Algorithme des K-Médoïdes

Classification Automatique - Méthode de réallocation

Ricco RAKOTOMALALA
Université Lumière Lyon 2

PLAN

1. Classification automatique – La notion de médoïde
2. Algorithmes des K-Médoïdes
3. Le critère silhouette
4. Extensions
5. Conclusion
6. Bibliographie

La classification automatique

Constitution des groupes à partir des caractéristiques de proximité

Classification automatique

Typologie, apprentissage non-supervisé, clustering

Variables « actives », servent à la constitution des groupes.

Souvent (mais pas toujours) toutes quantitatives.

Modele	puissance	cylindree	vitesse	longueur	largeur	hauteur	poids	co2
PANDA	54	1108	150	354	159	154	860	135
TWINGO	60	1149	151	344	163	143	840	143
YARIS	65	998	155	364	166	150	880	134
CITRONC2	61	1124	158	367	166	147	932	141
CORSA	70	1248	165	384	165	144	1035	127
FIESTA	68	1399	164	392	168	144	1138	117
CLIO	100	1461	185	382	164	142	980	113
P1007	75	1360	165	374	169	161	1181	153
MODUS	113	1598	188	380	170	159	1170	163
MUSA	100	1910	179	399	170	169	1275	146
GOLF	75	1968	163	421	176	149	1217	143
MERC_A	140	1991	201	384	177	160	1340	141
AUDIA3	102	1595	185	421	177	143	1205	168
CITRONC4	138	1997	207	426	178	146	1381	142
AVENSIS	115	1995	195	463	176	148	1400	155
VECTRA	150	1910	217	460	180	146	1428	159
PASSAT	150	1781	221	471	175	147	1360	197
LAGUNA	165	1998	218	458	178	143	1320	196
MEGANECC	165	1998	225	436	178	141	1415	191
P407	136	1997	212	468	182	145	1415	194
P307CC	180	1997	225	435	176	143	1490	210
PTCRUISER	223	2429	200	429	171	154	1595	235
MONDEO	145	1999	215	474	194	143	1378	189
MAZDARX8	231	1308	235	443	177	134	1390	284
VELSATIS	150	2188	200	486	186	158	1735	188
CITRONC5	210	2496	230	475	178	148	1589	238
P607	204	2721	230	491	184	145	1723	223
MERC_E	204	3222	243	482	183	146	1735	183
ALFA 156	250	3179	250	443	175	141	1410	287
BMW530	231	2979	250	485	185	147	1495	231

Objectif de l'étude : Identifier les catégories (groupes) de voitures « similaires » (c.-à-d. qui se ressemblent au regard de leurs propriétés)

Objectif : identifier des groupes d'observations ayant des caractéristiques similaires (ex. comportement d'achats de clients, caractère « polluant » de véhicules, etc.)

On veut que :

(1) Les individus dans un même groupe se ressemblent le plus possible

(2) Les individus dans des groupes différents se démarquent le plus possible

Pourquoi ?

→ Identifier des structures sous-jacentes dans les données

→ Résumer des comportements

→ Affecter de nouveaux individus à des catégories

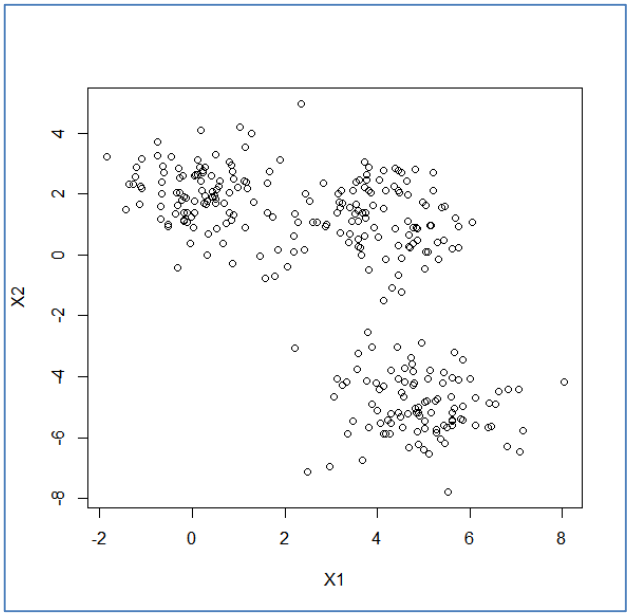
→ Identifier les cas totalement atypiques

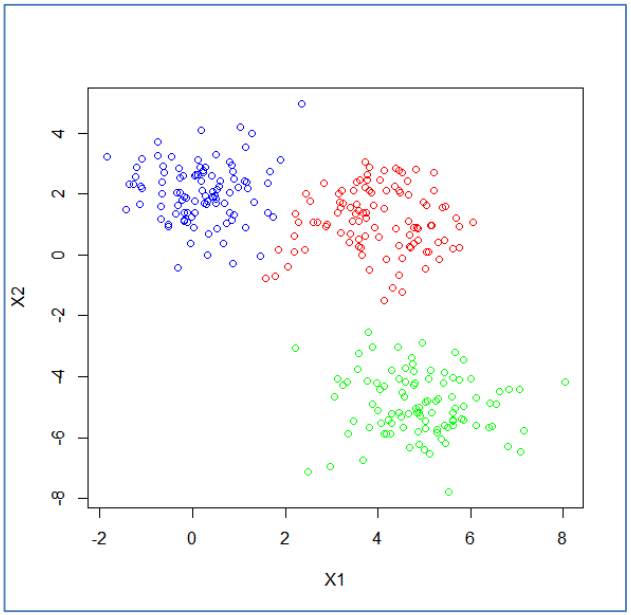
Classification automatique

Illustration graphique dans le plan

On « devine » les amas de points dans l'espace de représentation.

L'algorithme de classification automatique se charge de mettre en évidence les groupes « naturels » c.-à-d. qui se démarquent significativement les uns des autres.





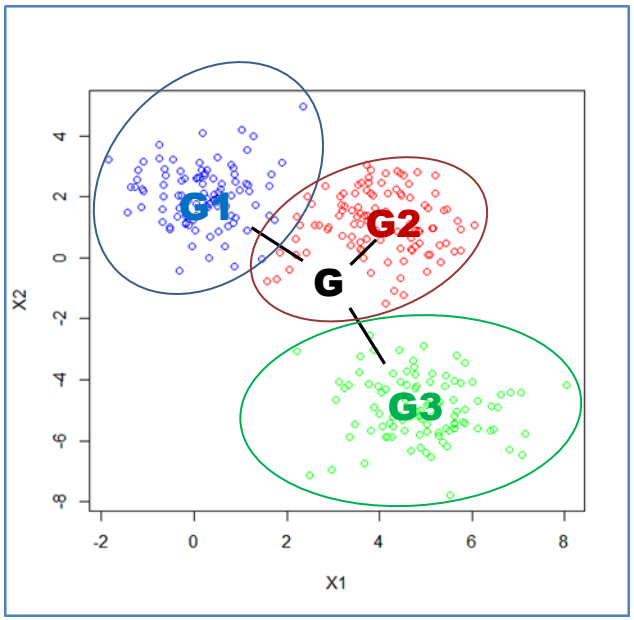
2 questions clés

1. Combien de groupes ?
2. Délimitation de ces groupes par le calcul

Caractérisation de la partition

Inertie intra-classes W

Donner un rôle crucial aux centres de classes



Remarque : les points étant rattachés à un groupe selon leur proximité avec le barycentre associé, les classes ont tendance à être convexes.

Relation fondamentale (Théorème d'Huygens)

$$\text{Inertie totale} = \text{Inertie inter - classes} + \text{Inertie intra - classe}$$

$$T = B + W$$

$$\sum_{i=1}^{n} d^2(i,G) = \sum_{k=1}^{K} n_k d^2(G_k,G) + \sum_{k=1}^{K}\sum_{i=1}^{n_k} d^2(i,G_k)$$

Dispersion des barycentres conditionnels autour du barycentre global. Indicateur de séparabilité des classes.

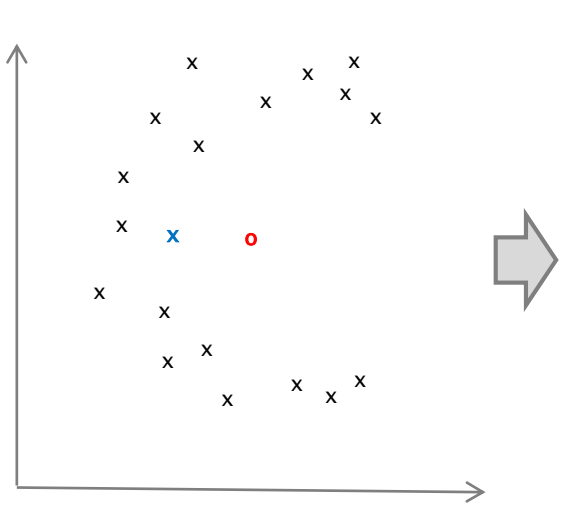
Dispersion à l'intérieur de chaque groupe. Indicateur de compacité des classes.

d() est une mesure de distance caractérisant les proximités entre les individus, distance euclidienne ou euclidienne pondérée par l'inverse de la variance (*attention aux points aberrants*)

Un objectif possible de la classification automatique serait de minimiser l'inertie intra-classes W, à nombre de classes K fixé.

La notion de médoïde

Point représentatif d'une classe



Le barycentre (**o**) peut être complètement artificiel, ne correspondant pas à la réalité des données.

On préfèrera dans certains cas la notion de médoïde (**x**), un point qui existe réellement et qui correspond à l'observation qui minimise sa distance avec l'ensemble des autres points.

$$M = \arg\min_{m} \sum_{i=1}^{n} d(i,m)$$

m = 1, …, n ; chaque point est candidat pour être médoïde.

$$E = \sum_{k=1}^{K} \sum_{i=1}^{n_k} d(i, M_k)$$

Peut faire office de mesure de qualité de la partition, à la place de l'inertie intra-classes W.

$$d(i,i') = \sum_{j=1}^{p} \left| x_{ij} - x_{i'j} \right|$$

Nous ne sommes plus cantonnés à la distance euclidienne, la distance de Manhattan (réduit considérablement l'impact des points atypiques) ou autres peuvent faire l'affaire.

Classification par partition

Les méthodes de réallocation

Principales caractéristiques

- Fixer a priori le nombre de classes K
- Définir une partition de départ des données
- **Réallocation.** Déplacer les objets (observations) d'un groupe à l'autre pour obtenir une partition meilleure
- L'objectif (implicite ou explicite) est d'optimiser une mesure d'évaluation globale de la partition
- Fournit une partition unique des données

Mais peut être évolutive en fonction d'autres paramètres telle que le diamètre maximum des classes. Reste un problème ouvert souvent.

Souvent de manière aléatoire. Mais peut également démarrer à partir d'une autre méthode de partition ou s'appuyer sur des considérations de distances entre les individus (ex. les K individus les moins éloignés de l'ensemble des autres).

En faisant passer tous les individus, ou encore en tentant des échanges (plus ou moins) aléatoires entre les groupes.

La mesure **E** sera utilisée.

On a une solution unique à K fixé. Et non pas une hiérarchie de partitions comme en CAH par ex.

Algorithme K-Médoïdes

Plusieurs approches possibles

Algorithme K-Médoïde

Una variante directement dérivée des K-Means

Il devient nécessaire de calculer la matrice des distances entres individus pris deux à deux d(i,i'), i,i' = 1,…,n

Algorithme particulièrement simple

Entrée : X (n obs., p variables), K #classes

Initialiser K médoïdes M_k

REPETER

Allocation. Affecter chaque individu à la classe dont le médoïde est le plus proche

Représentation. Recalculer les médoïdes des classes à partir des individus rattachés

JUSQU'À Convergence

Sortie : Une partition des individus caractérisée par les K médoïdes de classes M_k

Peut être K individus choisis au hasard. Ou encore, K points les moins distants des autres.

La distance entre les points pris deux à deux étant calculée au préalable, il n'est plus nécessaire de revenir sur les données.

Forcément, la dispersion E_k à l'intérieur de la classe C_k diminue (au pire reste stable)

Nombre d'itérations fixé
Ou encore lorsque E ne diminue plus
Ou lorsque les médoïdes M_k sont stables

Le processus minimise implicitement le critère global **E**

La complexité de calcul est particulièrement dissuasive !

Algorithme PAM

Partitioning around medoid (Kaufman & Rousseeuw, 1987)

Ici aussi, il est nécessaire de calculer la matrice des distances entres individus pris deux à deux d(i,i').

Entrée : X (n obs., p variables), K #classes

Initialiser K médoïdes M_k ← K points pris au hasard dans la base.

REPETER

Affecter chaque individu à la classe dont le médoïde est le plus proche

Pour chaque médoïde M_k

Sélectionner au hasard un point i non médoïde

Vérifier que le critère E diminue si on échange leur rôle. Si oui, on entérine la modification c.-à-d. i devient médoïde M_k de la classe C_k

JUSQU'A Aucun échange effectué

Sortie : Une partition des individus caractérisée par les K médoïdes de classes M_k

Phase **BUILD**

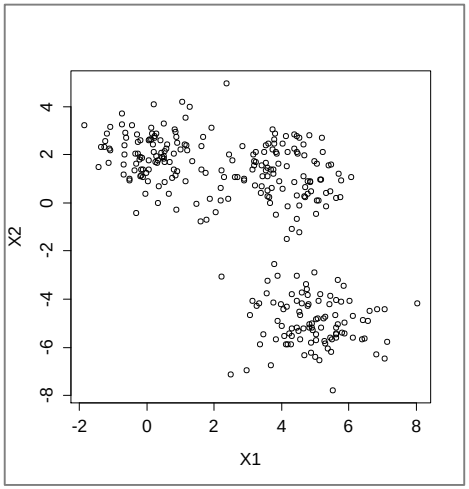
Phase **SWAP**

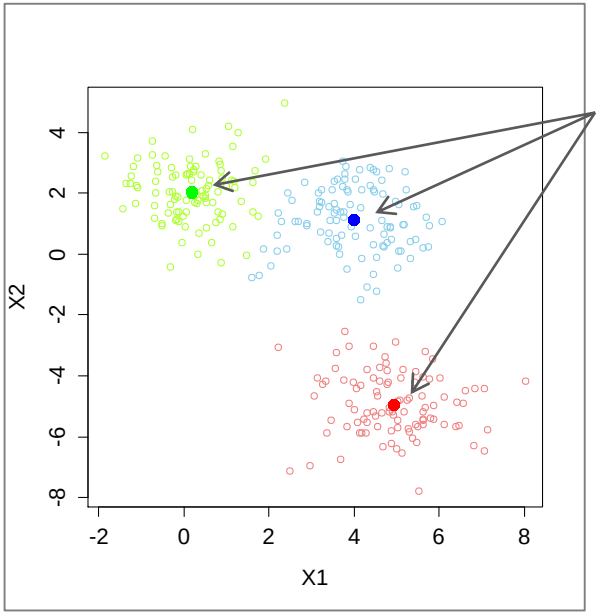
Voir un exemple pas à pas sur :
https://en.wikipedia.org/wiki/K-medoids

La complexité de calcul reste très importante

Algorithme PAM

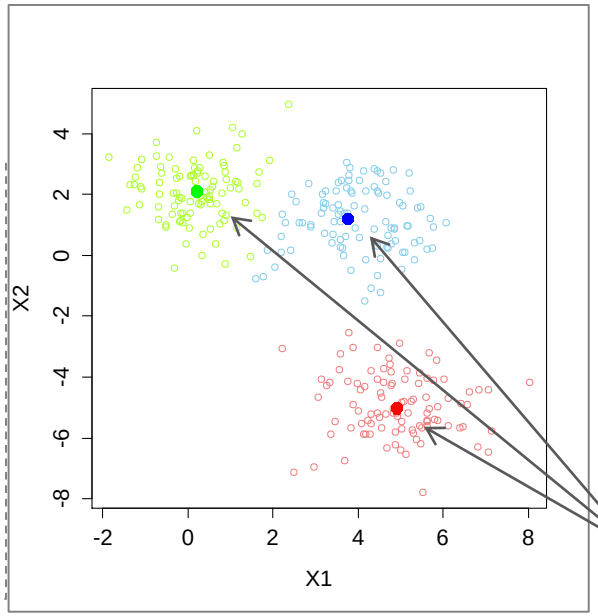
PAM vs. K-Means sur l'exemple fictif





Barycentres des classes

K-Means



PAM

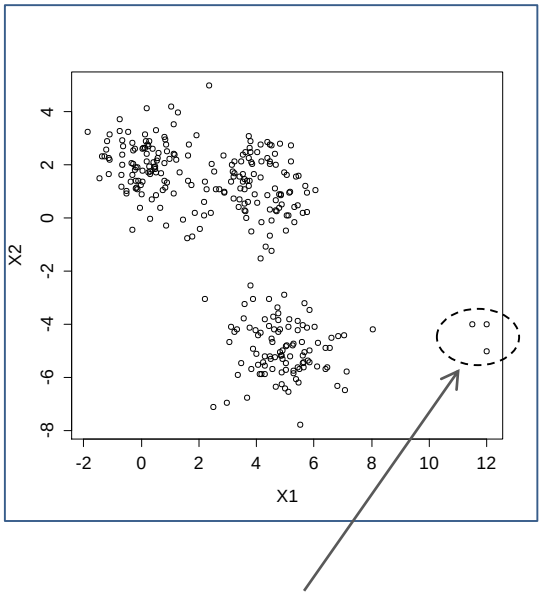
Les « pâtés » étant convexes, générés avec une loi normale, les médoïdes correspondent quasiment aux barycentres.

Médoïdes des classes

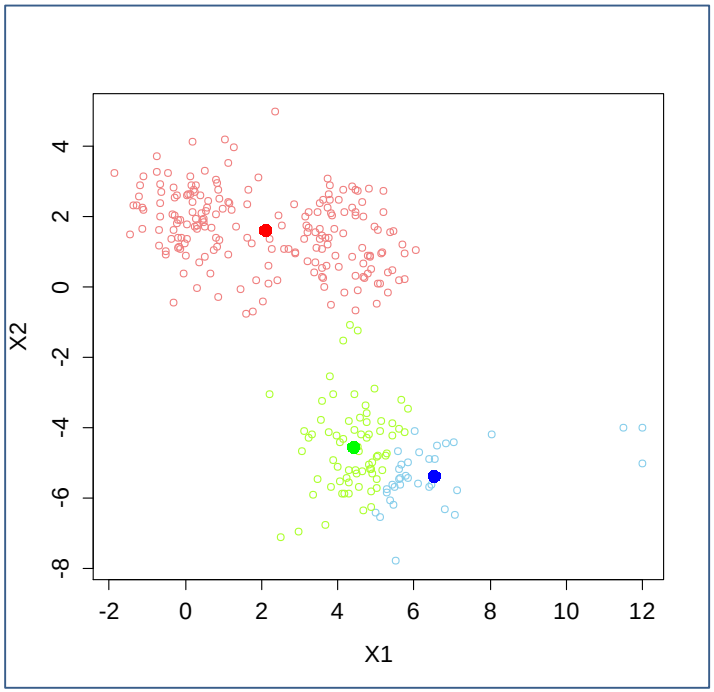
```
> library(cluster)
> res <- pam(X,3,FALSE,"euclidean")
> print(res)

> plot(X[,1],X[,2],type="p",xlab="X1",
ylab="X2",col=c("lightcoral","skyblue","greenyel
low")[res$clustering])
> points(res$medoids[,1],res$medoids[,2],
cex=1.5,pch=16,col=c("red","blue","green")[1:3])
```

Algorithme PAM

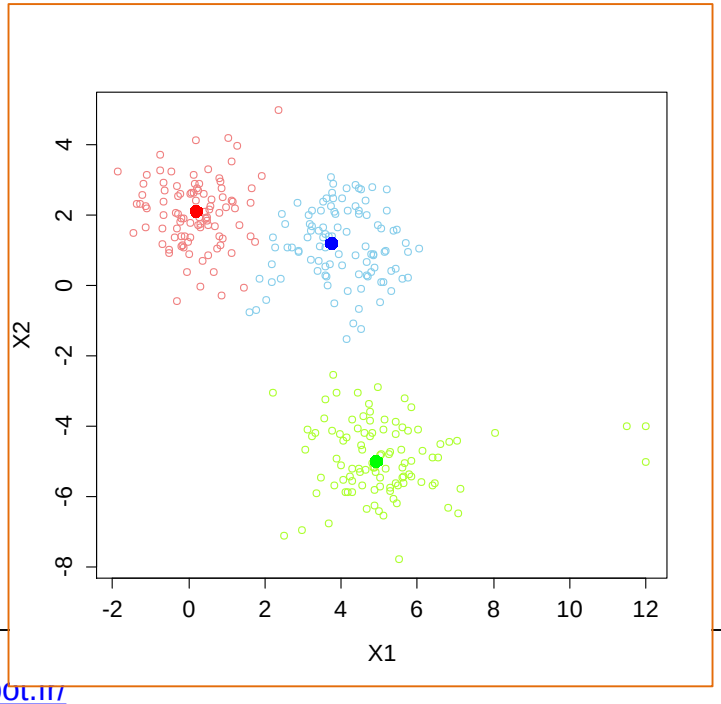
PAM vs. K-Means sur l'exemple fictif avec points atypiques



Pas bon ça. Source de problèmes généralement.



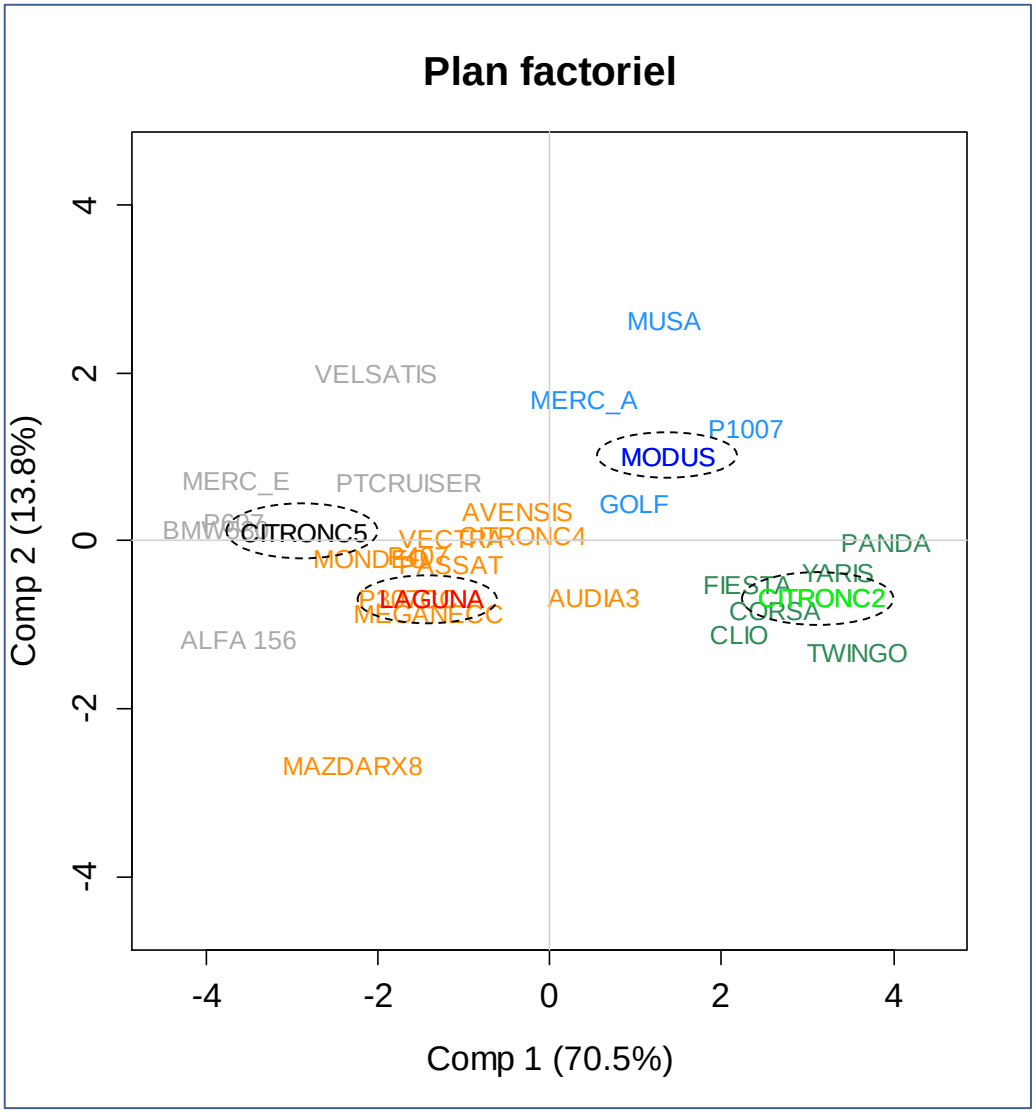
K-Means est susceptible d'être faussé.



PAM reste imperturbable. Les médoïdes sont placés judicieusement.

Algorithme PAM

PAM sur l'exemple des voitures



Premier plan factoriel de l'analyse en composantes principales

PAM

Appartenance aux classes des individus. On distingue les médoïdes pour chaque classe.

Algorithme CLARA

Clustering **L**arge **A**pplications (Kaufman & Rousseeuw, 1990)

Introduire la notion d'échantillonnage.

Entrée : X (n obs., p variables), K #classes

Construire S échantillons de taille η (η << n)

Appliquer PAM sur chaque échantillon → S vecteurs de médoïdes

Pour chaque vecteur de médoïdes

    Faire passer l'ensemble des observations

    Mesure la qualité de partition E

Retenir la configuration qui minimise E

Sortie : Une partition des individus caractérisée par les K médoïdes de classes M_k

En pratique : S = 5 et η = 40 +2 x K s'avèrent suffisants [*paramétrage par défaut de clara() dans le package « cluster » de R*].

Un seul passage sur les données suffit pour évaluer l'ensemble des configurations.

Possibilité de traiter des grandes bases

L'algorithme est fortement dépendant de la taille et de la représentativité des échantillons

Algorithme CLARA

Exemple sur les données « waveform » (Breiman et al., 1984)

Les données sont générées artificiellement. On connaît la « vraie » classe d'appartenance des individus.

21 descripteurs

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16	V17	V18	V19	V20	V21	CLASSE
-0.29	-2.24	-0.65	0.73	-0.15	1.37	3.21	1.31	1.94	2.06	1.86	2.67	1.42	4.4	3.99	4.17	1.57	1.28	2.34	2.32	-1.17	A
-1.82	1.89	1.1	0.76	2.42	4.59	3.54	3.55	3.07	4.81	1.74	1.65	2.31	2.64	2.73	2.2	1.85	0.33	0.04	-0.85	1.03	A
1.11	-0.42	1.4	-0.27	0.12	2.35	5.86	3.73	4.42	3.72	2.67	2.27	0.44	1.58	-0.02	2.48	0.58	1.04	0.46	1.55	-0.39	B
-1.57	0.52	0.55	1.67	4.56	2.15	0.04	5.24	2.94	1.15	0.48	1.64	0.2	0.26	1.37	3.03	2.03	1.28	0.53	1.07	0.23	A
-0.72	-0.44	-1.02	-0.49	-0.63	0.92	2.42	2.81	4.03	4.33	6.45	5.84	3.88	3.77	1.41	1.32	0.06	-1.22	0.28	-1.65	-0.42	C

30.000 obs.

A la sortie, les deux partitions proposées sont quasi-équivalentes. Le gain en temps de calcul est monumental

PAM : 443 sec. (+ de 7 min)

CLARA : 0.04 sec.

		CLARA		
		C1	C2	C3
PAM	A	9362	485	249
	B	2	9147	1277
	C	852	153	8473

V de Cramer = 0.85

En validation externe (comparaison par rapport à la vraie CLASSE), les différentes méthodes sont proches.

Croisement groupes affectés vs. CLASSE.

V de Cramer. PAM, CLARA, K-Means ≈ 0.5

Les 3 méthodes rencontrent les mêmes difficultés sur ces données.

Le critère silhouette

Outil pour la détection du bon nombre de classes

Critère silhouette

Degré d'appartenance à sa classe d'un individu

Rousseeuw (1987) propose une mesure d'évaluation des partitions non-dépendante du nombre de classes : le score silhouette.

$$a(i) = \frac{1}{n_a - 1} \sum_{\substack{i'=1 \\ i' \neq i}}^{n_a} d(i,i')$$

Moyenne des distances du point i avec l'ensemble des points de sa classe d'appartenance C_a dont l'effectif est n_a.

$$d(i,C_k) = \frac{1}{n_k} \sum_{i'=1}^{n_k} d(i,i')$$

Moyenne des distances du point i avec l'ensemble des points d'une classe C_k – autre que C_a - dont l'effectif est n_k.

$$b(i) = \min_{k \neq a} d(i,C_k)$$

Distance à la classe la plus proche au sens de $d(i,C_k)$

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}$$

Indique le degré d'appartenance à sa classe de l'individu en confrontant la distance moyenne à ses congénères avec la distance moyenne à la classe la plus voisine. s(i) est indépendant de K – nombre de classes – parce qu'on ne considère que la distance au voisin le plus proche !

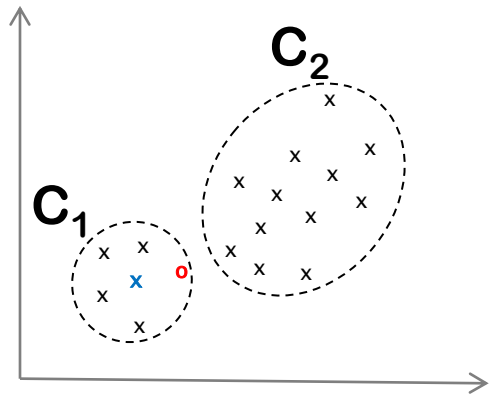
s(i) → 1 : le point est bien positionné dans sa classe

s(i) ≈ 0 : le point est aussi proche de ses congénères qu'avec les autres

s(i) → -1 : le point est plus proche des autres que de ses congénères

Critère silhouette

Evaluation des classes et des partitions



s(x) > s(o): (1) parce que « x » est placé en position centrale (c'est le médoïde de la classe) au sein de C_1 ; (2) parce que « o » se rapproche de la classe C_2.

$$\bar{s}_k = \frac{1}{n_k} \sum_{i \in C_k} s(i)$$

Caractérise à la fois la compacité du groupe C_k, et de son écartement (séparabilité) par rapport aux autres classes.

$$S_K = \frac{1}{n} \sum_{k=1}^{K} n_k \times \bar{s}_k$$

Caractérise la qualité globale de la partition en K classes. De manière empirique :

S ∈ [0,71 ; 1] : une structuration forte a été découverte

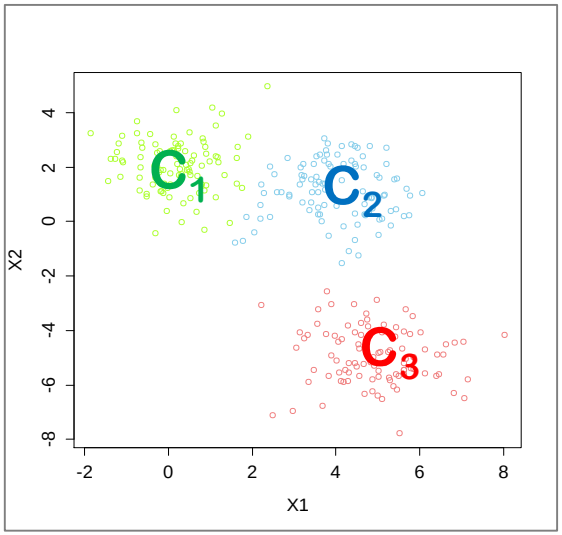
S ∈ [0,51 ; 0.70] : une structuration raisonnable existe

S ∈ [0,26 ; 0.50] : la structuration est faible, sujette à caution

S ∈ [0 ; 0.25] : pas de structuration des données

Critère silhouette

Outil pour le choix du nombre de partitions



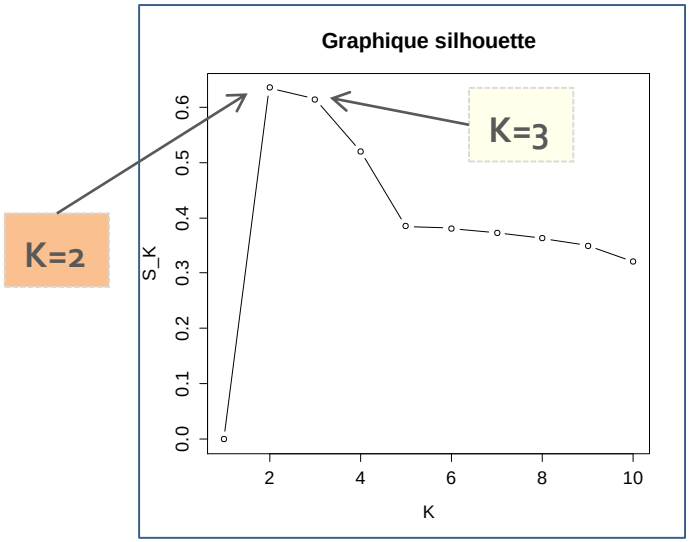
Choix du nombre adéquat de classes, l'arlésienne de la classification automatique. Le critère silhouette étant indépendant du nombre de classes, Il suffit de choisir la valeur K qui le maximise.

$$\bar{s}_1 = 0.60$$
$$\bar{s}_2 = 0.53$$
$$\bar{s}_3 = 0.70$$

La classe C3 est celle qui se démarque le plus des autres.

$$S_{K=3} = 0.61$$

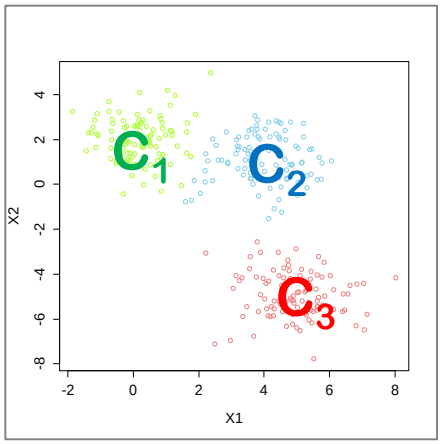
Qualité globale de la partition en K = 3 classes.



Tester les différentes valeurs de K et identifier les meilleurs configurations (partitions en K classes). Ici K= 2 (S_2 = 0.63) et (S_3 = 0.61) sont en concurrence. L'indicateur préfère la solution en K = 2 classes. Est-ce vraiment étonnant ?

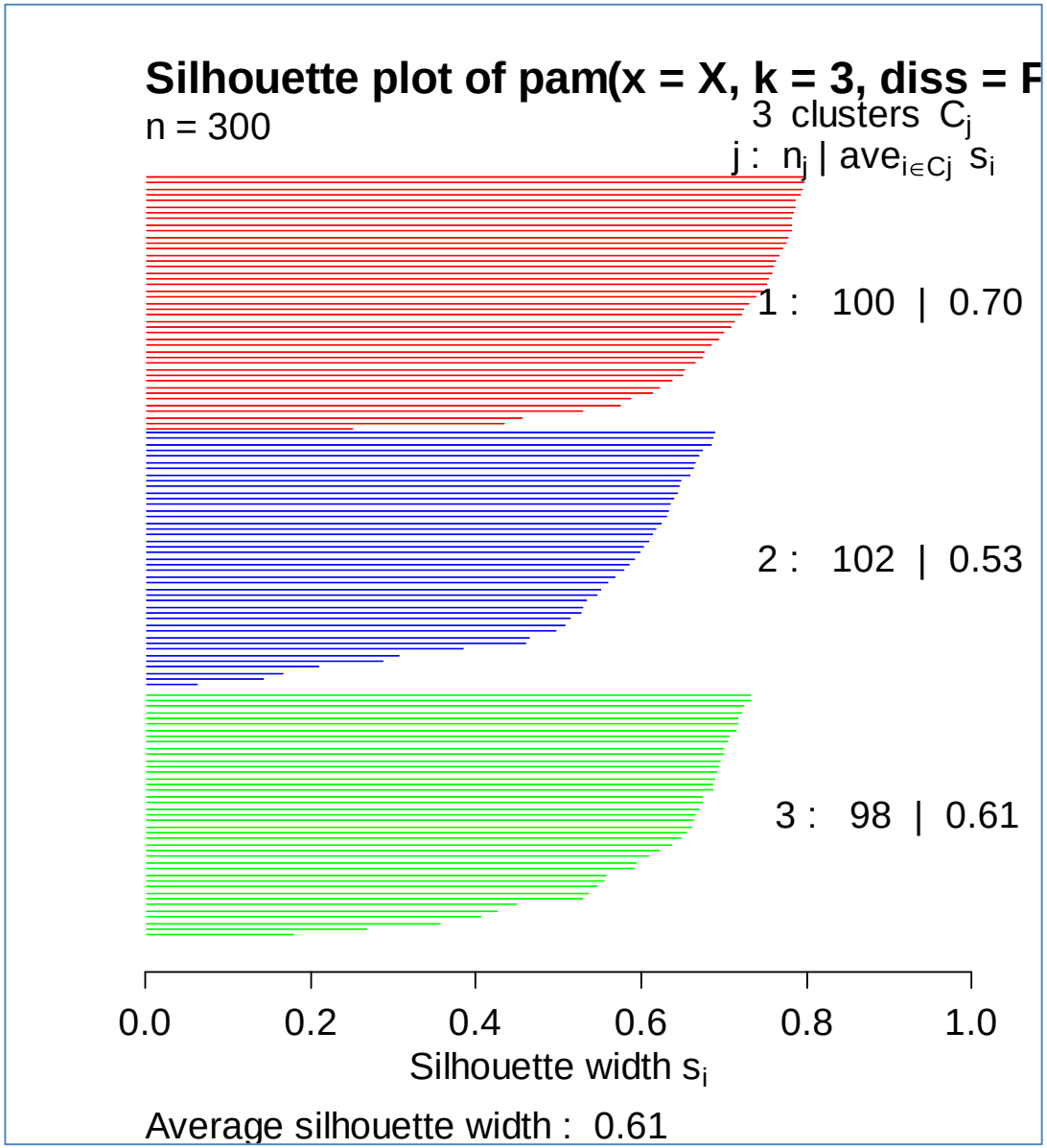
Graphique silhouette

Evaluation des classes et des partitions



Les packages de calcul les plus connus proposent la représentation graphique connue sous l'appellation « Graphique silhouette » (silhouette plot).

On observe d'une part la qualité globale des classes (le « bloc » présente-t-il en moyenne une valeur S_k plus élevée que les autres), mais aussi l'homogénéité de la situation des individus dans les classes. *Par ex. pour la classe « rouge », très peu d'individus ont une mauvaise valeur de silhouette s(i)*



Extensions

- Les algorithmes peuvent s'appliquer au traitement des variables qualitatives en redéfinissant la distance (distance 0/1 ou encore distance du Khi-2).
- La démarche « tandem analysis » (analyse factorielle + classification) est possible également, il est dès lors possible de traiter les données mixtes.
- L'extension à la classification floue est possible (algorithme « fanny »).

- L'extension à la classification de variables est naturelle : r^2 peut faire office de mesure de similarité, $(1 - r^2)$ de mesure de distance (ou respectivement r et (1-r) si l'on souhaite tenir compte du signe de la relation).

Conclusion

- Les techniques de partitionnement par réallocation présentent l'avantage de la simplicité.

- Les méthodes des K-Médoïdes permettent de dépasser l'écueil des points atypiques, d'une part en redéfinissant la notion de point représentatif de la classe, d'autre part en préconisant l'utilisation de distances qui y moins sensibles (ex. Manhattan).
- PAM (Partitioning Aroung Medoids) est une implémentation populaire de l'approche. Mais la nécessité de calculer les distances entre les individus pris deux à deux pénalise la scalabilité.
- On améliore la capacité à traiter les grandes bases de PAM en travaillant sur des échantillons (approche CLARA).

- Un critère d'évaluation des partitions insensible au nombre de classes accompagne la méthode : le critère silhouette.
- Nous pouvons le mettre à contribution pour identifier la meilleure configuration via le graphique silhouette. Mais ça reste une heuristique, le choix des solutions doit être appuyé par l'interprétation.

Bibliographie

Ouvrages et articles

Gan G., Ma C., Wu J., « Data Clustering – Theory, Algorithms and Applications », SIAM, 2007.

Nakache J.P., Confais J., « Approche pragmatique de la classification », Technip, 2005.

R. Rakotomalala, « Clustering : méthode des centres mobiles », octobre 2016.

Struyf A., Hubert M., Rousseeuw P., « Clustering in an Object-Oriented Environnement », Journal of Statistical Software, 1(4), 1997.

Wikipédia, « k-medoids », consulté le 19/10/2016.

Wikipédia, « Silhouette (clustering) », consulté le 19/20/2016.

Tutoriels

« Classification automatique sous R », octobre 2015.

« Classification automatique sous Python », mars 2016.

STHDA, « Partitioning cluster analysis : Quick start guide ».