Une Méthode de Catégorisation Multi-Echelle

A Multi-Scale Clustering Algorithm

Arnaud Ribert, Abdel Ennaji, Yves Lecourtier

P.S.I. Faculté des Sciences, Université de Rouen
76 821 Mont Saint Aignan Cédex, France
Arnaud.Ribert@univ-rouen.fr

Résumé

Nous proposons dans cet article une méthode originale de détermination du nombre et de la composition des agrégats (i.e. classes au sens non supervisé) présents dans une base de données à partir de l'analyse d'une hiérarchie indicée. Notre méthode est basée sur le principe d'une coupure multi-niveaux dans la hiérarchie permettant d'adapter l'échelle à laquelle considérer les données pour déterminer les agrégats. Pour cela, nous proposons un critère permettant de détecter la présence d'un agrégat unique dans un sous-arbre de la hiérarchie. Ceci permet d'aborder les configurations (très fréquentes dans la pratique) présentant des agrégats de densités variables. De plus, notre algorithme ne nécessite pas de retour sur les données, mais exploite uniquement l'information disponible dans la hiérarchie.

Abstract

This article deals with an original hierarchical clustering algorithm which automatically determines the number and composition of clusters in a database. The method is based upon a multi-level cutting in the hierarchy which allows one to deal with high density variations in the data (frequently encountered in real databases). The algorithm proceeds by an in depth exploration of the hierarchy, deciding a cutting when a single cluster is detected in the current sub-tree. This detection is performed by an original statistical criterion. Moreover, the clustering algorithm is particularly fast, since it does not require any computation involving original data, but exclusively exploits the information provided by the hierarchy.

1. Introduction

Nous proposons dans cet article une méthode originale de détermination du nombre et de la composition des agrégats présents dans un espace euclidien. Cette opération, appelée catégorisation, est souvent utilisée pour la segmentation d'image ainsi qu'en reconnaissance des formes. Les algorithmes de catégorisation les plus fréquemment rencontrés sont généralement des méthodes de partitionnement inspirées de la méthode des k-moyennes [11][7]. Le principal inconvénient de ces approches est la nécessité de connaître à l'avance le nombre d'agrégats ainsi que leur position approximative dans l'espace de représentation.

La méthode de catégorisation que nous proposons est basée sur l'analyse d'une hiérarchie indicée construite par Classification Ascendante Hiérarchique [3][2]. L'arbre ainsi obtenu présente la propriété de posséder des noeuds dont la hauteur (appelée indice) est proportionnelle à la dissemblance entre les groupes qu'ils réunissent. L'intérêt de cette approche réside dans le fait que l'analyse de la forme d'une hiérarchie peut permettre une catégorisation automatique des données considérées, sans a priori sur le nombre et la position des agrégats. La hiérarchie de la Figure 1 montre en effet sans ambiguïté la présence de six agrégats dans les données. Une étude comparative très complète, menée par Milligan et Cooper [12] a montré que de nombreuses méthodes ont été proposées pour exploiter ce potentiel [9][6]. La plupart d'entre elles sont, comme les méthodes de partitionnement, basées sur des critères de minimisation de la variance intra-agrégat et de maximisation de la variance inter-agrégat [5][4]. Elles nécessitent donc un retour sur les données. Ce point est un inconvénient non négligeable, car il implique un surcroît de calculs déjà très nombreux pour construire une hiérarchie indicée. De plus, il est connu que l'utilisation de la variance sur ce type de problème favorise la formation d'agrégats hyper-sphériques. Une analyse de données ne vérifiant pas cette hypothèse implicite est donc biaisée.

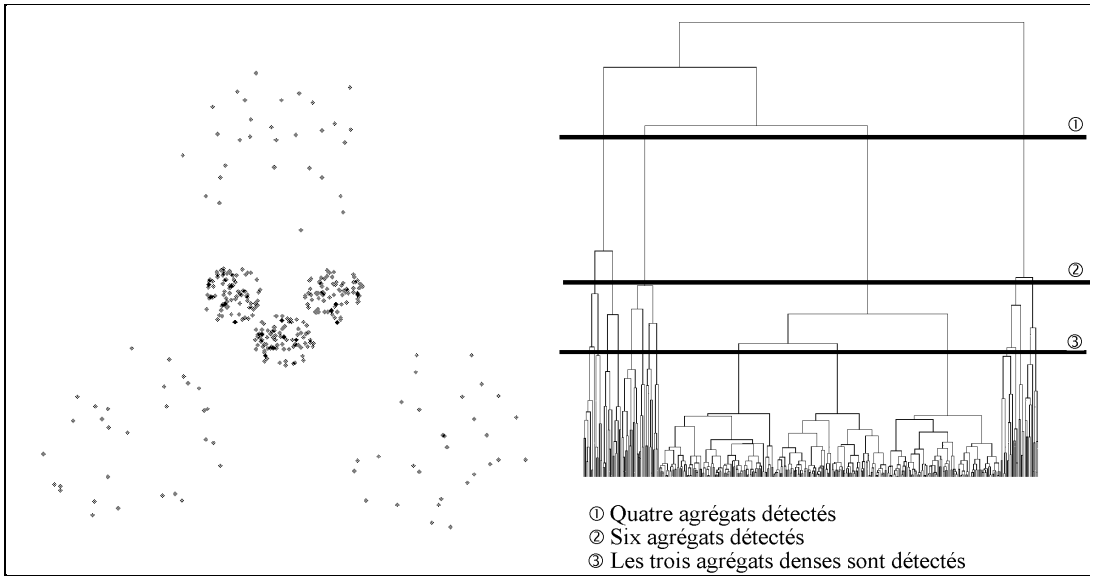

Figure 1 : Exemple d'échec de la coupure unique

Les méthodes recensées par Milligan et Cooper utilisent les hiérarchies indicées de la façon suivante : déterminer I agrégats dans les données en coupant la hiérarchie horizontalement au niveau du $(I-1)^{ième}$ rang de la liste des indices classés par ordre décroissant; puis juger de la validité de ces I agrégats à l'aide d'une mesure basée sur la variance. On considère que I est le nombre d'agrégats lorsque la mesure effectuée vérifie une certaine condition. Cependant, une telle procédure ne permet pas toujours de déterminer le bon nombre d'agrégats. Il s'avère en effet impossible de trouver la bonne répartition des données à l'aide d'une coupure unique lorsque les agrégats n'ont pas la même densité. Un exemple apparaît sur la Figure 1.

Cet exemple montre que trois situations peuvent être rencontrées lorsque l'on tente d'appliquer une coupure unique sur des données présentant de fortes variations de densité. La plupart des algorithmes classiques [12] détecterons vraisemblablement la présence de quatre agrégats dans les données. La structuration des données les plus denses est donc totalement ignorée. Si l'on souhaite mettre en évidence ces trois agrégats, la coupure unique conduit à fractionner les groupes les moins denses, révélant ainsi de façon erronée la présence de 17 agrégats. Enfin, si l'on considère que le nombre d'agrégats est connu (6), une coupure unique conduira de même à fractionner les agrégats les moins denses, sans pour autant faire apparaître les plus denses.

Il apparaît donc que la structure des données ne peut être mise en évidence avec une coupure unique. Cela prouve que couper une hiérarchie horizontalement pour en déduire la constitution de I agrégats est une simplification abusive du problème de partitionnement d'un ensemble. Finalement, il s'avère que cette stratégie n'est applicable que pour des cas particulièrement simples où la densité des points est quasiment constante.

Le retour systématique aux données et l'incapacité à traiter les différences de densité des algorithmes existants nous ont conduit à proposer une méthode exclusivement basée sur l'analyse de la hiérarchie indicée et utilisant non pas une, mais plusieurs coupures.

2. Musc : un algorithme de catégorisation multi-échelle

2. 1. Détermination de l'échelle maximale dans MUSC (MUlti-Scale Clustering)

Le problème soulevé par les différences de densité incite à introduire plusieurs points de coupure dans la hiérarchie indicée. L'exemple de la Figure 1 montre que l'ensemble des données doit tout d'abord être examiné d'un point de vue global, mais que certaines sous-parties bien structurées doivent être plus détaillées. L'algorithme que nous proposons vise à considérer les données à plusieurs échelles. Pour ce faire, nous définissons un critère de détection d'un unique agrégat dans un sous-arbre. Ainsi, si au cours du parcours en profondeur d'abord de la hiérarchie indicée, il s'avère que le sous-arbre courant ne représente qu'un agrégat, aucun raffinement supplémentaire n'est nécessaire.

Pour caractériser la forme d'une hiérarchie indicée, nous proposons d'établir un histogramme de ses indices, puis de calculer la variance des valeurs de cet histogramme (σ^2). La figure ci-dessous représente les histogrammes obtenus respectivement pour 1 et 6 nuages.

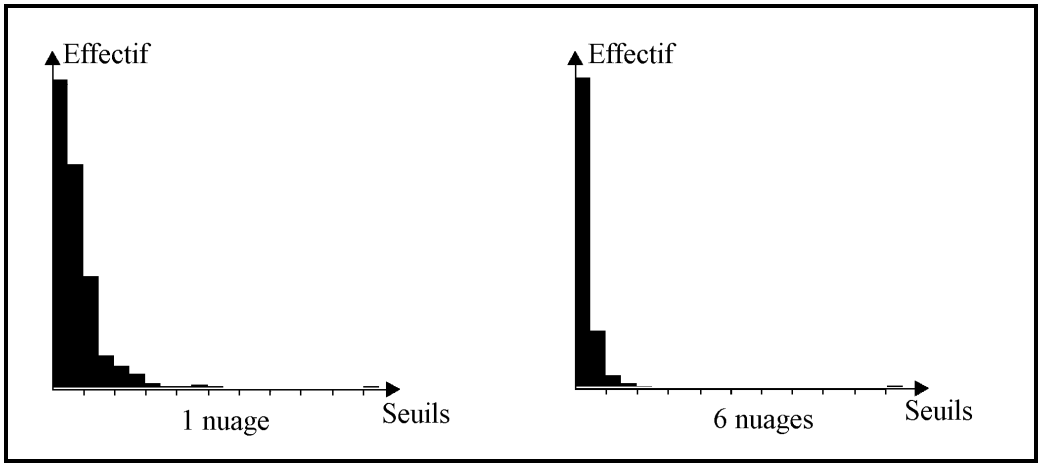

Figure 2 : Deux histogrammes associés à 1 et 6 nuages en 2D

On remarque que la valeur de σ^2 croît avec le nombre d'agrégats. La présence de plusieurs agrégats se traduit en effet par une augmentation de la proportion des indices de faibles valeurs dans la hiérarchie. La variance des valeurs de l'histogramme des indices d'une hiérarchie indicée peut donc révéler la présence d'un unique agrégat. Il faut cependant remarquer qu'une variance donnée traduit des écarts beaucoup plus importants lorsque la moyenne est faible que lorsqu'elle est forte. Nous normaliserons donc la variance des valeurs de l'histogramme par leur moyenne (μ).

L'utilisation de σ^2/μ impose de déterminer un seuil en deçà duquel on considère qu'un seul agrégat est présent dans un sous-arbre. Or, des tests effectués sur bases synthétiques ont montré qu'un seuil unique ne pouvait être déterminé. Il s'avère en particulier que la dimension de l'espace de représentation a une grande influence sur la forme des hiérarchies indicées et que la C.A.H. provoque une hausse artificielle de leurs seuils lorsque le nombre d'éléments pris en compte augmente. Nous avons donc construit une abaque (Figure 3) représentant σ^2/μ en fonction de la dimension de l'espace de représentation et du nombre d'éléments considérés pour des hiérarchies représentant un unique agrégat. Ceux-ci ont été générés de façon aléatoire selon une distribution uniforme dans un hypercube. Chaque point a donné lieu à la construction de 50 hiérarchies selon l'algorithme du lien moyen à l'aide de la formule de Lance-Williams [10].

Ainsi, si la valeur de σ^2/μ est supérieure à la valeur de l'abaque pour une configuration de points inconnue, c'est qu'il y a plus d'un agrégat dans le sous-arbre. Il faut donc explorer la hiérarchie plus en profondeur. Compte tenu de la variabilité en chaque point de l'abaque, nous n'en avons considéré que les maxima, au risque de ne pas descendre suffisamment dans la hiérarchie. Nous préférons en effet sous-estimer le nombre d'agrégats plutôt que de le surestimer afin d'éviter la multiplication de petits agrégats.

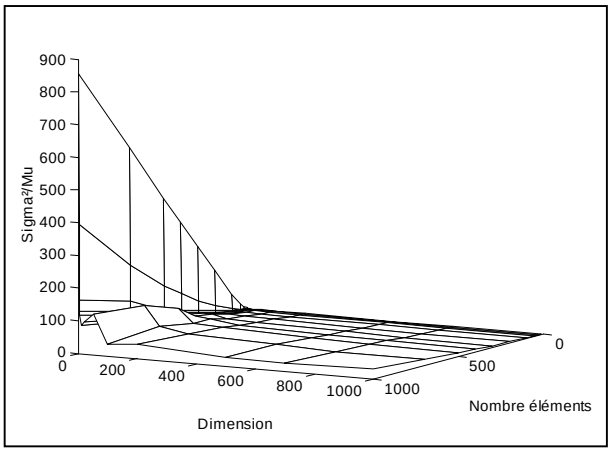

Figure 3 : Abaque σ^2/μ = *f(Nombre d'éléments, Dimension)*

Une analyse précise de la courbe obtenue montre que les dimensions inférieures à 6 induisent des valeurs de σ^2/μ particulièrement élevées. Cela peut s'interpréter en considérant que les effets de la dimensionnalité apparaissent à partir des dimensions 5 ou 6. Contrairement à la dimension de l'espace, l'augmentation du nombre d'éléments dans un sous-arbre engendre une augmentation de σ^2/μ, qui est linéaire à partir d'une certaine valeur du nombre d'éléments. La présence d'un mode centré sur la dimension 50 paraît plus étonnante. Aucune explication n'est actuellement avancée pour interpréter cette inversion passagère de la tendance générale de la courbe. Précisons enfin que les premiers tests effectués en utilisant une distribution Normale ont montré que les valeurs de σ^2/μ suivaient la même évolution que sur l'abaque de la Figure 3.

L'interpolation de cette courbe - effectuée par une fonction relativement complexe - permet en tout noeud de la hiérarchie de savoir si un ou plusieurs agrégats sont présents. Il faut remarquer que puisque cette courbe donne le maximum des σ^2/μ obtenus en chaque point, la présence de plusieurs agrégats risque de ne pas être détectée s'ils sont très proches les uns des autres. Il peut donc être avantageux de construire une abaque sur les valeurs moyennes de σ^2/μ. D'autres méthodes de validation de classe [8] sont également envisageables.

2.2. Parcours de la hiérarchie

Nous avons établi que la détection d'un unique agrégat pouvait être utilisée comme critère d'arrêt de la descente dans une hiérarchie. Il faut néanmoins déterminer la façon de parcourir la hiérarchie de façon à limiter les risques d'erreurs dans l'estimation des agrégats. Le critère de détermination d'un unique agrégat ne fournissant pas toujours une réponse correcte, une exploration dichotomique est insuffisante. L'expérience acquise sur de nombreux cas réels et synthétiques nous a conduit à retenir une méthode aussi simple, mais permettant une meilleure détection des agrégats bien séparés. Le principe est de classer les indices de la hiérarchie par ordre décroissant (en une liste *IndicesClassés*) et de retenir comme seuil de coupure la borne supérieure de l'écart maximal entre deux valeurs consécutives. L'algorithme que nous employons est finalement résumé ci-après.

```
NoeudCourant = Racine de la hiérarchie indicée;
Si (σ²/μ de NoeudCourant > σ̂²/μ (Dim,
   NbEléments de NoeudCourant) ) Alors
      Couper au point maximal de la dérivée de
      la courbe IndicesClassés;
      Explorer récursivement les sous-arbres
      issus de la coupure;
Sinon
      Les éléments appartenant à NoeudCourant
      ont dans le même agrégat;
Fin Si
```

3. Evaluation de musc

Nous avons tout d'abord soumis MUSC aux épreuves de Milligan et Cooper. Les agrégats identifiés correspondent parfaitement aux résultats attendus. La présence de plusieurs agrégats est détectées sans ambiguïté, puisque la valeur de σ^2/μ de la hiérarchie construite selon le saut moyen vaut environ le double de la valeur de la fonction d'interpolation de MUSC. La configuration de la Figure 1 a également été traitée avec succès, puisque les 6 agrégats sont correctement identifiés. La Figure 4 présente un échantillon des tests que nous avons menés.

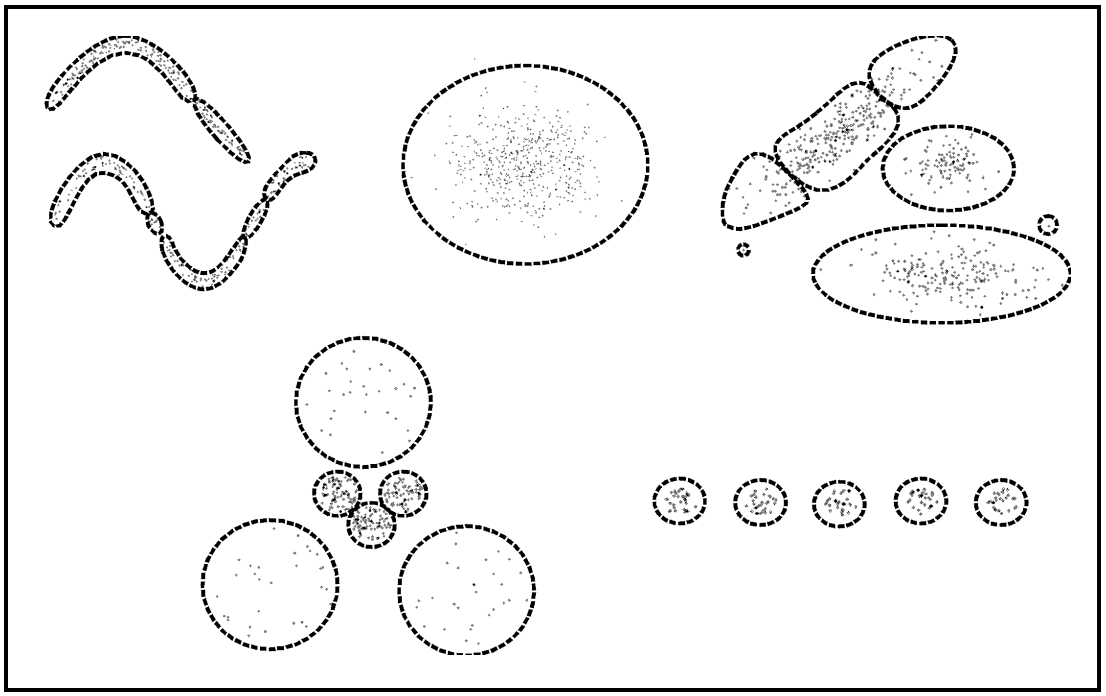

Figure 4 : Détection de configurations synthétiques par MUSC

On peut remarquer que les agrégats de forme sphérique sont généralement bien traités. En revanche, les agrégats les plus allongés ont tendance à être fractionnés. Ce phénomène n'est pas dû à notre algorithme, mais est caractéristique du saut moyen. Les premiers tests effectués avec un algorithme proche du lien simple ont montré que ce phénomène pouvait être pratiquement éliminé.

Nous présentons ci-dessous des tests effectués sur une base réelle constituée de chiffres manuscrits issus de la base NIST. Le vecteur de caractéristiques retenu est présenté sur la Figure 5 : Représentation du chiffre 2 par une pyramide de résolution

Chacun des axes de l'espace de représentation correspond à l'un des 85 carrés de la pyramide de résolution [1]. La valeur sur chaque axe est la moyenne des niveaux de gris des pixels du carré correspondant.

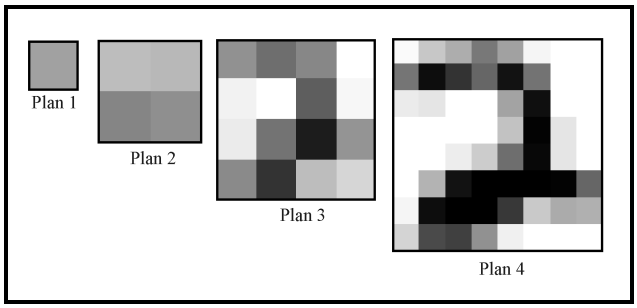

Figure 5 : Représentation du chiffre 2 par une pyramide de résolution

La catégorisation effectuée par MUSC sur une base de 21000 éléments apparaît ci-après. Seuls les agrégats de plus de 50 éléments sont représentés. On note le bon comportement de notre algorithme, puisque les agrégats déterminés correspondent bien aux classes "humaines". Chaque classe est en effet très fortement majoritaire dans au moins un agrégat à fort effectif. On remarque également que certaines classes présentent d'importantes sous-classes, comme le chiffre 8 dont la majorité ressemble aux 3, mais dont certains tracés se rapprochent du chiffre 6.

Il est important de souligner qu'une telle décomposition aurait été impossible en utilisant un coupure unique. En effet, les classes 0 et 1 présentent une densité particulièrement faible. Cela entraîne le même phénomène que sur la base synthétique de la Figure 1, à savoir qu'une coupure unique ne fera apparaître les agrégats les plus denses (classes 4, 7 et 9) qu'au prix d'un important fractionnement des agrégats des classes 0 et 1. Enfin, précisons que le traitement d'une telle hiérarchie requiert un temps processeur d'environ 2 minutes sur une Sparc Station 20. Ce faible temps d'exécution - qui est en O(N) - s'explique par l'absence de retour aux données. Ces premiers résultats de validation de MUSC montrent donc le potentiel de la méthode.

4. Conclusion

Nous avons proposé dans cet article une méthodologie permettant la détermination du nombre et de la composition des agrégats d'un jeu de données à partir d'une hiérarchie indicée. Outre le fait qu'il permette de résoudre les cas de densité variable, notre algorithme présente l'avantage de ne requérir qu'une faible quantité de calculs en exploitant directement l'information contenue dans la hiérarchie et d'offrir une grande simplicité de mise en œuvre. Le principal inconvénient de la méthode réside dans la constitution d'une abaque de référence. Les perspectives de cette étude sont donc essentiellement centrées sur la possibilité d'éliminer cette abaque en prenant en compte la densité des groupes fusionnés lors de la construction de la hiérarchie indicée, de façon à rendre son interprétation triviale.

Bibliographie

[1] Ballard D.H., Brown C.M., *Computer Vision*, Englewood Cliffs, Prentice Hall, 1982.

[2] Barthélémy J.P., Guénoche A., *Les Arbres et les Représentations des Proximités*, Paris, Masson, 1988.

[3] Benzecri J.P., *L'Analyse de données : la taxinomie*, Paris, Dunod, 1973.

[4] Celeux G., Diday E., Govaert G., Lechevallier Y., Ralambondrainy H, *Classification automatique des données*, Paris, Dunod Informatique, 1989.

[5] Diday E., *Optimisation en classification automatique*, INRIA, Tomes 1 et 2, 1980.

[6] Duda R.O., Hart P.E., *Pattern classification and scene analysis*, New-York, Wiley-Interscience Publisher, 1973.

[7] Forgy E., "Cluster analysis of multivariate data : efficiency versus interpretability of classifications", Biometrics, Vol. 21, 768, 1965.

[8] Gordon, A.D., "Hierarchical classification", In P. Arabie, L.J. Hubert, G. De Soete, (eds.): *Clustering and Classification*, (1996), World Scientific Publisher, River Edge, NJ, 65-121.

[9] Jain A.K., Dubes R.C., *Algorithms for clustering data*, Englewood Cliffs, Prentice Hall, 1988.

[10] Lance G.N., Williams W.T., "A general theory of classificatory sorting strategies. 1. Hierarchical systems", *Computer Journal*, Vol. 9 (1967), 373-380.

[11] MacQueen J.B., "Some methods for classification and analysis of multivariate observations". Proceedings of the 5th Berkeley Symposium on Math. Statistics and Probability, Vol. 1, pp. 281-297, 1967.

[12] Milligan G.W., Cooper M.C., "An examination of procedures for determining the number of clusters in a data set", *Psychometrika*, 50, n°2 (1985), 159-179.

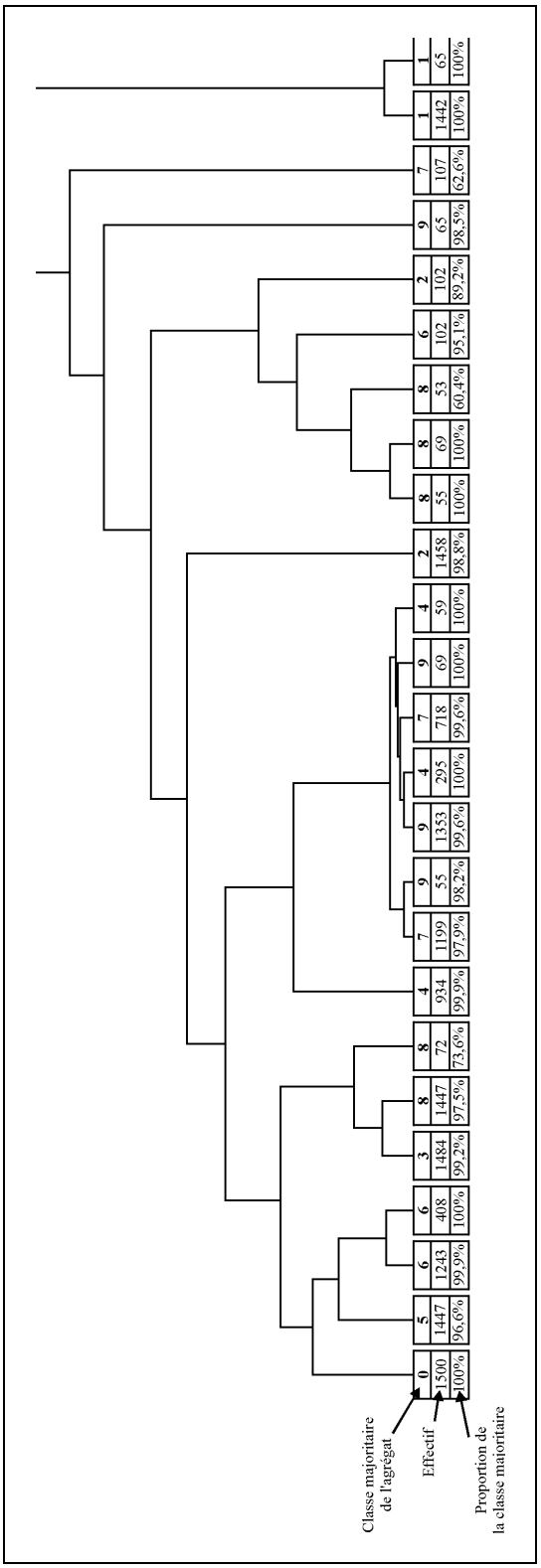

Figure 6 : Catégorisation d'une fraction de 21 000 chiffres de la base NIST à l'aide de MUSC