ELSEVIER

Future Generation Computer Systems 13 (1997) 135–147

FGCS
FUTURE GENERATION COMPUTER SYSTEMS

Clustering techniques

Pierre Michaud [1]

European Centre for Applied Mathematics, IBM ECAM-DSS, 67 Quai de la Rapee, 75012 Paris, France

Received 12 October 1996; accepted 5 April 1997

Abstract

Given a population of individuals described by a set of attribute variables, clustering them into "similar" groups has many applications. The clustering problem, also known as unsupervised learning, is the problem of partitioning a population into clusters (or classes). The population is a set of n elements that can be clients, products, shops, agencies, etc., described by m attributes. These attributes can be quantitative (salary), categorical (type of profession) or binary (owner of a credit card). The goal is to construct a partition in which elements of a cluster are "similar" and elements of different clusters are "dissimilar" in terms of the m attributes. Here we define the clustering problem and discuss the ideas behind some of the major approaches, including a relatively new method, called RDA/AREVOMS, that is based on the theory of voting.

Keywords: Unsupervised learning; Partitioning criteria; Neural network; New Condorcet criterion

1. Introduction

Partitioning a given population of individuals into "similar" groups of individuals has many applications in science and business. The goal of such partitioning, or clustering, may be to gain an insight into some structure inherent in the population (such as sub-species grouping of plants or animals) or to develop a business strategy that is customized to each cluster of customers for higher business efficiency. It is generally not possible to define what it means to be "similar" in terms of the given attributes of the individual elements. It is also true that comparing one clustering result with another is very difficult; judgement is generally subjective and application-dependent. Such difficulties notwithstanding, there are several ways of defining a measure of adequacy (or inadequacy) for a given partition, so that the defined measure can at least serve as an objective function to be maximized (or minimized) over all possible partitions.

Many clustering techniques are based on finding the partition that optimizes such an objective function, which we shall call a partitioning criterion. A partition is a set of mutually disjoint and exhaustive subsets of the population. Since it is impractical to search all possible partitions, clustering methods use various heuristic search strategies to obtain a nearly optimal (or a locally optimal) solution. All possible partitions, of course, refer to all those with

[1] E-mail: pierre_michaud@vnet.ibm.com.

PII S0167-739X(97)00017-4

one cluster (the entire populations), two clusters, etc., including the partition where each element is in its own one-member cluster.

We first discuss partitioning criteria, with two examples: the classical "intraclass inertia" criterion and a relatively new criterion that is derived from axiomatic principles for aggregating conflicting choices. We then present the key ideas behind various clustering approaches based on partitioning criteria and qualitatively compare the results of these techniques applied to a population of felines. We also discuss techniques based on neural networks and some statistical approaches that view a cluster as a probability distribution over the space of attributes, and the data as a sample from a mixture of such probability distributions.

2. Partitioning criteria

Given a population of n elements described by m attributes, an intuitive description of a desirable clustering is that it should have small distances (in the m-dimensional space of attributes) between elements of the same cluster and large distances between elements of different clusters. The partitioning criterion, or the chosen objective function $F(P)$ for a given partition P, is a numerical implementation of this intuitive notion. We shall examine a typical criterion called intraclass inertia, discuss desirable qualities of such criteria, and present a relatively new criterion called the new Condorcet criterion that possesses desirable qualities.

These criteria are useful as objective functions to be optimized in searching for an optimal solution. Optimality, however, cannot usually be claimed for the final outcome of a clustering application. All that can be said is that partitioning criteria tend to produce "acceptable" end results, some more often than others.

2.1. Intraclass inertia

Intraclass inertia is a measure of how compact each class (cluster) is in the m-dimensional space of numeric attributes. Usually the attributes are scaled to have the same range. Let $X_i = (x_{i1}, x_{i2}, \ldots, x_{im})$ be the vector of the m attributes of element i. The centroid (mean) of the kth class L_k that has n_k elements is defined as

$$\bar{X}_k = (\bar{x}_{k1}, \bar{x}_{k2}, \ldots, \bar{x}_{km}),$$

where

$$\bar{x}_{ka} = \frac{1}{n_k} \sum_{i \in L_k} x_{ia}, \quad a = 1, 2, \ldots, m.$$

The intraclass inertia I_k of class L_k is defined as

$$I_k = \sum_{i \in L_k} \sum_{a=1}^{m} (x_{ia} - \bar{x}_{ka})^2.$$

Finally, the intraclass inertia $F(P)$ for a given partition P that has p classes is defined as

$$F(P) = \frac{1}{n} \sum_{k=1}^{p} n_k I_k = \frac{1}{n} \sum_{k=1}^{p} \sum_{i \in L_k} \sum_{a=1}^{m} (x_{ia} - \bar{x}_{ka})^2.$$

One can see that $F(P)$ is the average squared Euclidean distance between each element and its class centroid. In the statistical literature on clustering, $nF(P)$ is known as the within-group sum of squares. This criterion attempts to measure the inadequacy of a partition and therefore one tries to find a partition that minimizes this objective

function. But ignoring the interclass proximity causes this measure to favor higher numbers of classes in general: in particular, the partition that puts each element into its own class, i.e. $p = n$, achieves the minimum value of 0. Therefore, the intraclass inertia criterion cannot be used for an unrestricted search for the best partition. One needs some other mechanism for determining the number of classes p before this criterion can be used as an objective function.

Intraclass inertia was originally defined for continuous or numeric attributes. One can adapt this measure to include categorical or discrete attributes by use of a set of "working variables" as a replacement for a categorical attribute as follows. For a v-valued categorical attribute we substitute a set of v 0–1 valued working variables. The length-v vector of the values of the working variables represents the unitary coding for the value of the attribute, i.e. for the ith value, the vector is all 0s except one 1 in the ith position. These 0–1 values are then treated as numeric.

2.2. *Clustering in the framework of collective choice theory*

Consider a clustering problem with all categorical attributes. One can view each attribute as a judge expressing an opinion that all elements that have the same category value belong to the same class. For instance, the "color" attribute says all things that are red are in one class, yellow in another class, etc. When there are many judges, what is an ideal way to aggregate many such opinions? One may not be able to define an ideal method, but one can attempt to axiomatize desirable properties of aggregation.

While the collective choice problem deals with many kinds of opinions, we are concerned with the problem of collective partitioning. Our hope is to arrive at a "better" partitioning criterion based on the desirable properties of aggregation. We give here only a brief sketch of some of the desirable properties since the theory of collective choice is not the main interest of this paper.

In 1951 and 1963 K.J. Arrow proposed an axiomatic definition of three conditions: non-dictatorship, paired unanimity and "Arrow's independence" (1963 version). He demonstrated the impossibility of satisfying all three conditions, which is the famous Arrow's theorem of impossibility for a "democratic aggregation rule". But in the foreword of the third edition (1974) of his book [2] Arrow pointed out a certain number of modifications that it would be desirable to bring to his theory. We have proposed "partial independence" [14] to replace the "Arrow's independence" condition, giving rise to a positive theory with three basic conditions (non-dictatorship, paired unanimity and partial independence). Many more conditions can be added to these three to define stronger properties [14,15].

As an illustration, let us consider one of the desirable conditions: the "paired unanimity" condition. This condition stipulates that if two elements have identical category values in all the variables, these two elements have to be in a same cluster of the optimal solution. A good criterion should satisfy this very sound and obvious condition. But most classical criteria, including intraclass inertia, do not necessarily yield a partition that satisfies this condition.

We now present the new Condorcet criterion, which meets all the axiomatized desirability conditions mentioned above. The reason for the name new Condorcet criterion is that Condorcet, in 1785, introduced a similar criterion for the aggregation of rankings.

2.3. *New Condorcet criterion*

The new Condorcet criterion (NCC) was originally proposed using the notion of equivalence relation since a partition is an equivalence relation. Fortuitously, the same criterion, designed to satisfy the conditions for desirable

collective choice, fully implements the intuitive notion of what a good partition should be mentioned earlier. So we will start with the criterion in the form that reveals the intuitive point of view.

The NCC is defined for categorical attributes. For a categorical attribute, we measure the distance between attribute values as 1 if two elements have different values and 0 otherwise. For m attributes, the distance between two elements is the sum of the component distances (i.e. Manhattan distance). The distance between two elements can be viewed as a modified Hamming distance, that is, the number of attributes for which the two elements have different values. Therefore, the distance d_{ij} between two elements i and j is the number of "judges" who "disagree" about whether elements i and j should be in the same class (and $m - d_{ij}$ is the number of agreements).

The NCC measures intraclass agreements as well as interclass disagreements and combines them in such a way that partitions that have small intraclass distances and large interclass distances will have higher measure. Such a measure can be expressed as

$$G(P) = \sum_{k=1}^{p} \sum_{i \in L_k} \left(\sum_{\substack{j \in L_k \\ j \neq i}} (m - d_{ij}) + \sum_{j \notin L_k} d_{ij} \right).$$

A partition can be represented by an equivalence relation using an $n \times n$ 0–1 matrix $Y = [y_{ij}]$, where $y_{ij} = 1$ if i and j belong to the same (equivalence) class and 0 otherwise. The NCC above can be rewritten with Y notation as

$$G(Y) = \sum_{i=1}^{n} \sum_{j \neq i} C'_{ij},$$

where $C'_{ij} = d_{ij}$ if $y_{ij} = 0$ and $C'_{ij} = m - d_{ij}$ if $y_{ij} = 1$. Since y_{ij} takes only 0–1 values, we have

$$G(Y) = \sum_{i=1}^{n} \sum_{j \neq i} \{d_{ij}(1 - y_{ij}) + (m - d_{ij})y_{ij}\} = \sum_{i=1}^{n} \sum_{j \neq i} (m - 2d_{ij})y_{ij} + \sum_{i=1}^{n} \sum_{j \neq i} d_{ij}.$$

The second term is a constant depending only on the elements and not on the given partition. Therefore, the NCC based formulation $F(Y)$ derived from the collective choice considerations [12] is essentially the same as $G(Y)$ above

$$F(Y) = \sum_{i=1}^{n} \sum_{j \neq i} (m - 2d_{ij})y_{ij} = \sum_{i=1}^{n} \sum_{j \neq i} C_{ij} y_{ij},$$

where $C_{ij} = m - 2d_{ij} = (m - d_{ij}) - d_{ij}$, i.e. the number of "agreements" minus the number of "disagreements" about (i, j) being in the same class.

The NCC measures the adequacy of a partition and one should try to maximize $F(Y)$ when searching for a corresponding partition P. Unlike the intraclass inertia criterion, the NCC does not favor larger numbers of classes. Therefore, it can be used as an objective function in searching for the optimal partition without fixing the number of classes.

When there are numerical variables, they must be discretized so that the new Condorcet criterion can be applied. Often, domain experts can offer meaningful discretization.

2.4. *NCC based evaluation and presentation of clusters*

The NCC in the form of $G(P)$ leads to a natural way of analyzing and understanding a given partition P. Various components of $G(P)$ are called *supports*. For an element i ($i \in L_k$ for some cluster L_k) its *intra-cluster-element*

support and *inter-cluster-element support* is

$$\sum_{\substack{j\in L_k \\ j\neq i}} (m - d_{ij}) \qquad \text{and} \qquad \sum_{j\notin L_k} d_{ij},$$

respectively. The sum of these two components is the *element support*. We see that $G(P)$ is the *total support*, i.e., the sum of all element supports. The sum of all element supports within a cluster yields its *cluster support*. These supports can be expressed in a relative sense as percentages of corresponding maximum possible supports (e.g., the maximum possible element support is $m(n-1)$). Such relative supports indicate the quality of the given partition on the whole, for a cluster, or for an element. For example, comparing the relative element support of each element in a cluster, we know which is most or least typical within the cluster. One can even break down such comparison for intra and inter qualities separately.

The total support of NCC can be divided into contributions from each attribute. The *attribute support* for an attribute a is

$$\sum_{k=1}^{p} \sum_{i\in L_k} \left(\sum_{\substack{j\in L_k \\ j\neq i}} (1 - d_{ij,a}) + \sum_{j\notin L_k} d_{ij,a} \right),$$

where $d_{ij,a}$ is 0 if the attribute a has the same value for elements i and j, 1 otherwise. The partial sum of the above, i.e. $\sum_{i\in L_k}$, for each cluster is *cluster-attribute support*. Attributes that have relatively high support are called leading attributes. The leading attributes for the clusters are usually the same as the global leading attributes, but it can be a different set. A cluster is generally characterized by the distribution of values of its leading attributes within the cluster vs. on the whole.

For further details of NCC support based analysis, see [13].

3. Partitioning based clustering

Many clustering techniques try to find a partition that optimizes the chosen partitioning criterion. The search for the optimum solution is impractical due to the sheer number of possible partitions. Depending on the kind of heuristics used in the search process, we have three major approaches; hierarchical, k-means family, and the NCC-based RDA/AREVOMS.

Most traditional partitioning criteria, such as intraclass inertia, favor larger numbers of clusters as pointed out earlier. When the clustering method uses one of these criteria as the objective function, one needs to determine the number of clusters in advance or during the process of generating the partition. The first two approaches we discuss here have this problem while the NCC based RDA/AREVOMS does not.

In many cases a statistician would approach a clustering problem by first trying to get a feel for the number of clusters and also to discover potentially useful derived variables. This is usually done by performing a factor analysis or correspondence analysis to see whether some clusters develop in a visualizable low-dimensional subspace of the space of attributes. While some useful insight may be gained by this practice, our concern here is limited to actual development of the clusters.

3.1. Hierarchical clustering methods

Hierarchical clustering is perhaps the oldest heuristic approach to obtain a nearly optimal solution. In agglomerative hierarchical clustering, one starts with n clusters with one element in each cluster. Iteratively, a pair of clusters

is merged into one, decreasing the number of clusters by 1. The pair of clusters to merge is determined by the best objective function value obtainable by the merge. There are many hierarchical methods differing basically in the choice of partitioning criterion. The well-known Ward's minimum-variance method [24] uses a criterion equivalent to the intraclass inertia criterion.

The iterative merging process thus develops a hierarchy of clusters. It may stop at a "reasonable" number of clusters, which may be suggested from knowledge of the domain of application, or when the next merge causes a relatively large increase (a "jump") in the objective function value.

Naive implementation of agglomerative hierarchical clustering have computational complexity $O(n^3)$, but $O(n^2)$ implementations have been developed [18]. The approach may still be impractical for very large problems.

In contrast to agglomerative procedures, divisive hierarchical clustering starts by considering the entire set of elements as a single cluster, and iteratively splits one cluster into two until each element is in its own cluster (or until some other stopping criterion is satisfied). Divisive methods tend to be less widely used than agglomerative methods, because they cannot recover from a poor choice in the early splits.

3.2. *K-means procedure*

A widely used clustering procedure searches for a nearly optimal partition with a fixed number of clusters. First an initial partition with the chosen number of clusters is built (it can be done in many ways). Then, keeping the same number of clusters, the partition is improved iteratively. Each element is handled sequentially and reassigned to the cluster such that the partitioning criterion is most improved by the reassignment. This is a "minor iteration"; a sequence of n minor iterations, one for each element, is a "major iteration". Usually the procedure ends when no improving reassignment is obtained in a major iteration, but stronger end tests can also be used.

As in the hierarchical approaches, different solutions are obtained depending on which partitioning criterion is used. The most widely used criterion is interclass inertia, and the resulting method is known as the k-means procedure. An early approach of this kind was proposed by Forgy in 1965 [6] (called H-means in [20]) and various improved variations of it have been proposed modifying the initialization step, the number of major iterations or the updating of the centroid of each class. MacQueen's k-means method [8] uses intraclass inertia criterion and other variations can be found in [1,20].

Since the main k-means procedure works on a fixed number of clusters, one needs to repeat the procedure with different numbers of clusters for a final solution. If only a small set of numbers of clusters is searched, the computation requirement is essentially linear in n.

3.3. *RDA/AREVOMS method*

The NCC criterion defined earlier can directly serve as the objective function of an integer programming optimization problem: maximize

$$F(Y) = \sum_{i=1}^{n} \sum_{j \neq i} C_{ij} y_{ij}$$

subject to the following constraints for enforcing a partition:

$$y_{ij} = 0, 1 \quad \text{(integer)},$$
$$y_{ij} = y_{ji} \quad \text{(symmetry)},$$
$$y_{ij} + y_{jk} - y_{ik} \leq 1 \quad \text{(transitivity)}.$$

Unlike the methods described earlier, the optimum solution automatically determines the number of clusters. This formulation was originally called *relational data analysis (RDA) for clustering* [12] where the optimization was carried out using integer programming software (RDA itself includes more than one equivalence relation). One serious problem in having to use integer programming is that its exponential computation time makes it impractical for large data sets. Using linear programming instead of integer programming reduces run time but the optimality suffers [12]. Early efforts to develop a fast and effective heuristic solution to this problem brought the computation time close to $O(n^2)$ [9]. Finally, in the late 1980s, a new heuristic algorithm was developed by Michaud based on a transformation of the problem into a simpler domain by use of his "S-theory" [16,17]. The new heuristic algorithm runs in time linear in n while producing solutions that are very close to optimal and it also allows fixing the number of clusters if desired.

Since the NCC satisfies the desirable conditions from the collective choice point of view, the RDA clustering solution can be shown to have useful properties that aid in the presentation and explanation of the clusters. One can show that each cluster has at least one attribute (called the characteristic attribute) such that the majority of the elements have the same value in that attribute. The clusters are analyzed and effectively explained by the NCC based support measures defined in Section 2.4.

We call our NCC based clustering and cluster explanation methodology RDA/AREVOMS (AREVOMS comes from French "Analyse du resultat d'un vote majoritaire" [13] and S-theory). After successful experience in generating clustering solutions to many large business applications in France, the RDA/AREVOMS method has been incorporated into the recent data mining system, Intelligent Miner, of IBM.

3.4. An example of clustering felines

We now use a small example to illustrate the clustering results of the three methods based on partitioning criteria. (For a benchmark style comparison of various clustering techniques, see [11].) The population consists of 30 different feline animals characterized by 14 categorical attributes, as shown in Appendices A and B.

Clustering results are shown in Appendix B by columns of cluster numbers. Animals with the same cluster number were clustered together by the method used for the column. For a hierarchical method, we used the CLUSTER procedure from SAS [19] with Ward's method option, and for k-means we used the FASTCLUS procedure also from SAS with least-squares option. Both of these procedures use the intraclass inertia criterion. For these two methods the attributes were transformed to working variables as described earlier. The intraclass inertia values of the hierarchical clustering are shown below for descending number of clusters (30 to 1) separated by a comma after groups of five.

```
( 0  0  0  0  0,   0  1.5  4.5  7.5 10.5, 13.5  16.5  19.5  22.5  26.2,
 30 34 38 42 47, 52.6 58.6 64.6 72.1 80.9, 90.6 101.3 116.6 138.2 179.7)
```

The columns H5, H4, and H3 in Appendix B are the results of 5, 4 and 3 cluster solutions, with intraclass inertia of 90.6, 101.3 and 116.6, respectively.

The k-means results are in the columns K5, K4 and K3 again for 5, 4 and 3 clusters; intraclass inertia were 96.5, 107.5 and 118.3, respectively. Usually, k-means yields better measures than hierarchical methods, especially if the clusters from the hierarchical method can be used as the seed. FASTCLUS is tuned for large numerical data. The sequential move could not lead to a better solution for this small case with all categorical attributes.

The RDA/AREVOMS result is shown in column R in Appendix B. It automatically generated four clusters. For the other two methods, there was no clear indication for selecting the number of clusters and we chose solutions for

5, 4 and 3 clusters so that they may be compared to the 4 cluster solution of the RDA/AREVOMS method. In cluster 1, for instance the most typical elements are GOLDEN, BORNEO and TEMMINCK with relative percentages for element, intra-cluster-element and inter-cluster-element support quality of 74.4, 81.6 and 60.7, respectively. The least typical element is CARACAL with the corresponding quality of 57.6, 56 and 60.7. An analysis of the leading attributes, in the appendix, gives an insight into the reason why there are the two small clusters, (LION, TIGER) and (GUEPARD, SERVAL).

Depending on the application, some of these results may be more useful than others.

4. Model based clustering

Clustering methods that do not depend on optimizing a partitioning criterion make use of certain models for the clusters and attempt to optimize the fit between the data and the model.

4.1. Neural network approach

Neural network approaches have been used to solve a range of problems related to data mining. Predictive modeling problems are frequently addressed using supervised training approaches such as the traditional backpropagation method. Neural networks, using unsupervised training, can also be applied to clustering problems; such applications are often referred to as "unsupervised classification" methods.

Perhaps the best-known neural network approach to clustering is the self-organizing feature map (SOM) often identified with Kohonen [7]. The SOM can be viewed as a nonlinear projection from an m-dimensional input space onto a low-order (typically two-dimensional) regular lattice of cells. Such a mapping is often useful in detecting and visualizing characteristic features of the input data, and ultimately in identifying clusters of elements that are similar (in a Euclidean sense) in the original m-dimensional space.

A reference or weight vector $M_k \in \mathbb{R}^m$ is associated with each cell k in the two-dimensional lattice or "map". After random initialization, the reference vectors are updated during the training phase by making repeated passes (epochs) over the input data set. Let $X \in \mathbb{R}^m$ represent an input data vector. Each input vector X is compared with the current set of reference vectors, and the best matching cell C is chosen based on a Euclidean distance measurement

$$C = \arg\min_k \| X - M_k \| .$$

A key aspect of the SOM learning algorithm is that the reference vector associated with the "winning" cells and its neighbors on the SOM lattice are adjusted using the learning rule

$$M_k(t+1) = M_k(t) + h_{ck}(t)(X_t - M_k(t)).$$

Here, t is an integer denoting the tth input element X_t (X_is repeated in epochs), and $h_{ck}(t)$ is the so-called neighborhood kernel specifying the neighboring cells around the winning cell C. The neighborhood function is often represented by the Gaussian function, with a width that decreases as learning progresses. Other learning rules can be used, e.g. the batch SOM [7] approach which resembles the traditional k-means algorithm when only the reference vector associated with the winning cell C is updated.

The converged weight vectors obtained in the above training process can be viewed as prototype vectors ordered on a two-dimensional map. Input vectors used in the training, as well as newly arriving input elements, are clustered by assigning each element to the closest cell determined as above. Techniques for visualizing the SOM map are also discussed in [7].

The SOM process is linear in n. The topology of the cells, the number of cells and the number of epochs for the SOM process is usually determined by several experiments and depends on the application.

4.2. Statistical approaches

In statistical clustering, the data are regarded as a sample from a random process that generates elements as points in attribute space. Points in cluster k are assumed to occur with probability π_k and to be distributed according to a probability distribution that depends on a vector of parameters θ_k and has a probability density (or, for discrete-valued attributes, a probability function) $p_k(\cdot\,; \theta_k)$. The probability density for each element is then the mixture of densities for each cluster,

$$\sum_{k=1}^{p} \pi_k p_k(\cdot\,; \theta_k),$$

and the likelihood function for the data $\{X_1, \ldots, X_n\}$ is the probability density for the entire data set,

$$\prod_{i=1}^{n} \sum_{k=1}^{p} \pi_k p_k(X_i; \theta_k).$$

A standard procedure is to estimate the parameters π_k and θ_k $(k = 1, \ldots, p)$ by maximizing the likelihood function, obtaining estimates $\hat{\pi}_k$ and $\hat{\theta}_k$ $(k = 1, \ldots, p)$, and to assign element i to the cluster k for which $p_k(X_i; \hat{\theta}_k)$ is largest. Fractional assignment, with element i's cluster membership assigned proportionally to $p_k(X_i; \hat{\theta}_k)$, is also used. The likelihood function typically has many local maxima, and the estimation procedure can be complex; it is often implemented using the EM algorithm [5]. A general discussion of mixture models is given in [21].

For continuous-valued attributes, the cluster densities $p_k(\cdot\,; \theta_k)$ are often assumed to be multivariate Normal, and θ_k contains the elements of the mean vector and covariance matrix of the distribution. If the covariance matrices for different clusters are constrained to be the same multiple of the identity matrix, then maximizing the likelihood is equivalent to minimizing the intraclass inertia, and estimation is analogous to the k-means procedure. Other possible structures for the covariance matrices are discussed in [3].

The maximum value of the likelihood function increases as the number of clusters increases, so the maximum likelihood approach cannot be used to identify the appropriate number of clusters. Statistical tests to determine the number of clusters have been proposed, but none seems to be reliable for all classification problems [10].

Bayesian statistical analysis provides a way of estimating the number of clusters, at the cost of having to specify prior distributions for the number of clusters and for all of the π_k and θ_k parameters. "Bayes factors" can be computed for choosing between clusterings with different numbers of clusters [3].

AutoClass [4] is a thorough implementation of a Bayesian clustering procedure based on mixture models. Various kinds of discrete and continuous attributes are permitted. For example, continuous real-valued attributes are modeled by the Normal distribution, continuous positive-valued attributes by the lognormal distribution, integer-valued attributes by the Poisson distribution and categorical attributes by the multinomial distribution. In general, AutoClass makes the "naive Bayesian" assumption that attribute values within each cluster are independently distributed. However, some forms of correlation between attributes are permitted: for example, correlated continuous attributes may be modeled by a multivariate Normal distribution, and categorical attributes by a multinomial distribution fitted to the cross-product of individual attribute values. AutoClass has been successfully used on several large-scale problems; an analysis of satellite data on infrared stellar spectra (5425 elements, 100 attributes) is reported in [4].

The minimum message length criterion (MML) [23] also provides a means of estimating the number of clusters. "Snob" [22] is a clustering procedure that uses mixture models and MML.

5. Conclusions

The utility of clustering in a given application is so difficult to formalize that it is not possible in general to compare the merits of one clustering with another without actually evaluating the effect of the clustering in the application. All the methods we have presented here have success stories in a variety of applications. Each of these can produce many different clustering results depending on the choice of attributes, perhaps weighting them, and, of course, by fixing the number of clusters or other parameters of the method. Given the situation, the clustering task often requires some experimental search.

The RDA/AREVOMS method based on the new Condorcet criterion was motivated by the desirable qualities of collective clustering decision. But the quality of attributes (the individual decisions) has strong influence in the result and its ultimate utility for the application. Therefore, it remains a conjecture that RDA/AREVOMS method will produce useful clustering results more often than others.

Since there are so many high-impact applications, such as customer segmentation and automatic generation of text groupings, clustering will continue to be an important research area. If the reason for the clustering task can be formalized in the context of the application, we may then be able to produce the "best" clustering for the ultimate purpose.

Acknowledgements

I would like to thank Jonathan Hosking, Richard Lawrence and Se June Hong for thoughtful comments and contributions to improve the original draft of this paper. I also thank Eric Azoulay for the clustering results on the feline problem obtained from SAS (CLUSTER and FASTCLUS) programs.

Appendix A. Description of the variables

1	COATTYPE	1-Plain	2-Dotted	3-Striped	4-Mottled
2	HAIRLENG	1-Short	2-Long		
3	CLAWS	1-Nonretr	2-Retrac		
4	COMPORT	1-Diurnal	2-Diu-Noct	3-Nocturnal	
5	EARS	1-Round	2-Prick		
6	LARYNX	1-No-bone	2-Bone		
7	SIZE	1-Small	2-Medium	3-Large	
8	WEIGHT	1-Light	2-Medium	3-Heavy	
9	LENGTH	1-Small	2-Medium	3-Big	
10	TAIL	1-Long	2-Medium	3-Short	
11	TEETH	1-Can-NDev	2-Can-Dev		
12	PREYTYPE	1-Big	2-Big-Small	3-Small	
13	TREES	1-Notclimb	2-Climb		
14	HUNTING	1-No	2-Yes		

Appendix B. Attribute values and cluster results

Attributes		Characteristics of animals														Clustering results						
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	R	H5	H4	H3	K5	K4	K3
1	LION	1	1	2	1	1	2	3	3	3	2	2	1	1	2	3	2	2	2	2	2	2
2	TIGER	3	1	2	3	1	2	3	3	3	2	2	1	1	1	3	2	2	2	2	2	2
3	JAGUAR	2	1	2	2	1	2	3	3	2	1	2	1	2	1	2	2	2	2	2	2	2
4	LEOPARD	2	1	2	3	1	2	3	3	2	2	2	2	2	1	2	2	2	2	4	4	2
5	OUNCE	2	2	2	1	1	2	2	2	2	3	2	2	2	1	2	5	4	2	4	4	3
6	GUEPARD	2	1	1	1	1	1	3	2	2	3	1	2	1	2	4	4	3	3	5	4	3
7	PUMA	1	1	2	2	1	1	2	3	2	3	2	2	2	1	2	5	4	2	4	4	2
8	NEBUL	4	1	2	3	1	2	2	2	2	3	2	3	2	1	2	5	4	2	4	4	2
9	SERVAL	2	1	2	1	2	1	2	2	2	1	1	3	2	2	4	3	3	3	3	3	3
10	OCELOT	2	1	2	2	1	1	2	2	2	2	1	3	2	1	1	4	3	3	1	1	1
11	LYNX	2	2	2	2	2	1	2	2	2	1	2	2	2	1	2	5	4	2	4	4	3
12	CARACAL	1	1	2	2	2	1	2	2	1	1	1	3	2	2	1	3	3	3	3	3	1
13	VIVERRIN	2	1	2	2	1	1	1	1	2	2	1	3	1	1	1	4	3	3	1	1	1
14	YAGUARUN	1	1	2	2	1	1	1	2	2	3	1	3	2	1	1	4	3	3	1	1	1
15	CHAUS	1	2	2	3	2	1	1	2	1	2	1	3	2	1	1	1	1	1	3	3	1
16	GOLDEN	1	1	2	3	1	1	1	1	1	2	1	3	2	1	1	1	1	1	1	1	1
17	MERGUAY	2	1	2	3	1	1	1	1	1	2	1	3	2	1	1	1	1	1	1	1	1
18	MARGERIT	1	2	2	2	1	1	1	1	1	2	1	3	1	1	1	1	1	1	1	1	1
19	CAFER	3	1	2	3	1	1	1	1	1	2	1	3	2	2	1	1	1	1	1	1	1
20	CHINA	1	1	2	2	2	1	1	1	1	1	1	3	2	1	1	3	3	3	3	3	1
21	BENGAL	2	1	2	3	1	1	1	1	1	2	1	3	2	1	1	1	1	1	1	1	1
22	ROUILLEU	2	1	2	2	1	1	1	1	1	2	1	3	2	1	1	1	1	1	1	1	1
23	MALAY	1	2	2	3	1	1	1	1	1	1	1	3	2	1	1	1	1	1	1	3	1
24	BORNEO	1	1	2	3	1	1	1	1	1	2	1	3	2	1	1	1	1	1	1	1	1
25	NIGRIPES	2	1	2	2	1	1	1	1	1	1	1	3	2	2	1	1	3	3	1	3	1
26	MANUL	1	2	2	3	1	1	1	1	1	1	1	3	2	1	1	1	1	1	1	3	1
27	MOTTLED	4	1	2	3	1	1	1	1	1	3	1	3	2	1	1	1	1	1	1	1	1
28	TIGRIN	2	1	2	3	1	1	1	1	1	2	1	3	2	1	1	1	1	1	1	1	1
29	TEMMINCK	1	1	2	3	1	1	1	1	1	2	1	3	2	1	1	1	1	1	1	1	1
30	ANDES	2	2	2	3	1	1	1	1	2	2	1	2	2	1	1	1	1	1	1	1	1
									Number of clusters							4	5	4	3	5	4	3

Appendix C. Support analysis of RDA/AREVOMS clustering (column R)

The leading attributes of clusters 1–3 are nearly the same, but cluster 4 has a different set of leading attributes.

Global quality of clusters

	Both%	Intra%	Inter%
Total support	65.7	72.5	59.5
Leading attribute supports			
11 Teeth	88.1	100.0	77.2
7 Size	84.6	78.3	90.4
12 Preytype	81.8	86.0	78.1
6 Larynx	77.0	96.1	59.7
8 Weight	75.9	64.7	86.0
9 Length	75.2	69.1	80.7

Atttribute supports for Cluster 3: (LION, TIGER)

	Both%	Intra%	Inter%	Maj.cat	%class	%tot
All attributes	66.2	78.6	65.9			
Leading attribute supports						
9 Length	100.0	100.0	100.0	Big	100.0	6.7
12 Preytype	96.5	100.0	96.4	Big	100.0	10.0
7 Size	89.5	100.0	89.3	Large	100.0	16.7
8 Weight	89.5	100.0	89.3	Heavy	100.0	16.7
13 Trees	89.5	100.0	89.3	Notclimb	100.0	16.7
6 Larynx	86.0	100.0	85.7	Bone	100.0	20.0

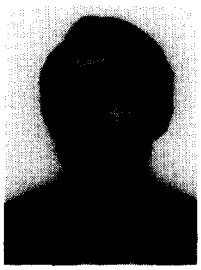
Atttribute supports for Cluster 4: (GURPARD, SERVAL)

	Both%	Intra%	Inter%	Maj.cat	%class	%tot
All attributes	58.3	57.1	58.3			
Leading attribute supports						
4 Comport	93.0	100.0	92.9	Diurnal	100.0	13.3
14 Hunting	86.0	100.0	85.7	Yes	100.0	20.0

References

[1] M.R. Anderberg, *Cluster Analysis for Applications* (Academic Press, New York, 1973).
[2] K.J. Arrow, *Social Choice and Individual Values* (Wiley, New York, 1963), 3rd Ed. in French (1974).
[3] J.D. Banfield and A.E. Raftery, Model-based Gaussian and non-Gaussian clustering, *Biometrics* 49 (1993) 803–822.
[4] P. Cheeseman and J. Stutz, Bayesian classification theory (AutoClass): Theory and results, in: *Advances in Knowledge Discovery and Data Mining*, eds. U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth and R. Uthurusamy (AAAI Press/MIT Press, Cambridge, MA, 1996).

[5] A.P. Dempster, N.M. Laird and D.B. Rubin, Maximum likelihood from incomplete data via the EM algorithm, *J. Roy. Statist. Soc. Ser.* B 39 (1) (1977) 1–38.
[6] E.W. Forgy, Cluster analysis of multivariate data: Efficiency versus interpretability of classifications, *Biometrics* 21 (3) (1965) 768.
[7] T. Kohonen, *Self-Organizing Maps* (Springer, Berlin, 1995).
[8] J.B. Mac Queen, Some methods for classification and analysis of multivariate observations, *Proc. 5th Berkeley Symp. on Mathematical Statistics and Probability* 1 (1) (1967).
[9] J.-F. Marcotorchino and P. Michaud, Heuristic approach of the similarity aggregation problem, *Methods Oper. Res.* 43 (1981) 395–404.
[10] G.J. McLachlan and K.E. Basford, *Mixture Models: Inference and Applications to Clustering* (Marcel Dekker, New York, 1988).
[11] M. Zait and H. Messatfa, A comparative study of clustering methods, *Future Generation Computer Systems* (1997).
[12] P. Michaud, Opinions aggregation, in: *New Trends in Data Analysis and Applications*, eds. J. Janssen, J.-F. Marcotorchino and J.-M. Proth (North-Holland, Amsterdam, 1983) 5–27.
[13] P. Michaud, Agrégation à la majorité II: Analyse du résultat d'un vote, Centre Scientifique IBM France, étude No F-052, Paris (1985).
[14] P. Michaud, Hommage à Condorcet (version intégrale pour le bicentenaire de l'essai de Condorcet), Centre Scientifique IBM France, étude No F-094, Paris (1985).
[15] P. Michaud, Condorcet – A man of the Avant-garde, in: *Applied Stochastic Models and Data Analysis*, eds. J. Janssen, F. Marcotorchino, J. Proth and P. Purdue, Vol. 3, No. 3 (Wiley, Chichester, 1987).
[16] P. Michaud, Simulated computation in automatic classification, *Proc. 2nd Symp. on High Performance Computing*, eds. M. Durand and F. El Dabaghi (Montpellier, France, 7–9 October 1991) 381–396.
[17] P. Michaud, Variational data analysis versus classical data analysis, in: *Advances in Stochastic Modelling and Data Analysis*, eds. J. Janssen, C.H. Skiadis and C. Zopounidis (Kluwer Academic Publisher, Dordrecht, 1995) 128–158.
[18] F. Murtagh, *Multidimensional Clustering Algorithms* (Physica-Verlag, Vienna, 1985).
[19] SAS/STAT User's guide, Version 6, 4th Ed., Vols. 1 and 2, SAS Institute, Cary NC (1989).
[20] H. Spath, *Cluster Analysis Algorithms for Data Reduction and Classification of Objects* (Ellis Horwood, Chichester, 1980).
[21] D.M. Titterington, A.F.M. Smith and U.E. Makov, *Statistical Analysis of Finite Mixture Distributions* (Wiley, New York, 1985).
[22] C.S. Wallace and D.L. Dowe, Intrinsic classification by MML – the Snob program, *Proc. 7th Australian Joint Conf. on Artificial Intelligence* (1994) 37–44.
[23] C.S. Wallace and M.P. Georgeff, A general selection criterion for inductive inference, *Proc. ECAI-84* (1984) 473–482.
[24] J.H. Ward, Hierarchical grouping to optimize an objective function, *J. Amer. Statist. Assoc.* 58 (301) (1963) 236–244.



P. Michaud received his M.S. degree from Paris VI University in 1964, and Ph.D. in numerical analysis and statistics and Doctorat d'Etat degree from Paris in 1967 and 1982. He joined IBM White Plains in 1967 working in various fields of optimization. He joined IBM Paris Scientific Centre, Paris in 1970 where he was project leader (1970–1990) and group leader (1991–1994) in the field of data analysis and decision support. He is currently Scientific Adviser of IBM ECAM-GBIS (European Centre for Applied Mathematics-Global Business Intelligence Solution). He is member of ASMDA (Applied Stochastic Model and Data Analysis) and AFCET (Association Française pour la Cybernétique Economique Technique) and external Professor at Paris University. His research interest includes data analysis and decision support, collective choice theory operational research, network optimization, and compression theory for intensive computing.