La Classification

1 Introduction

Les méthodes de classification cherchent à regrouper un ensemble de ***n*** individus en un nombre restreint ***k*** de classes homogènes. Pour ce faire, on peut distinguer deux types de classification :

La classification manuelle (subjective) : les classes sont séparées visuellement, via des experts humains, ce qui est impossible si on a plusieurs variables, à moins de passer par une Analyse en Composantes Principales (ACP). Toutefois, dans ce cas on ne tient pas compte de l'inertie totale (perte d'information).

La classification automatique (objective) : les classes sont obtenues automatiquement, via des algorithmes. Dans ce cas, on distingue deux types de méthodes : la classification supervisée (machine learning), à condition d'avoir à la disposition un ensemble de données déjà classifiées (labeled dataset), et la classification non supervisée. Pour la classification non supervisée, on distingue les méthodes non hiérarchiques, dites aussi itératives, qui décomposent l'ensemble d'individus en un nombre fixé de classes (donc, on doit relancer le code pour classifier la même population d'individus en un autre nombre de classes) et les méthodes hiérarchiques, qui décomposent l'ensemble d'individus en une arborescence de groupes, indépendamment du nombre de classes (le nombre de classes peut ne pas être fixé d'avance).

Notation : On dispose de ***n*** individus X_1, …, X_n munis de poids m_1, …, m_n qui ont des caractéristiques représentant des valeurs des variables V_1, … , V_p.

Dans la majorité des cas, ce qui est le cas le long de ce chapitre, la distance utilisée entre deux individus X_i et X_j est la distance Euclidienne, sachant que d'autres métriques peuvent être appliquées. En effet, le carré de la distance Euclidienne $d^2(X_i, X_j)$ entre deux individus X_i et X_j est défini comme suit :

$$\sum_{k=1}^{p}(X_{ik} - X_{jk})^2$$

Le centre de gravité G du nuage de points est défini par (sachant que si les données sont centrées, ce qui est le cas dans la majorité des cas, G est l'origine du repère.) :

$$G = \frac{\sum_{i=1}^{n} m_i X_i}{\sum_{i=1}^{n} m_i}$$

2 Classification non supervisée

Le but est de regrouper ***n*** individus en ***k*** classes (***k*** << ***n)*** tels que les individus d'une même classe soient les plus semblables que possible (homogénéité) **et** les classes sont bien séparées (hétérogénéité).

Homogénéité : Soient C_1, … , C_k les ***k*** classes. Chaque classe C_j est constituée de t_j individus. Dans chaque classe, les individus doivent être le plus possible moins dispersés. C'est-à-dire, ayant G_j le centre de gravité de la classe C_j, on cherche à minimiser la quantité suivante :

$$I(C_j) = \sum_{i=1}^{t_j} m_i d^2(X_i, G_j)$$

Ainsi, afin d'aboutir à ce critère d'homogénéité, on doit minimiser alors l'inertie intra-classes I_W (*'within'*) pour tout le nuage, qui est définie comme suit :

$$I_W = \sum_{j=1}^{k} I(C_j)$$

Hétérogénéité : Les classes doivent être au maximum les plus distantes. Ainsi, on définit l'inertie inter-classes I_B (*'Between'*), qu'on cherche à maximiser, par :

$$I_B = \sum_{j=1}^{k} P_j d^2(G_j, G)$$

avec P_j est la somme des poids des individus de la classe C_j.

Donc le but est de maximiser I_B **et** de minimiser I_W. Or d'après le théorème d'Hygens, on a :

$$I_W + I_B = cste = I(G)$$

Donc, maximiser I_B revient implicitement à minimiser I_W, et vice versa. Ainsi, on peut se contenter à chercher la solution qui permet de maximiser I_B **ou** de minimiser I_W. Toutefois, si on vise à trouver la meilleure solution possible, on doit tester touts les solutions possibles (dont le nombre est égal à C_n^k) et de choisir celle qui maximise I_B **ou** qui minimise I_W. Pour cela, on cherche souvent à trouver une solution optimale via des heuristiques.

2.1 Classification Itérative

Les méthodes de classification non hiérarchique, ou itérative, aboutissant à la décomposition de l'ensemble de tous les individus en ***k*** classes disjointes. Le nombre de classes est fixé au préalable.

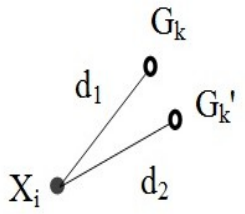
Méthode 1 : Méthode des centres mobiles ou k-means

1. Input : n individus de masses m_i avec $i = 1, \dots, n$ et k est le nombre de classes (fixé);
2. On choisit k points, illustrant les centres de gravités des k classes, au hasard $G_1, \dots, G_k$;
3. Pour chaque point, on calcule sa distance avec $G_1, \dots, G_k$. Le point est regroupé avec celui dont la distance est minimale.
4. On constitue, donc, les k classes et on détermine les nouveaux centres de gravités $G_1, \dots, G_k$ de ces classes. On calcule $I^{(t)}$ à l'itération courante t.
5. On réitère les deux étapes précédentes jusqu'à ce que T_W devient stable (c'est à dire $|I^{(t+1)} - I^{(t)}| < \varepsilon$).

Remarque : $I_W^{(t+1)} \leq I_W^{(t)}$ ce qui garantie la convergence de l'algorithme.

Méthode 2 : Méthode des nuées dynamiques

Cette méthode consiste à représenter la classe non pas par un centre de gravité mais par un noyau de q éléments (exemple $k = 3$ et $q = 2$), afin de minimiser la dépendance des résultats finaux de l'étape d'initialisatio, souvent faire aléatoirement.
On choisit 3 couples de points et on calcule pour chaque point sa distance avec le noyau. La distance entre un point X_i et un noyau $\{G_k, G_k'\}$ peut être définie comme $d_1 + d_2$ ou $\min(d_1, d_2)$ ou $\max(d_1, d_2)$...



À une partition on associe le noyau des q points les plus représentatifs de chaque classe (par exemple, le premier point c'est le centre de gravité G_j qui réalise le minimum de :

$$\sum_{i=1}^{t_j} d(X_i, y_j)$$

Le deuxième c'est l'individu qui minimise la même fonction objective suivante, autre que Gj, …).

$$\sum_{i=1}^{t_j} d(X_i, y_j)$$

Remarque : L'avantage de ces deux méthodes réside dans la rapidité du calcul. Par contre, l'inconvénient majeur de ces méthodes, notamment pour la méthode k-means, réside dans la partition finale qui dépend des noyaux (ou centres) choisies au départ et donc on risque d'avoir des classes vides.

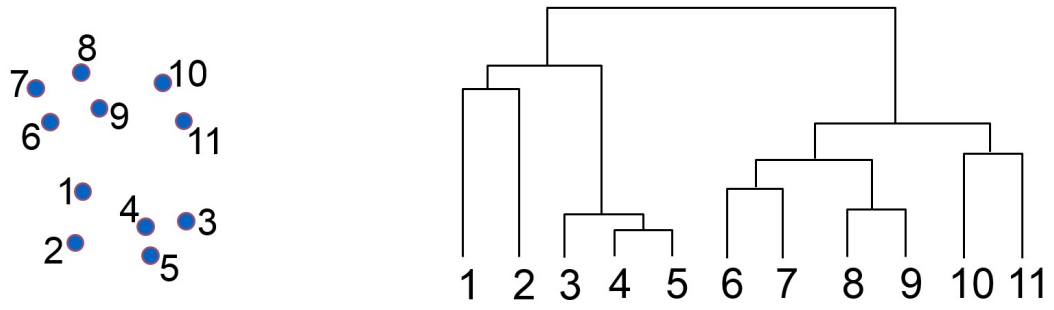
2.1 Classification Hiérarchique

Les méthodes de classification hiérarchique identifient une classification optimale d'une manière récursive. Le résultat est un arbre qui illustre des classifications de la population de n individus en (n-2) partitions : soit de 1 classe, 2 classes, …, n-1 classes, n classes (pour le cas de classification hiérarchique descendante) ; soit en n classes, n-1 classes, …., 2 classes, 1 class (pour le cas de classification hiérarchique ascendante). Ceci a l'avantage de trouver touts les partitions indépendamment du nombre de classes, et même de trouver le nombre de classes qui permet une solution optimale, ce qui est très utile pour les cas d'indisponibilité de nombres de classes (clustering/regroupement).

peremettent d'ifen, ou itérative, aboutissant à la décomposition de l'ensemble de tous les individus en k classes disjointes. Le nombre de classes est fixé au préalable. Toutefois, l'inconvénient majeure de la classification hiérarchique réside dans le problème de propagation d'erreurs.

Classification Hiérarchique Ascendante (AHC)

Le principe est de construire une suite de partition en n classes, n-1 classe,…, 2 classes, 1 classe, qui sont emboitées les unes dans les autres. En effet, la partition en k classes est obtenue en regroupant à partir de la partition en k + 1 classes les deux classes les plus semblables. La suite des partitions obtenue est représentée sous forme d'un arbre de classification. à chaque niveau de l'arbre, on doit trouver les classes à fusionner selon un critère (le critère le plus utilisé est le critère de Ward).



En effet, de l'itération t à l'itération $t + 1$, l'inertie intra-classes I_W augmente et l'inertie inter-classes I_B diminue, or le but est de maximiser I_B (ou de minimiser I_W) pour une meilleure séparation entre les classes. Ainsi, on doit choisir les 2 classes qui permettent de minimiser la perte, soit en termes d'inertie intra-classes I_W ou d'inertie inter-classes I_B. Par exemple, le critère de Ward cherche la solution qui permet de minimiser la perte de I_B c.à.d. minimiser $(I_B^{(t)} - I_B^{t+1})$.

Exemple : Soit 4 classes A, B, C et D de l'itération t à $t + 1$, on va fusionner A et B dans ce cas :

$$I_B^{(t)} = P_A d^2(G_A, G) + P_B d^2(G_B, G) + P_C\, d^2(G_C, G) + P_D d^2(G_D, G)$$

Et

$$I_B^{(t+1)} = (P_A + P_B)\, d^2(G_{AB}, G) + P_C\, d^2(G_C, G) + P_D d^2(G_D, G)$$

Donc la perte provoquée par la fusion des deux classes A et B est égale à :

$$\delta(A, B) = I_B^{(t)} - I_B^{t+1} = P_A d^2(G_A, G) + P_B d^2(G_B, G) - P_A + P_B)\, d^2(G_{AB}, G)$$

Or $G_{AB} = (P_A G_A + P_B G_B)/(P_A + P_B)$

Donc,

$$\frac{P_A \cdot P_B}{P_A + P_B}\, d^2(G_A, G_B) = \delta(A, B)$$

Algorithme : D'une itération à une autre, on calcule toutes les pertes de I_B possibles en fusionnant deux classes quelconques et on fusionne par la suite les deux classes qui enregistrent la plus petite valeur de la perte δ.

Dans l'exemple précédent, au départ on a 4 classes, on calcule la matrice symétrique suivante :

	A	B	C	D
A	0	$\delta(A, B)$	$\delta(A, C)$	$\delta(A, D)$
B		0	$\delta(B, C)$	$\delta(B, D)$
C			0	$\delta(C, D)$
D				0

Si $\delta(A, B)$ minimal, on fusionne A et B et on recalcule la matrice symétrique suivante :

	AB	C	D
AB	0	$\delta(AB, C)$	$\delta(AB, D)$
C		0	$\delta(C, D)$
D			0

$$\text{avec } \delta(AB, C) = \frac{(P_A + P_C)\delta(A, C) + (P_B + P_C)\delta(B, C) - P_C\delta(A, B)}{P_A + P_B + P_C}$$

$$\text{ou } \delta(AB, C) = \frac{P_{AB} \cdot P_C}{P_{AB} + P_C}\, d^2(G_{AB}, G_C).$$

On répète la même chose jusqu'à obtenir la suite des partitions jusqu'à regrouper toute la population en une seule classe. La suite obtenue est souvent représentée sous la forme d'un arbre de classification, dite dendrogramme, et chaque niveau de cet arbre est appelé niveau d'agrégation et représente la perte d'inertie δ. Dans le cas d'indisponibilité de la valeur de c, le nombre de classes retenue est celui le plus petit possible pour lequel la perte de IB reste relativement faible.