
Copyright : 2001, IEEE
Fuzzy clustering of non-convex patterns using global optimization

Gleb Beliakov
School of Computing and Mathematics
Deakin University
662 Blackburn Rd., Clayton, 3168, Australia

Abstract

This paper discusses various extensions of the classical within-group sum of squared errors (WGSS) functional, routinely used as the clustering criterion. Fuzzy c-means algorithm is extended to the case when clusters have irregular shapes, by representing the clusters with more than one prototype. The resulting minimization problem is non-convex and non-smooth. A recently developed cutting angle method of global optimization is applied to this difficult problem.

I. INTRODUCTION

Clustering, or unsupervised classification, is based on the assumption that the data are made up of distinct classes, but allocation of data to these classes is not known a priori. The general task is to discover homogeneous groups or categories in the data, based on the attribute values. The elements in each cluster must be as similar as possible to each other and dissimilar to those in other clusters. Unsupervised classification is a useful tool in exploratory data analysis, which allows pattern discovery in the data.

Objective function-based cluster analysis is a popular technique in unsupervised data classification [1-4]. It is based on optimization of a function measuring the quality of partitioning the data into clusters. Prototypes are usually represented by their centres, and more generally, prototypes (linear, ellipsoidal, etc. [2, 4]). The objective function measures within class similarity of data to the prototypes in a certain metric. Its global optimum is generally regarded as the best clustering, although there are examples where this is not the case [1].

The classical WGSS (within-group sum of squared errors) functional is the most extensively studied clustering criterion [1, 5, 6] which generates hard partition of the data. One of the most popular algorithms of approximating its minimum is c-means algorithm. Its extension for fuzzy partitions is known as fuzzy c-means [1]. Fuzzy c-varieties algorithm was developed for recognition of lines, planes and hyperplanes [4]. Fuzzy c-shells algorithms optimize similarity to shell prototypes, specified in advance with the help of few parameters.

C-means algorithms (both hard and fuzzy) have two easily identifiable drawbacks: a) they can deal only with clusters of regular (spherical, elliptical, linear) shape, and b) they converge to the nearest local (and not global) minimum, thus generating suboptimal partition. The extensions (c-varieties, c-shells, generalised c-means, c-mixed prototypes) handle clusters with other shapes, but those must be specified in advance [2, 4, 7]. Developments in recognition of clusters of arbitrary shape using varying metrics are described in [8, 9].

The lack of global convergence is due to the non-convexity of the WGSS functional. Suboptimal solutions result in bogus clusters and misleading data associations, and poor performance of c-means algorithm in this respect has been illustrated in [10]. Modifications of the WGSS functional (i.e., k-median algorithm [11]) do not help, and alternative techniques relying on random search, simulated annealing and genetic algorithms have been studied in [10, 12-15].

In this paper we describe a method of recognition of clusters of arbitrary shape, not fixed a priori. It is based on the representation of cluster prototypes using a vector of vectors (cluster "centres"), rather than single vector. Each cluster can have several "centres", (or prototypes, control points), and their positions determine the shape of the cluster. This approach is more general than representing clusters parametrically [4, 7]. The modified WGSS functionals are non-convex and non-smooth, and their minimization is a more difficult problem than in the case of a single centre. However, recent advances in global optimization techniques allow one to find the optimum solution to this problem with reasonable computational effort. We present one such technique, known as the cutting angle method. This deterministic approach does not rely on the smoothness of the functional, but on its Lipschitz properties. Hence it is suitable for optimization of WGSS in its original and modified forms.

II. C-MEANS FUNCTIONALS

The classical WGSS (within-group sum of squared errors) functional is the most extensively studied clustering criterion [1, 6], which generates hard partition of the data. It can be defined as

$$J(U, \mathbf{v}) = \sum_{k=1}^{K} \sum_{i=1}^{c} u_{ik} \| x_i - v_k \|^2,$$

where vectors $v_i, i = 1, ..., c$ are cluster centres, $x_i, k = 1, ..., K$ are the data, and

$$u_{ik} \in \{0,1\}, \sum_{i=1}^{c} u_{ik} = 1, \text{ for } \forall k$$

determine data allocation into the clusters. Minimization of (1) is a difficult combinatorial problem.

However, $U$ and $\mathbf{v}$ are interdependent: given cluster partition $U$, centres $\mathbf{v}$ are determined as centroids of the data

$$v_i = \frac{\sum_{k=1}^{K} u_{ik} x_k}{\sum_{k=1}^{K} u_{ik}},$$
and given cluster centres \( \mathbf{v} \), data vectors \( \mathbf{x} \) are assigned to the closest cluster. The validity of this relationship at extreme points of (1) was established in [5]. Hard c-means algorithm [6] iteratively minimises (1) using the above mentioned relationship between \( U \) and \( \mathbf{v} \) (Picard iteration).

Fuzzy c-means algorithm allows for allocation of data to several classes simultaneously varying degrees. Mathematically, the fuzzy c-means functionals look similar to (1)

\[
J_m(U, \mathbf{v}) = \frac{K}{m} \sum_{k=1}^{K} \sum_{i=1}^{m} u_{ik}^m \| \mathbf{x}_i - \mathbf{v}_k \|_a, \quad m \geq 1, \quad (2)
\]

but the partition \( U \) is now fuzzy:

\[
u_{ik} \in [0,1], \quad \zeta \sum_{i=1}^{m} \nu_{ik} = 1, \quad \text{for } \forall k.
\]

The weights in (2) are the \( m \)-th powers of membership of \( \mathbf{x}_i \) in cluster \( i \). Case \( m=1 \) corresponds to hard partition.

Fuzzy c-means method consists in minimising (2), when the norm is the squared Euclidean norm, \( I \) norm the method is known as k-median algorithm [11]. Replacing the Euclidean norm with Mahalanobis distance allows one to recognise clusters of elliptical shape [3, 4]. Distance to parametrically defined curves and surfaces can also be chosen as the norm in (2), in such cases the method is known as c-shells algorithm [4]. It is possible to introduce constraints into c-varieties algorithm, as proposed in [7] by fixing certain parameters at a priori known values. It is also possible to mix several prototypes in one model (c-mixed prototypes) [7].

In all these cases, the functional (2) (or its modification) is not convex, and it possesses many suboptimal local minima. This problem is known (e.g., [1, 2]), and it was addressed in [12-15], where stochastic global techniques were used.

If we consider \( J_m(U, \mathbf{v}) \) as the function of either \( U \) or \( \mathbf{v} \) (i.e., when the other variable is fixed), then it becomes a convex function, and the solution can be found explicitly, as the necessary optimum condition [2]

\[
u_{ik} = \left( \frac{1}{\sum_{j=1}^{K} \left( \frac{x_i - v_j}{\| x_i - v_j \|_a} \right)^{m-1}} \right)^\alpha, \quad (3a)
\]

\[
v_j = \sum_{i=1}^{L} u_{ik}^m x_i / \sum_{i=1}^{m} u_{ik}^m. \quad (3b)
\]

The c-means family of algorithms is simply Picard iteration through the necessary optimum condition. This method is called alternate minimization. Only local convergence can be demonstrated, and the solution is very sensitive to the initial partition. Starting c-means algorithm from several randomly chosen points and choosing the best solution is a frequently used technique, but the outcome can still be very poor, as illustrated in [10]. The authors attempted to partition 2500 data points into 21 clusters using c-means algorithm. They took 10000 randomly chosen initial points and obtained 9874 different suboptimal solutions. This experiment illustrated that c-means is very far from being a robust technique.

### III. Clusters with multiple control points

To be able to recognise clusters of arbitrary shape, not defined a priori, we allow clusters to have several prototype elements (no longer their centroids) \( \mathbf{v}_i^d, d = 1, \ldots, D \). The measure of dissimilarity of the data \( x_k \) to prototype elements of the cluster \( i \) is the distance from \( x_k \) to the set

\[
V_i = \left\{ \bigcup_{a=1}^{D} x_k \right\}, \quad d_i = d(x_k, V_i) = \min_{a=1}^{D} \| x_k - v_k^d \|_a.
\]

Fig. 1 illustrates clusters with multiple control points. It is clear from the picture, that positions of the control points determine the shape of the clusters. This approach is different from c-shells family of algorithms, where the shapes are specified parametrically in advance. It is also different from c-mixed prototypes, where several parametrically defined prototypes are mixed in the same model [7]. Being non-parametric, it is a more general approach, which requires no prior knowledge of the problem at hand. Shape of one cluster is independent of the shapes of other clusters, hence heterogeneous data groupings can be handled.

The WcSS functional now takes the form

\[
J_m(U, V) = \frac{K}{m} \sum_{k=1}^{K} \sum_{i=1}^{m} u_{ik}^m \min_{a=1}^{D} \left( \| x_k - v_k^d \|_a \right), \quad m \geq 1. \quad (4)
\]

Its minimization is a non-convex and non-smooth mathematical programming problem, and like (2), the functional (4) will have many local extrema. First order optimum conditions will no longer produce centroids of clusters as their prototype elements \( v \) (3b). However, conditions (3a) still apply, since they are nothing but the stationary points of the Lagrangian

\[
F(\lambda, u_{ik}) = \sum_{i=1}^{L} u_{ik}^m s_k^2 - \lambda \sum_{i=1}^{m} u_{ik} - \sum_{j=1}^{D} d_{jk}
\]

when \( d_{jk} \) are fixed. The optimum is found by minimising

\[
J_m(V) = J_m(U(V), V), \quad \text{with}
\]

\[
u_{ik} = \left( \frac{d_{ik}}{\sum_{j=1}^{D} d_{jk}} \right)^{\frac{1}{m-1}}, \quad \zeta \sum_{j=1}^{D} \left( \min_{a=1}^{D} \| x_k - v_k^d \|_a \right)^{\frac{1}{m-1}} = 1
\]

We leave the problem of minimization of (4) till the next section, and discuss now some of the properties of (4) and its possible modifications.

Our first remark with respect to (4) is that in its present form it has several global minima due to its symmetry with respect to \( v_k^d \). This is not unexpected, because even a simpler WcSS functional (2) possesses c1 global minima due to its symmetry with respect to permutations of elements of \( v \). Indeed, if we swap \( v_j \) and \( v_j \) in (2) and accordingly elements of \( U \), the value of \( J_m(U, V) \) will remain the same.

The worse news is that (4) it is also symmetric with respect to \( v_k^d \) and \( v_j^d \), taken from two different clusters \( i \) and \( j \). This is illustrated on Fig. 2. This means that even if we find a global minimum of (4), it will not necessarily correspond to a good data partition.
The cause of this problem is that in its present formulation, the cluster prototypes \( \mathbf{V}_i^\alpha \) are not required to be close to each other, but could be located anywhere. We propose here a remedy to this problem.

The first approach is to modify (4) to penalise \( \mathbf{V}_i^\alpha \) that are far from each other. The modified functional takes the form

\[
J_m(U(V), V) = \sum_{i} \sum_{\alpha} w_{ik} m \min_{\alpha} \left( \left\| \mathbf{x}_k - \mathbf{V}_i^\alpha \right\| \right) + p \max_{\alpha} \min_{i} \left\| \mathbf{V}_i^\alpha - \mathbf{V}_j^\beta \right\|, \quad m \geq 1,
\]

where \( p \) is the penalty parameter. Since the added term does not depend on \( U \), condition (5) remains valid. Furthermore, if the distance between control points is taken in the data induced metric [9], it would ensure that these points are connected (by data points lying in between).

The second approach is to define the distance to the set \( \mathbf{V}_i \) differently from \( \min_{\alpha} \left\| \mathbf{x}_k - \mathbf{V}_i^\alpha \right\| \). Generalized mean function can be used:

\[
d(x, \mathbf{V}_i) = f^{-1} \left( \sum_{\alpha} a_{\alpha} f \left( \left\| x - \mathbf{V}_i^\alpha \right\| \right) \right), \sum_{\alpha} a_{\alpha} = 1
\]

where \( f \) is a monotone function. When \( f(x) = x \), we obtain the usual weighted mean,

\[
d(x, \mathbf{V}_i) = \sum_{\alpha} a_{\alpha} \left\| x - \mathbf{V}_i^\alpha \right\|, \sum_{\alpha} a_{\alpha} = 1.
\]

When \( f(x) = 1/x \) and \( p \to \infty \) we obtain

\[
d(x, \mathbf{V}_i) = \left\| \mathbf{x}_k - \mathbf{V}_i^\alpha \right\|.
\]

Hence, (4) translates into

\[
J_m(U(V), V) = \sum_{i} \sum_{\alpha} w_{ik} m \left[ \sum_{\alpha} a_{\alpha} f \left( \left\| \mathbf{x}_k - \mathbf{V}_i^\alpha \right\| \right) \right].
\]

The use of generalized means as the distance function implicitly relates \( \mathbf{V}_i^\alpha \) of the same cluster \( i \) and penalises clusters in which the control points are far apart. Hence the symmetry between \( \mathbf{V}_i^\alpha \) and \( \mathbf{V}_j^\beta \) from different clusters is broken, and we can proceed with minimization of \( J_m(U(V), V) \).

The way \( U \) depends on \( V \) does not change, and we use

\[
u_{ik}(V) = \sum_{j \in \mathcal{K}(i)} \left( d_{ik}^{-\gamma} \right)^{(m-1)}
\]

with the appropriate expressions for \( d_{ik} \).

The symmetry between \( \mathbf{V}_i^\alpha \) and \( \mathbf{V}_j^\beta \) of the same cluster \( i \) is not that desirable, because in this case various global minima correspond to exactly the same partition. However, this presents annoying problems at the minimization stage, when the (global) algorithm tries to identify several global minima at once. This symmetry can be removed by ordering \( \mathbf{V}_i^\alpha \) in some way, say with respect to the first coordinate. Then the domain of \( \mathbf{V}_i^\alpha \) becomes the intersection of a hypercube and a simplex, and the restriction for simplex can be dealt with using an appropriate transformation of variables.

IV. CUTTING ANGLE METHOD OF GLOBAL OPTIMIZATION

Global minimization of \( J_m(U(V), V) \) in (6) or (7) is a very difficult problem. The functional is non-convex, non-smooth and possesses many local minima and stationary points. Local descent techniques for non-smooth functions (such as discrete gradient method [16]) can be applied, but they may lead to suboptimal solutions.

Global optimization methods aim at finding the global minimum of the objective function. There are two broad categories: stochastic and deterministic approaches [17, 18]. Stochastic techniques are better known: random search, simulated annealing and genetic algorithms. These methods have been successfully used in clustering problems [10, 13, 14]. We consider the other category: deterministic methods.

Deterministic global optimization is an NP-hard problem. Several techniques are available: cutting plane, branch and
bound, tabu search, smoothing methods, etc. [17, 18]. All these methods suffer from increasing combinatorial complexity for greater number of variables, and their practical applicability is usually limited to small size problems. We consider a recently developed method of cutting angle [19, 20]. This method constructs a piecewise linear saw-tooth cover of the objective function, which always underestimates it. The nonlinear optimization problem is substituted with a series of max-min type auxiliary problems, and the global minima of the auxiliary problems converge to the global minimum of the objective function. The crucial step of the cutting angle algorithm is minimization of the auxiliary function. It can be formulated as a combinatorial problem [20, 21]. The reason we chose the cutting angle method is the availability of a fast algorithm which can handle high dimensional problems in a reasonable time [21].

Despite the availability of a fast algorithm, the convergence of cutting angle method is very slow, and this is most noticeable in higher dimensional problems (n=10-20). This is due to the fact that the optimization problem is NP-hard. Simulated annealing and genetic algorithms are also used to solve this difficult problem, however we cannot directly compare them. Deterministic techniques are qualitatively different from stochastic techniques because they can guarantee (in principle) that the global minimum has been found, and can give an estimate on the lower bound of the global minimum. In contrast, stochastic methods converge in probabilistic sense it is up to the end user to decide whether a guaranteed global minimum is worth computational effort.

Combination of the cutting angle algorithm with local descent methods (e.g., discrete gradient method [16]) is a feasible alternative, allowing one to accelerate the convergence of the cutting angle at the last iterations.

V. CONCLUSION

We have presented several extensions of the WGS algorithm, under the assumption that clusters may have several prototypes. This approach allows one to represent clusters of arbitrary complex shape, not fixed a priori. The price for this flexibility is the necessity to solve a difficult non-convex optimization problem. One of the necessary condition of the minimum (3a) holds. The optimization is performed with respect to the coordinates of cluster prototypes. It keeps the size of the problem bounded, and independent of the data.

Alternating optimization used in classical and fuzzy means family of algorithms is not appropriate, because it converges to suboptimal solutions. Recently developed array of tools for global and non-smooth optimization helps tackling this difficult optimization problem. Cutting angle method is directly applicable to clustering problem, and its combination with local discrete gradient method allows one to obtain results in short time on inexpensive PCs.

REFERENCES