intra-similarités :

sim. pénalisée=moyenne sim inter - moyenne(moyennes sim intra)

Illustration

- $k = \{a, b, c\}; l = \{d, e\}; m = \{f, g\}$
- k et l fusionnent.

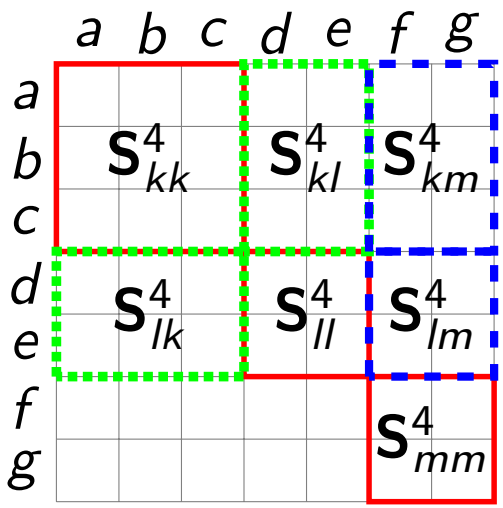
$$\mathbf{S}^{t+1}_{(kl)m} = \mathfrak{a}(k, l)\mathbf{S}^{t}_{km} + \mathfrak{a}(l, k)\mathbf{S}^{t}_{lm}$$
$$\mathbf{S}^{t+1}_{(kl)(kl)} = \mathfrak{b}(k, l)\mathbf{S}^{t}_{kl} + \mathfrak{c}(k, l)\mathbf{S}^{t}_{kk} + \mathfrak{c}(l, k)\mathbf{S}^{t}_{ll}$$



Illustration

- $k = \{a, b, c\}; l = \{d, e\}; m = \{f, g\}$
- k et l fusionnent.

$$\mathbf{S}^{t+1}_{(kl)m} = \mathfrak{a}(k,l)\mathbf{S}^{t}_{km} + \mathfrak{a}(l,k)\mathbf{S}^{t}_{lm}$$
$$\mathbf{S}^{t+1}_{(kl)(kl)} = \mathfrak{b}(k,l)\mathbf{S}^{t}_{kl} + \mathfrak{c}(k,l)\mathbf{S}^{t}_{kk} + \mathfrak{c}(l,k)\mathbf{S}^{t}_{ll}$$



Method	$\mathfrak{a}(k,l)$	$\mathfrak{b}(k,l)$	$\mathfrak{c}(k,l)$	$\mathfrak{p}(i,j)$
Group average	$\frac{\lvert k\rvert}{\lvert k\rvert+\lvert l\rvert}$	0	$\frac{\lvert k\rvert}{\lvert k\rvert+\lvert l\rvert}$	1
Mcquitty	1/2	0	1/2	1
Centroid	$\frac{\lvert k\rvert}{\lvert k\rvert+\lvert l\rvert}$	$\frac{2\lvert k\rvert\lvert l\rvert}{(\lvert k\rvert+\lvert l\rvert)^2}$	$\frac{\lvert k\rvert^2}{(\lvert k\rvert+\lvert l\rvert)^2}$	1
Median	1/2	1/2	1/4	1
Ward	$\frac{\lvert k\rvert}{\lvert k\rvert+\lvert l\rvert}$	$\frac{2\lvert k\rvert\lvert l\rvert}{(\lvert k\rvert+\lvert l\rvert)^2}$	$\frac{\lvert k\rvert^2}{(\lvert k\rvert+\lvert l\rvert)^2}$	$\frac{\lvert i\rvert\lvert j\rvert}{\lvert i\rvert+\lvert j\rvert}$
W-Median	1/2	1/2	1/4	$\frac{\lvert i\rvert\lvert j\rvert}{\lvert i\rvert+\lvert j\rvert}$

Sparsification de **S**

- On revient sur le pb de passage à l'échelle et de prise en compte de la géométrie des données.

Sparsification de **S**

- On revient sur le pb de passage à l'échelle et de prise en compte de la géométrie des données.
- On suggère deux méthodes classiques de **sparsification** :
  - L'une basée sur un **seuil** $\theta > 0$:

$$\mathbf{S}_{ab} \leftarrow \begin{cases} \mathbf{S}_{ab} & \text{if } \mathbf{S}_{ab} \geq \theta \\ 0 & \text{otherwise} \end{cases}$$

Sparsification de **S**

- On revient sur le pb de passage à l'échelle et de prise en compte de la géométrie des données.
- On suggère deux méthodes classiques de **sparsification** :
  - L'une basée sur un **seuil** $\theta > 0$:

$$\mathbf{S}_{ab} \leftarrow \begin{cases} \mathbf{S}_{ab} & \text{if } \mathbf{S}_{ab} \geq \theta \\ 0 & \text{otherwise} \end{cases}$$

  - L'autre basée sur les k **plus proches voisins** :

$$\mathbf{S}_{ab} \leftarrow \begin{cases} \mathbf{S}_{ab} & \text{if } b \in \mathbb{NN}_k(a) \text{ or } a \in \mathbb{NN}_k(b) \\ 0 & \text{otherwise} \end{cases}$$

Sparsification de **S**

- On revient sur le pb de passage à l'échelle et de prise en compte de la géométrie des données.
- On suggère deux méthodes classiques de **sparsification** :
  - L'une basée sur un **seuil** $\theta > 0$:

$$\mathbf{S}_{ab} \leftarrow \begin{cases} \mathbf{S}_{ab} & \text{if } \mathbf{S}_{ab} \geq \theta \\ 0 & \text{otherwise} \end{cases}$$

  - L'autre basée sur les k **plus proches voisins** :

$$\mathbf{S}_{ab} \leftarrow \begin{cases} \mathbf{S}_{ab} & \text{if } b \in \mathbb{NN}_k(a) \text{ or } a \in \mathbb{NN}_k(b) \\ 0 & \text{otherwise} \end{cases}$$

- Par la suite **S** est une matrice de similarités creuse (Sparse Normalized Kernels).

Sparsification de **S**

- On revient sur le pb de passage à l'échelle et de prise en compte de la géométrie des données.
- On suggère deux méthodes classiques de **sparsification** :
  - L'une basée sur un **seuil** $\theta > 0$:

$$\mathbf{S}_{ab} \leftarrow \begin{cases} \mathbf{S}_{ab} & \text{if } \mathbf{S}_{ab} \geq \theta \\ 0 & \text{otherwise} \end{cases}$$

  - L'autre basée sur les k **plus proches voisins** :

$$\mathbf{S}_{ab} \leftarrow \begin{cases} \mathbf{S}_{ab} & \text{if } b \in \mathbb{NN}_k(a) \text{ or } a \in \mathbb{NN}_k(b) \\ 0 & \text{otherwise} \end{cases}$$

- Par la suite **S** est une matrice de similarités creuse (Sparse Normalized Kernels).
- **S** n'est plus sdp après la sparsification et donc les hypothèses géométriques initiales ne sont plus valides.

Sparsification de **S**

- On revient sur le pb de passage à l'échelle et de prise en compte de la géométrie des données.
- On suggère deux méthodes classiques de **sparsification** :
  - L'une basée sur un **seuil** $\theta > 0$:

$$\mathbf{S}_{ab} \leftarrow \begin{cases} \mathbf{S}_{ab} & \text{if } \mathbf{S}_{ab} \geq \theta \\ 0 & \text{otherwise} \end{cases}$$

  - L'autre basée sur les k **plus proches voisins** :

$$\mathbf{S}_{ab} \leftarrow \begin{cases} \mathbf{S}_{ab} & \text{if } b \in \mathbb{NN}_k(a) \text{ or } a \in \mathbb{NN}_k(b) \\ 0 & \text{otherwise} \end{cases}$$

- Par la suite **S** est une matrice de similarités creuse (Sparse Normalized Kernels).
- **S** n'est plus sdp après la sparsification et donc les hypothèses géométriques initiales ne sont plus valides.

▷ Mais on verra que ceci n'est pas un pb pour certaines méthodes.

SNK-AHC (Sparse Normalized Kernels based AHC)

- Les procédures D-AHC et K-AHC précédentes ont un coût en $O(n^3)$: il faut $n-1$ itérations pour construire le dendrogramme et à chaque $t \in \mathbb{T}$, il faut trouver la paire optimale qui coûte au pire $O(n^2)$.

SNK-AHC (Sparse Normalized Kernels based AHC)

- Les procédures D-AHC et K-AHC précédentes ont un coût en $O(n^3)$: il faut $n-1$ itérations pour construire le dendrogramme et à chaque $t \in \mathbb{T}$, il faut trouver la paire optimale qui coûte au pire $O(n^2)$.
- La recherche de la paire optimale est le **goulot d'étranglement**. Dans l'extension de notre approche, SNK-AHC, on cherche cette paire dans le sous-ensemble des paires de **similarités strictement positives**. On introduit les sous-ensembles suivants : $\forall t \in \mathbb{T}$,

$$\mathbb{S}^t = \{(i,j) \in \mathbb{C}^t \times \mathbb{C}^t, \mathbf{S}^t_{ij} > 0\} \tag{7}$$

SNK-AHC (Sparse Normalized Kernels based AHC)

- Les procédures D-AHC et K-AHC précédentes ont un coût en $O(n^3)$: il faut $n-1$ itérations pour construire le dendrogramme et à chaque $t \in \mathbb{T}$, il faut trouver la paire optimale qui coûte au pire $O(n^2)$.
- La recherche de la paire optimale est le **goulot d'étranglement**. Dans l'extension de notre approche, SNK-AHC, on cherche cette paire dans le sous-ensemble des paires de **similarités strictement positives**. On introduit les sous-ensembles suivants : $\forall t \in \mathbb{T}$,

$$\mathbb{S}^t = \{(i,j) \in \mathbb{C}^t \times \mathbb{C}^t, \mathbf{S}^t_{ij} > 0\} \tag{7}$$

▷ La règle de fusion (4) est alors remplacée par :

$$\boxed{(k,l) = \underset{(i,j)\in\mathbb{S}^t, i\neq j}{\arg\max}\ \mathfrak{p}(i,j)\mathbf{\Lambda}^t_{ij}} \tag{8}$$

SNK-AHC (Sparse Normalized Kernels based AHC)

- Les procédures D-AHC et K-AHC précédentes ont un coût en $O(n^3)$: il faut $n-1$ itérations pour construire le dendrogramme et à chaque $t \in \mathbb{T}$, il faut trouver la paire optimale qui coûte au pire $O(n^2)$.
- La recherche de la paire optimale est le **goulot d'étranglement**. Dans l'extension de notre approche, SNK-AHC, on cherche cette paire dans le sous-ensemble des paires de **similarités strictement positives**. On introduit les sous-ensembles suivants : $\forall t \in \mathbb{T}$,

$$\mathbb{S}^t = \{(i,j) \in \mathbb{C}^t \times \mathbb{C}^t, \mathbf{S}^t_{ij} > 0\} \tag{7}$$

▷ La règle de fusion (4) est alors remplacée par :

$$\boxed{(k,l) = \underset{(i,j)\in\mathbb{S}^t, i\neq j}{\arg\max}\ \mathfrak{p}(i,j)\mathbf{\Lambda}^t_{ij}} \tag{8}$$

- Ainsi, pour que i et j puissent fusionner, il est nécessaire que leur inter-similarité soit strictement positive.

SNK-AHC (suite)

- On introduit **S**parse **N**ormalized **K**ernels based **A**gglo. **H**ier. **C**lust.

Input: **S** (kernel matrix), sparsification method, AHC method
Output: D a dendrogram

1 **if** the diagonal of **S** is not constant **then**
2 | Normalize **S** using cosine normalization;
3 **end**
4 Translate **S** in order to have non negative values;
5 Sparsify **S** in order to have a sparse **S**;
6 Initialize D with n leaves;
7 Set $\mathbf{S}^1 = \mathbf{S}$;
8 Determine $\mathbb{S}^1$ according to (7);
9 **while** $\mathbb{S}^t \neq \emptyset$ **do**
10 | Find the pair of clusters (k, l) according to (8) ;
11 | Merge (k, l) into (kl) and update D;
12 | Update $\mathbb{S}^{t+1}$ from $\mathbb{S}^t$;
13 | Compute $\mathbf{S}^{t+1}$ by applying (5) and (6).
14 **end**

Complexité

Proposition

Soit **S** *la matrice de similarités sparse obtenue après l'étape 5 de l'algorithme SNK-AHC. Soit* z *le nombre d'entrées non-nulles de* **S**. *La construction du dendrogramme donnée par les étapes 6 à 14 de l'algorithme SNK-AHC, a une complexité en mémoire en* $O(z)$ *et une complexité en temps de traitement en* $O(nz)$.

Invariance par translation de la diagonale

- Après sparsification **S** n'est plus sdp.

Invariance par translation de la diagonale

- Après sparsification **S** n'est plus sdp.
- On peut **augmenter sa diagonale** pour la rendre sdp à nouveau mais ceci aboutit à une "distorsion" de l'espace initial.

Invariance par translation de la diagonale

- Après sparsification **S** n'est plus sdp.
- On peut **augmenter sa diagonale** pour la rendre sdp à nouveau mais ceci aboutit à une "distorsion" de l'espace initial.
- Certaines méthodes sont **invariantes par translation de la diagonale** !

Invariance par translation de la diagonale

- Après sparsification **S** n'est plus sdp.
- On peut **augmenter sa diagonale** pour la rendre sdp à nouveau mais ceci aboutit à une "distorsion" de l'espace initial.
- Certaines méthodes sont **invariantes par translation de la diagonale** !

Proposition

Soit $\mathbf{S}$ *la matrice de similarités sparse obtenue après l'étape 5 de l'algorithme SNK-AHC. Pour les méthodes group average, Mcquitty et Ward, l'algorithme SNK-AHC produit deux dendrogrammes equivalents si l'on prend comme matrices sparses* $\mathbf{S}$ *et* $\mathbf{T} = \mathbf{S} + w\mathbf{I}_n$, $w \in \mathbb{R}$.

Invariance par translation de la diagonale

- Après sparsification **S** n'est plus sdp.
- On peut **augmenter sa diagonale** pour la rendre sdp à nouveau mais ceci aboutit à une "distorsion" de l'espace initial.
- Certaines méthodes sont **invariantes par translation de la diagonale** !

Proposition

Soit $\mathbf{S}$ *la matrice de similarités sparse obtenue après l'étape 5 de l'algorithme SNK-AHC. Pour les méthodes group average, Mcquitty et Ward, l'algorithme SNK-AHC produit deux dendrogrammes equivalents si l'on prend comme matrices sparses* $\mathbf{S}$ *et* $\mathbf{T} = \mathbf{S} + w\mathbf{I}_n$, $w \in \mathbb{R}$.

- Donc pour ces trois techniques, les hypothèses géométriques (C1) et (C2) restent valides même si **S** n'est pas sdp.

Composantes connexes et nombre de clusters

- Nous interprétons **S** telle la matrice d'adjacence pondérée d'un graphe non orienté sur $\mathbb{O}$.

Composantes connexes et nombre de clusters

- Nous interprétons **S** telle la matrice d'adjacence pondérée d'un graphe non orienté sur $\mathbb{O}$. Si **S** est sparse, le graphe n'est pas complet et peut contenir **plusieurs composantes connexes**.

Composantes connexes et nombre de clusters

- Nous interprétons **S** telle la matrice d'adjacence pondérée d'un graphe non orienté sur $\mathbb{O}$. Si **S** est sparse, le graphe n'est pas complet et peut contenir **plusieurs composantes connexes**.
- En fait, les étapes de fusion de SNK-AHC sont identiques à celles d'un algorithme permettant de détecter les **sous-ensembles disjoints** des sommets d'un graphe non connexe (s'il existe un chemin rejoignant deux sommets alors ils sont mis dans le même sous-ensemble).

Composantes connexes et nombre de clusters

- Nous interprétons **S** telle la matrice d'adjacence pondérée d'un graphe non orienté sur $\mathbb{O}$. Si **S** est sparse, le graphe n'est pas complet et peut contenir **plusieurs composantes connexes**.
- En fait, les étapes de fusion de SNK-AHC sont identiques à celles d'un algorithme permettant de détecter les **sous-ensembles disjoints** des sommets d'un graphe non connexe (s'il existe un chemin rejoignant deux sommets alors ils sont mis dans le même sous-ensemble).

Proposition

Soit **S** *la matrice de similarités sparse obtenue après l'étape 5 et* $\mathbb{S} = \mathbb{S}^1$ *le sous-ensemble de couples d'objets obtenu après l'étape 8 de l'algo. SNK-AHC. Soit* $G = (\mathbb{O}, \mathbb{S})$ *le graphe non orienté sur* $\mathbb{O}$ *et de matrice d'adjacence* $\mathbb{S}$*. Si* G *possède* κ *composantes connexes alors l'algo. SNK-AHC s'arrête à l'iteration* $n - \kappa - 1$ *et donne comme résultat une fôret d'arbres binaires où chaque arbre est une composante connexe.*

Rappel du Sommaire

Protocol expérimental

- Illustrations des propriétés sur deux jeux de données artifielles (“aggregation” et “compound”) et deux jeux de données réelles (“landsat”, “pendigits”).

Protocol expérimental

- Illustrations des propriétés sur deux jeux de données artifielles (“aggregation” et “compound”) et deux jeux de données réelles (“landsat”, “pendigits”).
- **Plusieurs niveaux de sparsification** utilisant soit le seuil θ soit les k plus proches voisins.

Protocol expérimental

- Illustrations des propriétés sur deux jeux de données artifielles ("aggregation" et "compound") et deux jeux de données réelles ("landsat", "pendigits").
- **Plusieurs niveaux de sparsification** utilisant soit le seuil θ soit les k plus proches voisins.
- **Coefficient cophénétique (CC)** pour mesurer la similarité entre le dendrogramme de la baseline et ceux donnés par SNK-AHC. Le résultat de référence est celui obtenu avec la matrice complète **S** (qui est donc équivalent à l'approche classique D-AHC).

Protocol expérimental

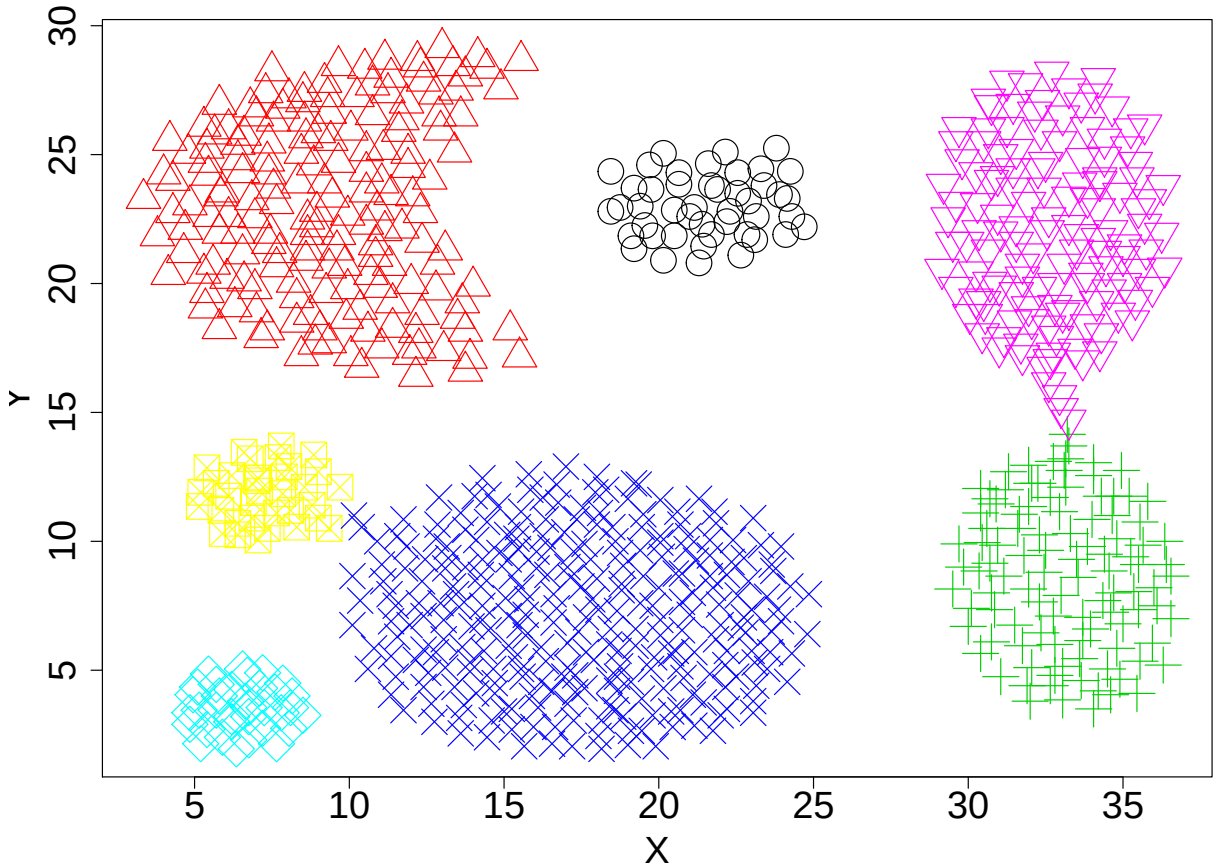
- Illustrations des propriétés sur deux jeux de données artifielles ("aggregation" et "compound") et deux jeux de données réelles ("landsat", "pendigits").
- **Plusieurs niveaux de sparsification** utilisant soit le seuil θ soit les k plus proches voisins.
- **Coefficient cophénétique (CC)** pour mesurer la similarité entre le dendrogramme de la baseline et ceux donnés par SNK-AHC. Le résultat de référence est celui obtenu avec la matrice complète **S** (qui est donc équivalent à l'approche classique D-AHC).
- L'**indice de Rand corrigé (ARI)** pour mesurer la qualité du résultat de clustering vis à vis de la vérité terrain (on coupe le dendrogramme au nombre correct de clusters).

Protocol expérimental

- Illustrations des propriétés sur deux jeux de données artifielles ("aggregation" et "compound") et deux jeux de données réelles ("landsat", "pendigits").
- **Plusieurs niveaux de sparsification** utilisant soit le seuil θ soit les k plus proches voisins.
- **Coefficient cophénétique (CC)** pour mesurer la similarité entre le dendrogramme de la baseline et ceux donnés par SNK-AHC. Le résultat de référence est celui obtenu avec la matrice complète **S** (qui est donc équivalent à l'approche classique D-AHC).
- L'**indice de Rand corrigé (ARI)** pour mesurer la qualité du résultat de clustering vis à vis de la vérité terrain (on coupe le dendrogramme au nombre correct de clusters).
- Les **diminutions** de stockage **mémoire** et de **temps de traitement** lorsque **S** est de plus en plus sparse, sont mesurées relativement aux performances obtenues avec la matrice complète **S**.

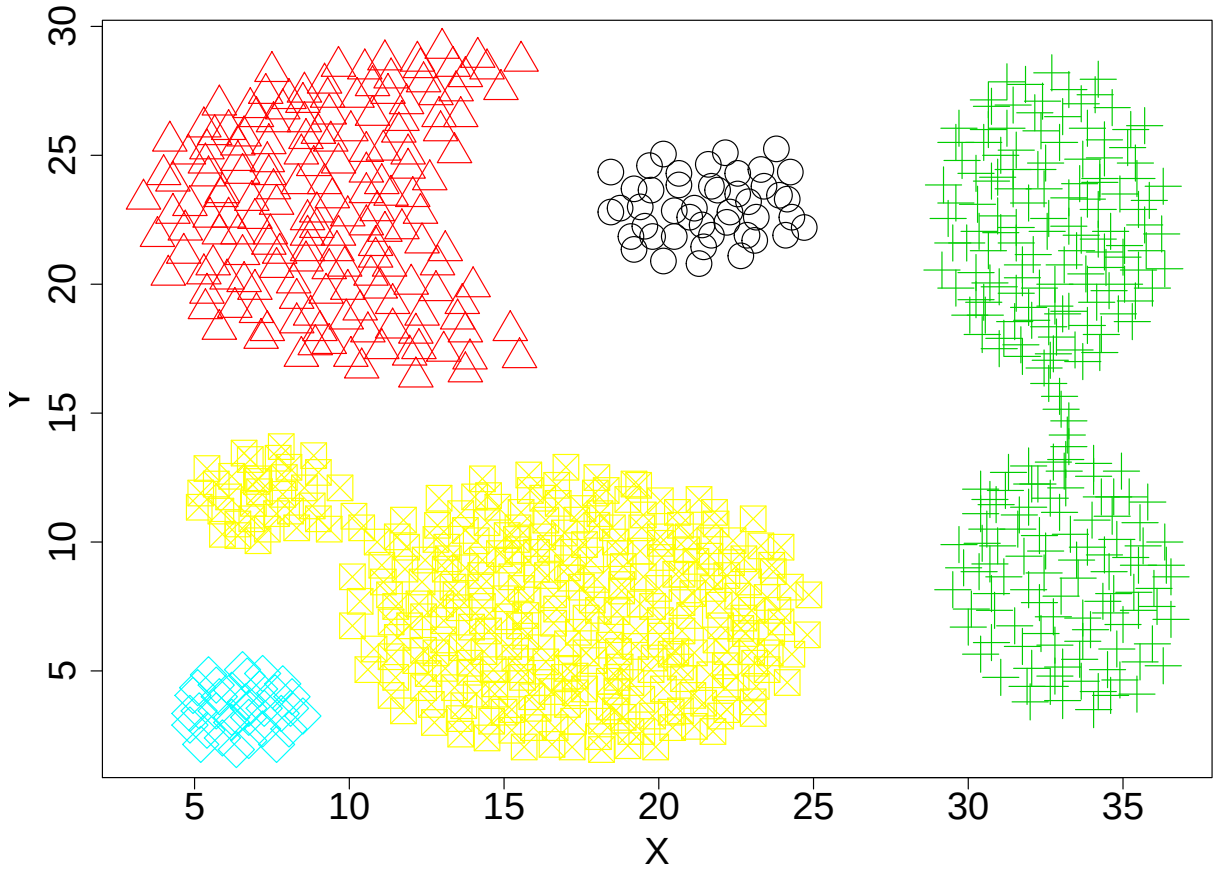
Données “aggregation”

- 788 observations en 2D.
- 7 clusters.



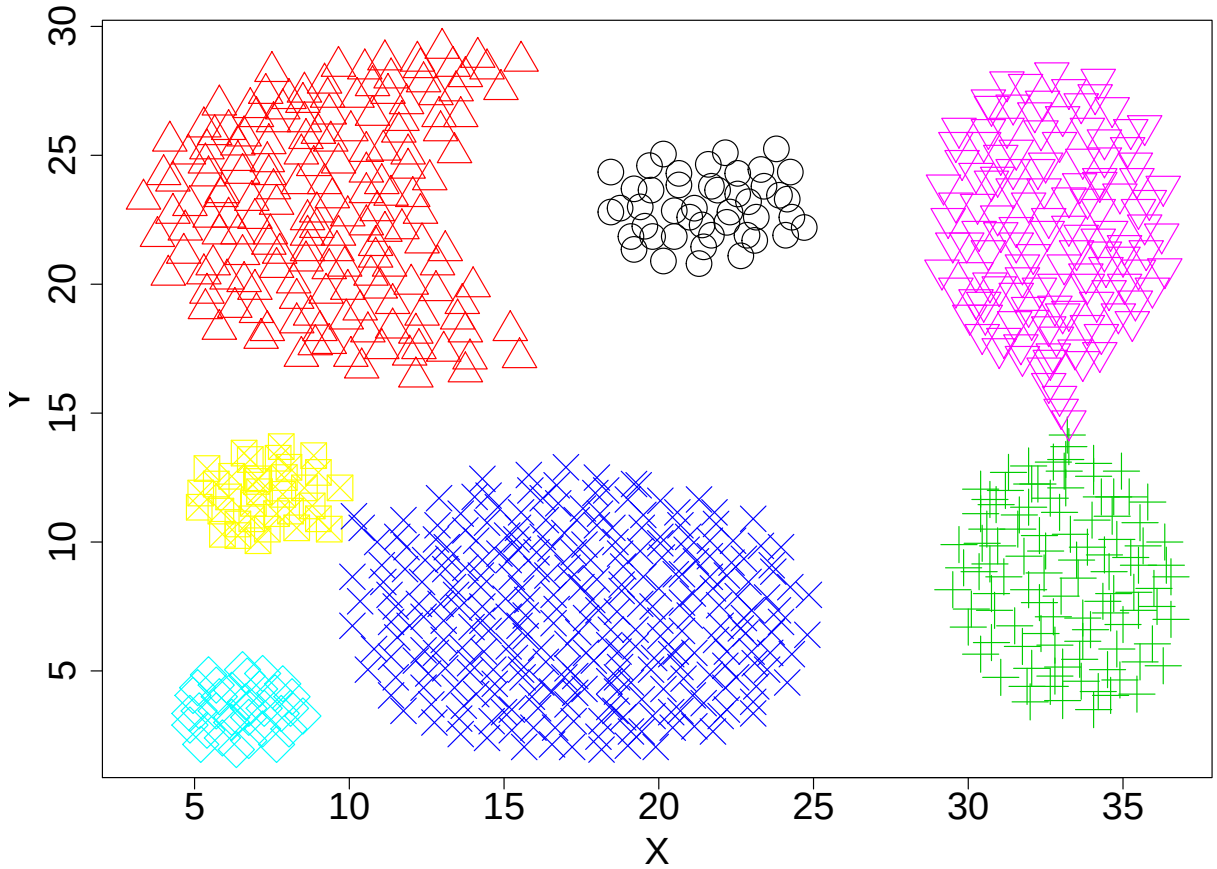
Résultats sur les données “aggregation”

- Résultats avec noyau Gaussien, $k = 8$ ($\sim$10% de knn).
- 5 composantes connexes sont détectées automatiquement.



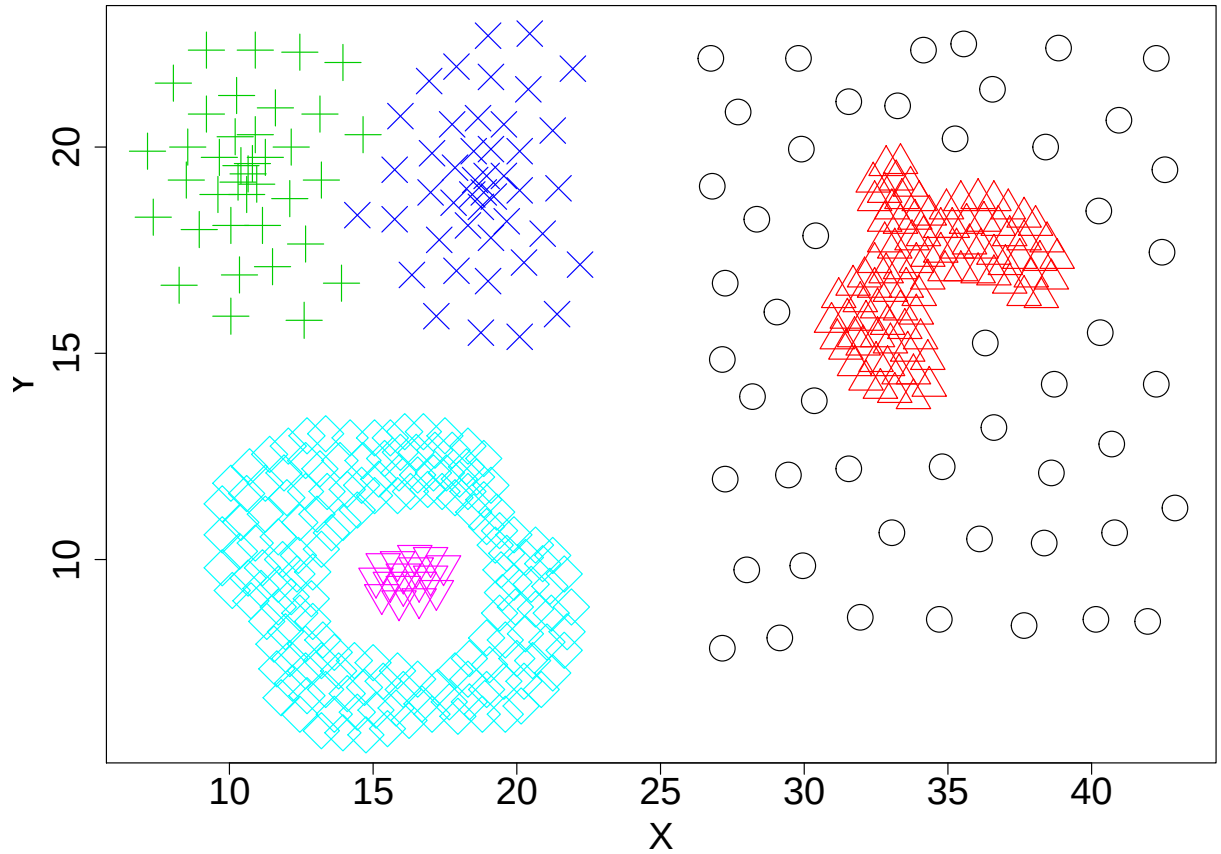
Résultats sur les données “aggregation”

- Résultats avec noyau Gaussien, $k = 8$ ($\sim$10% de knn).
- Si on coupe à 7 clusters on obtient la solution exacte.



Données “compound”

- 399 observations en 2D.
- 6 clusters.



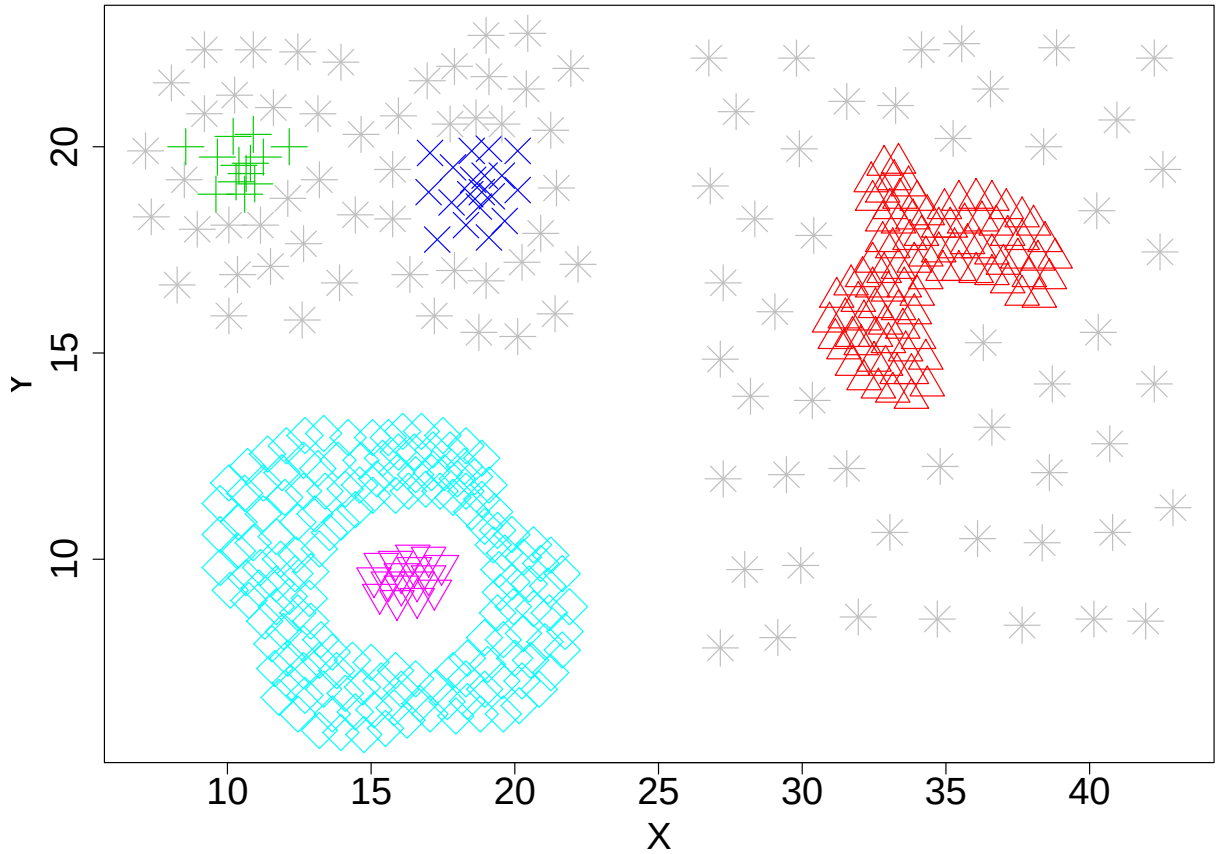
Résultats sur les données “compound”

- Résultats avec noyau Gaussien et la méthode group average.
- Le seuil de sparsification θ varie.

Method	θ	CC	ARI	κ
Group average	0.010	1.000	0.811	1
	0.143	1.000	0.811	1
	0.245	1.000	0.811	1
	0.463	0.999	0.811	1
	0.819	0.947	0.802	1
	0.948	-0.766	0.818	3
	0.996	-0.741	**0.906**	99

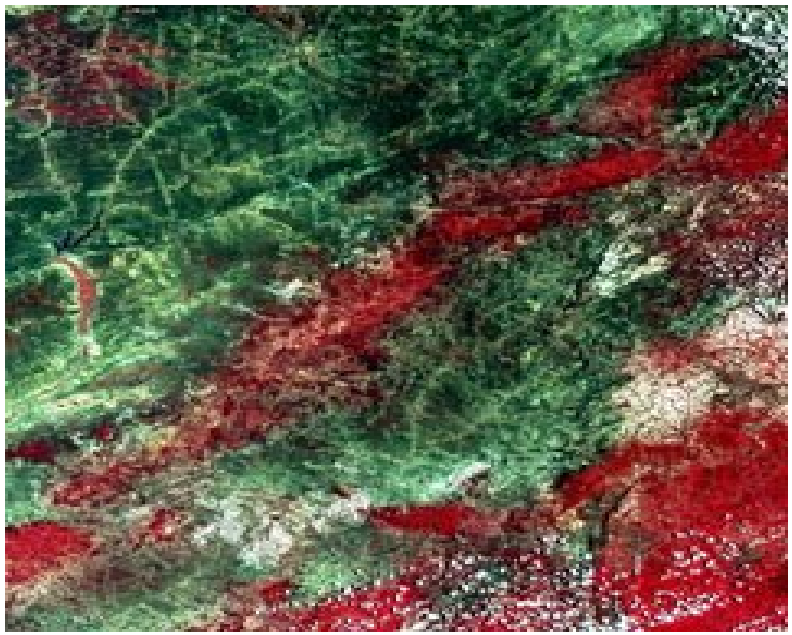
Résultats sur les données “compound” (suite)

- Illustration avec $\theta = 0.996$, on obtient 99 composantes connexes.
- Si les clusters sont de taille ≤ 3, on représente les individus en gris.



Données “landsat”

- Images satellites multispectrales. Jeu de données disponible sur UCI [2].
- 6,435 observations (1 obs = patch de 3x3 pixels)
- 36 variables (chaque pixel a 4 valeurs de spectres)
- 6 clusters : red soil, cotton crop, grey soil, damp grey soil, soil with vegetation stubble, very damp grey soil.



2. https://archive.ics.uci.edu/ml/datasets/Statlog+(Landsat+Satellite)

Résultats sur les données “landsat”

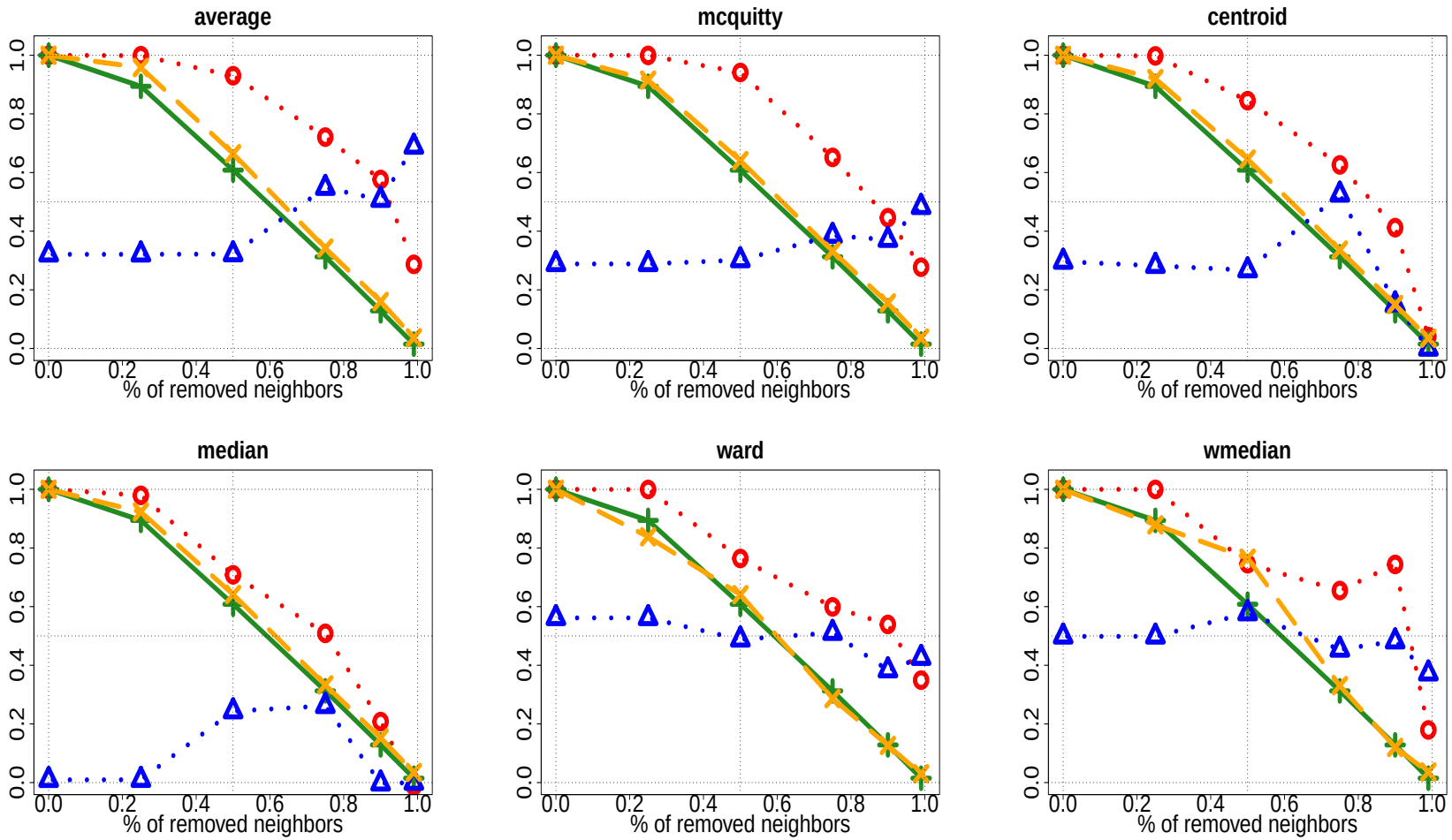
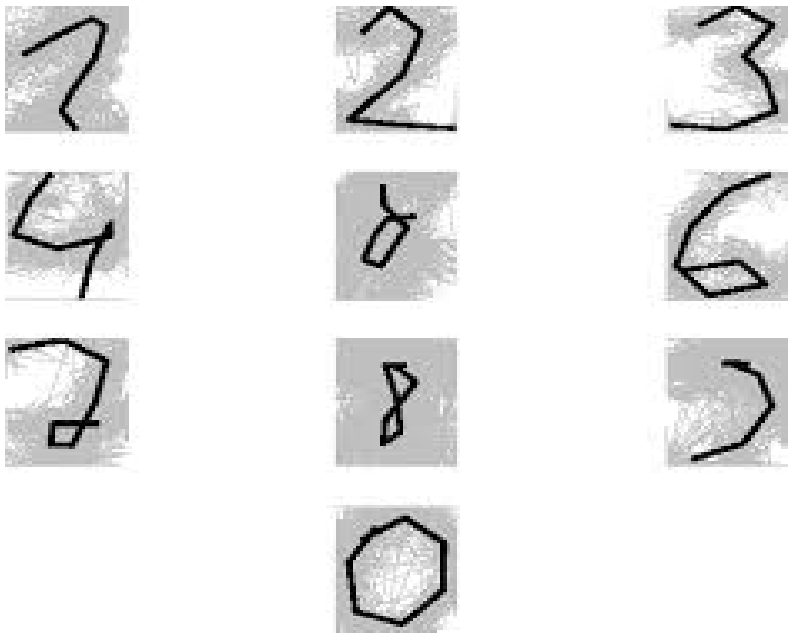


FIGURE – The x-axis corresponds to the % of removed neighbors. The y-axis corresponds to the observed values in $[0, 1]$. Solid lines with plus signs represent the relative memory use, dotted lines with circles indicates the CC values, dotted lines with triangles give the ARI values.

Données “pendigits”

- Reconnaissance de chiffres écrits à la main par 44 personnes différentes. Jeu de données disponible sur UCI [3].
- 10,992 observations (1 obs = 1 chiffre = $\{x_t, y_t\}_{t=1,\ldots,8}$)
- 16 variables (chaque pixel a 4 valeurs de spectres)
- 10 clusters : 0,. . . ,9.



3. https://archive.ics.uci.edu/ml/datasets/Pen-Based+Recognition+of+Handwritten+Digits

Résultats sur les données “pendigits”

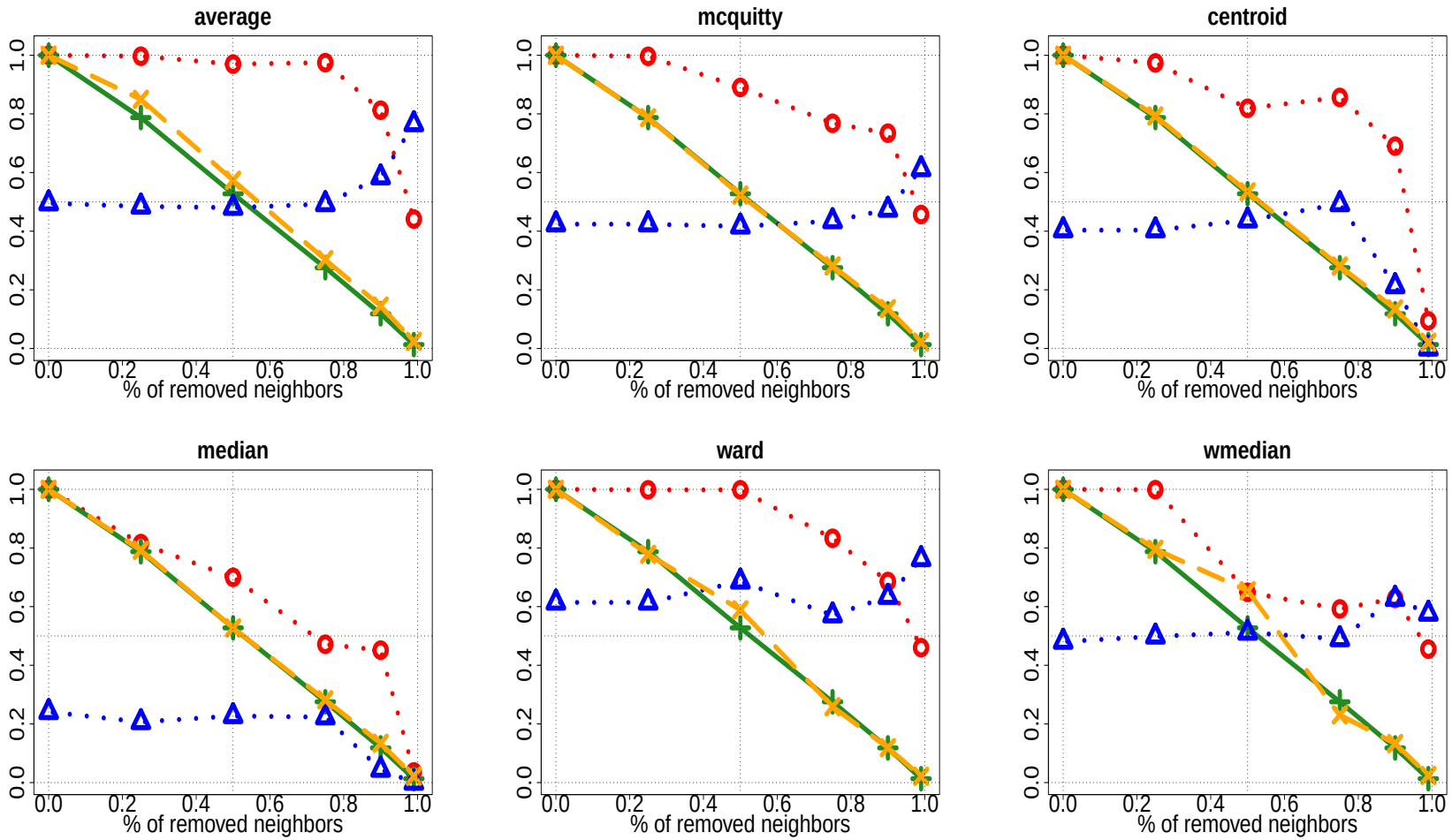


FIGURE – Solid lines with plus signs represent the relative memory use, dashed lines with cross signs show the relative running time, dotted lines with circles indicate the CC values, dotted lines with triangles give the ARI values.

Rappel du Sommaire

Lien avec le partitionnement spectral / Travaux futurs

- Liens avec les techniques modernes de partionnement spectral (spectral clustering) :

Lien avec le partitionnement spectral / Travaux futurs

- Liens avec les techniques modernes de partionnement spectral (spectral clustering) :
  - Point de vue graphe de similarités (sparse).

Lien avec le partitionnement spectral / Travaux futurs

- Liens avec les techniques modernes de partionnement spectral (spectral clustering) :
  - Point de vue graphe de similarités (sparse).
  - Utilisation de la matrice Laplacienne du graphe et de son spectre.

Lien avec le partitionnement spectral / Travaux futurs

- Liens avec les techniques modernes de partionnement spectral (spectral clustering) :
  - Point de vue graphe de similarités (sparse).
  - Utilisation de la matrice Laplacienne du graphe et de son spectre.
  - Plongement des données dans un espace euclidien (des vecteurs propres de la matrice Laplacienne).

Lien avec le partitionnement spectral / Travaux futurs

- Liens avec les techniques modernes de partionnement spectral (spectral clustering) :
  - Point de vue graphe de similarités (sparse).
  - Utilisation de la matrice Laplacienne du graphe et de son spectre.
  - Plongement des données dans un espace euclidien (des vecteurs propres de la matrice Laplacienne).
  - Utilisation de l'algorithme des k-moyennes dans cet espace.

Lien avec le partitionnement spectral / Travaux futurs

- Liens avec les techniques modernes de partionnement spectral (spectral clustering) :
  - Point de vue graphe de similarités (sparse).
  - Utilisation de la matrice Laplacienne du graphe et de son spectre.
  - Plongement des données dans un espace euclidien (des vecteurs propres de la matrice Laplacienne).
  - Utilisation de l'algorithme des k-moyennes dans cet espace.
- Quelques pistes pour des travaux futurs :

Lien avec le partitionnement spectral / Travaux futurs

- Liens avec les techniques modernes de partionnement spectral (spectral clustering) :
  - Point de vue graphe de similarités (sparse).
  - Utilisation de la matrice Laplacienne du graphe et de son spectre.
  - Plongement des données dans un espace euclidien (des vecteurs propres de la matrice Laplacienne).
  - Utilisation de l'algorithme des k-moyennes dans cet espace.
- Quelques pistes pour des travaux futurs :
  - Comment sparsifier $\mathbf{S}$? Selon les applications, le voisinage peut être donné par des informations externes (données images, spatiales, temporelles, ...)

Lien avec le partitionnement spectral / Travaux futurs

- Liens avec les techniques modernes de partionnement spectral (spectral clustering) :
  - Point de vue graphe de similarités (sparse).
  - Utilisation de la matrice Laplacienne du graphe et de son spectre.
  - Plongement des données dans un espace euclidien (des vecteurs propres de la matrice Laplacienne).
  - Utilisation de l'algorithme des k-moyennes dans cet espace.
- Quelques pistes pour des travaux futurs :
  - Comment sparsifier $\mathbf{S}$? Selon les applications, le voisinage peut être donné par des informations externes (données images, spatiales, temporelles, ...)
  - Peut-on utiliser d'autres opérateurs d'agrégation que les moyennes pondérées ?

Lien avec le partitionnement spectral / Travaux futurs

- Liens avec les techniques modernes de partionnement spectral (spectral clustering) :
  - Point de vue graphe de similarités (sparse).
  - Utilisation de la matrice Laplacienne du graphe et de son spectre.
  - Plongement des données dans un espace euclidien (des vecteurs propres de la matrice Laplacienne).
  - Utilisation de l'algorithme des k-moyennes dans cet espace.
- Quelques pistes pour des travaux futurs :
  - Comment sparsifier $\mathbf{S}$? Selon les applications, le voisinage peut être donné par des informations externes (données images, spatiales, temporelles, ...)
  - Peut-on utiliser d'autres opérateurs d'agrégation que les moyennes pondérées ?
  - Les méthodes invariantes par translation de la diagonale semblent mieux marcher que les autres notamment quand $\mathbf{S}$ est très sparse, pourquoi ?

Merci de votre attention !
Des questions ?