Clustering based on the principle of finding centers and robust averaging functions of aggregation

Z M Shibzukhov1,2, M A Kazakov2

1Institute of Mathematics and Informatics of Moscow Pedagogical State University, Krasnoprudnaya str., 4, Moscow, Russia, 119435
2Institute of Applied Mathematics and Automation Kabardino-Balkarian Scientific Centre Russian Academy of Sciences, Shortanova str., 89a, Nalchik, Russia, 123610

e-mail: szport@gmail.com

Abstract. A new approach to robust clustering based on the search for cluster centers is proposed. It is based on minimizing the robust estimates of the averages and the sum of the functions of pseudo-distances to cluster centers. An algorithm of iterative reweighing type for finding cluster centres is proposed. Examples are given showing the stability of the method with respect to a large number of outliers.

1. Introduction

The basis of one classical approach for clustering is the search for cluster centers. The center of the cluster is the point from which the sum of the distances to all its points is minimal. The division into clusters is carried out according to a simple rule: the point refers to the cluster, to the center of which the distance is minimal.

We propose to expand this principle of clustering. To do this, we will use continuously differentiable averaging aggregation functions instead of the usual arithmetic sum and taking the minimum in determining the center of the cluster and for assigning the point to the cluster. This allows us to expand the approach and propose new algorithms for finding cluster centers proposed in [1].

2. Algorithm HCD

Let \( S \subset \mathbb{R}^n \) be an open subset, \( X = \{x_1, \ldots, x_N\} \subset S \) be a finite set that needs to be divided into \( K \) clusters.

Let \( d: S \times S \rightarrow \mathbb{R}_+ \) be a function of the generalized distance between points from \( S \), which, by definition, satisfies the following requirements:
- \( d_c(x) = d(x, c) \) – strictly convex and twice differentiable on \( S \);
- for any \( c \in S \): \( \lim_{\|x\| \to \infty} d_c(x) = \infty \).

We define the classical method of splitting into clusters. Let \( C_1, \ldots, C_K \) be a partition of the set \( X \) into \( K \) clusters, \( y_j(x) \) be the characteristic function of the \( j \)-th cluster, \( 1 \leq j \leq K \):

\[
y_j(x) = \begin{cases} 
1, & \text{if } x \in C_j \\
0, & \text{else.}
\end{cases}
\]
Deterministic characteristic functions of clusters are completely defined by their centers. They are usually defined as follows:

\[ y_j(x) = \begin{cases} 
1, & \text{if } d(x, c_j) = D(x, c_1, \ldots, c_K) \\
0, & \text{else},
\end{cases} \]

where

\[ D(x, c_1, \ldots, c_K) = \min\{d(x, c_1), \ldots, d(x, c_K)\}. \]

That is, the point refers to those clusters, the distance to which is minimal.

The task of finding clusters, following [1], can be formulated as an optimization problem:

\[ c_1^*, \ldots, c_K^* = \arg \min_{c_1, \ldots, c_K} Q(c_1, \ldots, c_K), \]

(1)

where

\[ Q(c_1, \ldots, c_K) = \sum_{k=1}^{N} \nu_k D(x_k, c_1, \ldots, c_K), \]

the value of \( \nu_k \geq 0 \) reflects the significance of the \( k \)-th point, \( \nu_1 + \cdots + \nu_N = 1 \).

Note that in this formulation, the weighted average is used instead of the usual sum or weighted sum. This is not a hindrance when the number of points \( N \) and the weights of the points are given numerical parameters, since any weighted sum is easily replaced by its equivalent weighted average.

Since

\[ \nabla c_j Q = \sum_{k \in I_j} \nu_k \nabla c_j d_k(c), \]

where \( d_k(c_j) = d(x_k, c_j) \), \( I_j = \{k : x_k \in C_j\} \), then the center of the \( j \)-th cluster is the solution to the following optimization problem:

\[ c_j = \arg \min_c \sum_{k \in I_j} \nu_k d_k(c). \]

The classical clustering algorithm based on the search for cluster centers HCD (Hard Clustering with Distance-like functions) is an iterative process of refining the position of the centers \( c_1, \ldots, c_K \). It is a kind of descent algorithms with alternative directions that correspond to cluster centers. The pseudocode of the algorithm can be written in the following form:

**procedure** HCD(\( \{x_1, \ldots, x_N\}, \{c_1^0, \ldots, c_K^0\} \))

\[
c_1, \ldots, c_K \leftarrow c_1^0, \ldots, c_K^0
\]

repeat

\[
\text{for all } j = 1, \ldots, K \text{ do}
\]

\[
I_j = \{k : y_j(x_k) = 1\}
\]

\[
c_j \leftarrow \arg \min_c \sum_{k \in I_j} \nu_k d_k(c),
\]

end

\[
\text{until the positions of the centers are stabilized}
\]

return \( c_1, \ldots, c_K \)

end

Usually \( \nu_k = 1/N \). Non-standard sets of weights can occur indirectly in the framework of the robust clustering procedure, when \( d(x, c) = \varrho(\|x - c\|^2) \).

Robust clustering can be obtained under certain conditions, when \( \varrho(r) \) grows much slower than the linear function.

For example, when \( \varrho(r) = \sqrt{r} \), \( \varrho(r) = (1 + r)^y - 1 (y < 1/2) \) or even \( \varrho(r) = \ln(1 + r) \).

The center search algorithm can be thought of as an iterative weighting procedure based on a weighted version of the KMeans algorithm.

The pseudocode of this IR-KMeans (Iteratively Reweighted K-Means) algorithm can be written as follows:

**procedure** IR-KMeans(\( \{x_1, \ldots, x_N\}, \{c_1^0, \ldots, c_K^0\} \))

\[
c_1, \ldots, c_K \leftarrow c_1^0, \ldots, c_K^0
\]

repeat
\[ I_j = \{ k : y_j(x_k) = 1 \}, j = 1, ..., K \]
\[ v_{kj} = \varphi(\| x_k - c_j \|), k \in I_j \]
\[ c_j = \frac{\sum_{k \in I_j} v_{kj} x_k}{\sum_{k \in I_j} v_{kj}}, j = 1, ..., K \]

until the positions of the centers are stabilized

return \( c_1, ..., c_K \)

end

Here \( \varphi(r) = \varphi'(r)/r \).

3. M-means

In this article, we consider generalizations of this formulation of the problem, based on the fact that the shift of the centers of the desired clusters is manifested.

Let \( \rho(r) \) be a convex function. We define the M-average as the solution to the following problem:

\[ M_\rho(r_1, ..., r_m) = \arg \min_{s} \sum_{j=1}^{m} \rho(r_j - s). \]

If \( \rho(r) \) is a strictly convex function, then \( M_\rho \) is an averaging aggregating function [6,7]. Most of the known functions for calculating the empirical average can be represented as the M-average. If there is \( \rho''(r) \), then

\[ \frac{\partial M_\rho}{\partial r_j} = \frac{\rho''(r_j - \bar{r})}{\rho''(r_1 - \bar{r}) + ... + \rho''(r_m - \bar{r})}, \]

where \( \bar{r} = M_\rho(r_1, ..., r_m) \). At the same time, \( \frac{\partial M_\rho}{\partial r_1} + ... + \frac{\partial M_\rho}{\partial r_m} = 1 \).

For any \( M_\rho \) and strictly monotonic function \( h \), one can define an averaging aggregating function \( M_\rho^h \):

\[ M_\rho^h(r_1, ..., r_m) = h^{-1}(M_\rho(h(r_1), ..., h(r_m))). \]

It is the solution to the minimization problem:

\[ M_\rho^h(r_1, ..., r_m) = \arg \min_{s} \sum_{j=1}^{m} \rho(h(r_j) - h(s)). \]

If the function \( \rho \) and \( h \) have second order derivatives, then

\[ \frac{\partial M_\rho^h}{\partial r_j} = \frac{h'(r_j)}{h'(\bar{r})} \frac{\partial M_\rho}{\partial z_j}, \]

where \( z_1 = h(r_1), ..., z_m = h(r_m), \bar{z} = h(\bar{r}) \).

A typical example is the Kolmogorov average:

\[ M^h(r_1, ..., r_m) = h^{-1}\left( \sum_{j=1}^{m} h(r_j) \right). \]

In this case, \( \frac{\partial M^h}{\partial r_j} = \frac{1}{m} \frac{h'(r_j)}{h'(\bar{r})} \).

4. Algorithm IR-HCD-M1

Consider the first generalization by replacing the weighted arithmetic average by \( M_\rho \):

\[ Q(c_1, ..., c_K) = M_\rho(D(x_1), ..., D(x_K)). \]

Such a replacement makes sense when there are outliers among the points that can significantly shift the value of the functional if there is a sufficiently large number of them. Because of such a shift, the shift of the centers of the desired clusters is manifested.

Gradients \( Q \) over \( c_j \) (1 \( \leq j \leq K \)) has the form:
\[ \nabla_{c_j} Q = \sum_{k \in I_j} \frac{\partial M_\rho(D(x_1), \ldots, D(x_N))}{\partial r_k} \nabla_{c_j} d_k(c_j), \]

where \( I_j = \{ k : y_{kj} = 1 \} \).

To solve the problem of finding centers that minimize the objective functional, you can construct a descent procedure in alternative directions - gradients \( \nabla_{c_j} Q \). It is also a type of iterative reweighting procedure, differing from the IR-KMeans algorithm in the method of weights recalculation. The pseudocode can be listed as follows:

```plaintext
procedure IR-HCD-M1(\{x_1, \ldots, x_N\}, \{c_1^0, \ldots, c_K^0\})
    c_1^0, \ldots, c_K^0 \leftarrow \nabla_{c_j} Q
    repeat
        for all \( j = 1, \ldots, K \) do
            for all \( k \in I_j \) do \( \nu_k = \frac{\partial M_\rho(D(x_1), \ldots, D(x_N))}{\partial r_k} \)
            \( c_j \leftarrow \arg\min_c \sum_{k \in I_j} \nu_k d_k(c) \)
        end for
        until the positions of the centers are stabilized
    return c_1^0, \ldots, c_K^0
end
```

At the end of the algorithm, we obtain the weights of points \( \nu_1^*, \ldots, \nu_N^* \). Under certain conditions, the centers found are also a solution to problem (1)-(2) with weights \( \nu_1^*, \ldots, \nu_N^* \).

If \( d(x, c) = \| x - c \|_2^2 \), then the IR-HCM-M1 algorithm becomes an analogue of IR-KMeans, in which the weights \( \nu_{kj} = \frac{\partial M_\rho(d_{k1}, \ldots, d_{kN})}{\partial r_j} \), where \( d_{kj} = \| x_k - c_j \|_2^2 \).

![Figure 1](image_url). The use of KMeans and IR-KMeans-M1 algorithms in Example 1.

If we choose \( \rho \) so that \( M_\rho \) determines the robust function of the M-average, then we can construct a robust clustering algorithm.
4.1. Illustrative examples

To illustrate the capabilities of the IR-KMeans-M1 algorithm, we consider a number of examples. They clearly demonstrate its ability to find centers that lie quite close to real cluster centers under conditions where data contain outliers or when not all cluster centers are searched for, but only some of them. If in the first case, the cause of the displacement of the found centers are emissions, in the second case, the cause of the displacement is excessive clusters.

Example 1.

This example compares the use of KMeans and IR-KMeans-M1 algorithms with the M-average $M_{p,a,e}$, where $\rho_e(r) = \sqrt{\varepsilon^2 + r^2}$, $\varepsilon = 0.001$, to search for the centers of two clusters. The points of the artificially generated data set belong to two clusters with a normal distribution of the distances of the cluster points from their center. Additionally, a cloud of “outliers” was added, which is located far from the centers of the first two clusters and has a large dispersion in the distribution of the distance of points from its center.

In the first case, the first two clusters contain 200 points, the “outliers” cloud contains 100 points (33%). The conventional KMeans method provides cluster centers that are biased towards outliers. The algorithm IR-KMeans-M1 with $\alpha = 0.4$. In the second case, there are already 200 points in the “outliers” cloud (50%), and the scatter of its points is slightly increased. For robust clustering, the IR-KMeans-M1 algorithm with $\alpha = 0.3$ was applied. The found centers also lie within their clusters, which makes it possible to cut out emissions using the distribution of values $\{D(x_k, e'_1, e'_2); k = 1, ..., N\}$. The results are presented in Figure 1.

Example 2.

In this example, 8 clusters are randomly generated, so that in each cluster the points are distributed according to the normal law, the clusters do not overlap. The IR-KMeans-M1 algorithm is used for clustering for $K = 2, 3, 4, 5, 6, 7$ clusters, respectively. In this case, M-means $M_{p,a,e}$ were used, where $\rho_e(r) = \sqrt{\varepsilon^2 + r^2} - \epsilon$, $\varepsilon = 0.001$ with $\alpha = \alpha = 0.15, 0.20, 0.25, 0.30, 0.35, 0.40$, respectively. In all cases, the centers found fall inside the clusters. The results are presented in Figure 2.

5. Algorithm IR-HCD-MM

Note that the averaging function $\min$ can be approximated using M-means. For example, with the help of Kolmogorov averages:

\[
M_p[r_1, ..., r_m] = \left(\frac{r_1^p + ... + r_m^p}{m}\right)^{1/p},
\]

\[
M_p[r_1, ..., r_m] = \frac{1}{p} \ln \left(\frac{e^{-pr_1} + ... + e^{-pr_m}}{m}\right).
\]

Where\n
\[
\lim_{p \to +\infty} M_p[r_1, ..., r_m] = \min\{r_1, ..., r_m\}.
\]

Another approximation method can be constructed on the basis of using the differentiable $\alpha$-quantile approximation for sufficiently small $\alpha$. It can be constructed using the function $M_{p,a,e}$, where

\[
\rho_{a,e}(r) = \begin{cases}
\alpha \rho_e(r), & \text{if } r > 0 \\
\frac{\alpha}{2} \rho_e(0_+) + \frac{1-\alpha}{2} \rho_e(0_-), & \text{if } r = 0 \\
(1-\alpha) \rho_e(r), & \text{if } r < 0,
\end{cases}
\]

$\rho_e(r)$ is a function such that

1. $\lim_{\varepsilon \to 0} \rho_e(r) = |r|$;
2. $\lim_{\varepsilon \to 0} \rho'_e(r) = \text{sign } r$;
3. $\lim_{\varepsilon \to 0} \rho''_e(r) = \delta(r) - \text{Dirac } \delta$-function.

For example:

1. $\rho_e(r) = \sqrt{\varepsilon^2 + r^2} - \varepsilon$;
2. $\rho_e(r) = |r| - \varepsilon \ln(|r|) + \varepsilon \ln \varepsilon$.

Such approximations are preferable in cases where the point cannot be reliably attributed to the center of one cluster due to the fact that another one or several centers of other clusters are very close.
Using the approximations of the minimum using $M^\chi$ for some function $\chi = \rho_{a,e}$ we construct the second generalization (2), when

$$D(x) = M^\chi[d(x, c_1), ..., d(x, c_K)].$$

Combine both methods outlined above. Now the task of finding centers is:

$$c_1^*, ..., c_K^* = \arg\min_{c_1, ..., c_K} Q(c_1, ..., c_K),$$

where

$$Q(c_1, ..., c_K) = M_{\rho}[D(x_1), ..., D(x_N)].$$

Gradient $Q$ over $c_j$ ($1 \leq j \leq K$) has the form:

$$\nabla c_j Q = \sum_{k=1}^N \frac{\partial M_{\rho}[d_{kj}, ..., d_{Nj}]}{\partial r_k} \frac{\partial M^\chi[d_{k1}, ..., d_{kK}]}{\partial r_j} \nabla c_j d(x_k, c_j),$$

where $d_{kj} = d(x_k, c_j)$.

The pseudo-code of the iterative re-weighting procedure for searching centers $c_1^*, ..., c_K^*$ can be written as follows:
procedure IR-HCD-MM\((\{x_1, ..., x_N\}, \{c_1^0, ..., c_K^0\})\)

\(c_1, ..., c_K \leftarrow c_1^0, ..., c_K^0\)

repeat

for all \(k = 1, ..., N\) do

\(v_{k1}, ..., v_{kK} = \text{grad} M_X(d_k(c_1), ..., d_k(c_K))\)

for all \(j = 1, ..., K\) do

\(v_{1j}, ..., v_{Nj} = \text{grad} M_P(d_1(c_j), ..., d_N(c_j))\)

for all \(j = 1, ..., K\) do

\(c_j \leftarrow \arg\min_c \sum_{k=1}^N v_{kj} v_{kj} d_k(c)\)

until the positions of the centers are stabilized

return \(c_1, ..., c_K\)

End

Note that \(v_{k1} + ... + v_{kK} = 1\). This allows interpreting the values \(v_{k1}, ..., v_{kK}\) as the degree of belonging of the point \(x_k\) to the classes with labels \(1, ..., K\), respectively, obtaining a fuzzy division into clusters. At the same time, the fuzzy cluster membership function is built on the basis of the selected differentiable averaging function \(M_X\) as follows:

\[y_1(x), ..., y_K(x) = \text{grad} M_X(d(x, c_1), ..., d(x, c_K))\]

The IR-HCM-MM algorithm represents a version of the generalization of the SKM algorithm [1]. In the same place (in [1]) it was shown that SKM generalizes already known clustering algorithms, such as FCM [8], EM [9], DA [10], Bergman Soft Clustering [11]. Accordingly, if instead of \(M_X\) we use Kolmogorov averages, then we can also construct an algorithm, which SKM generalizes directly.

6. Conclusion

Thus, the use of \(M\) -averages made it possible to construct new procedures for finding cluster centers, which generalize HCD and SKM, making it possible to use a wide range of methods for finding an average value for both clear and fuzzy assignment of points to clusters. The proposed method are resistant to a large number of outliers. In addition, they have the ability to find centers of real clusters in cases where a given number of clusters is less than the number of real clusters. Similar properties of resistance to emissions were obtained for regression and classification problems solved on the basis of the principle of minimization of robust empirical risk [5]. The results show that the application of the proposed algorithms allows us to solve clustering problems [12] in a large number of outliers.

7. References

[1] Teboulle M A 2007 Unified Continuous Optimization Framework for Center-Based Clustering Method Journal of Machine Learning Research 8 65-102

[2] Mesiar R, Komornikova M, Kolesarova A and Calvo T 2008 Aggregation functions: A revision Fuzzy Sets and Their Extensions: Representation, Aggregation and Models (Springer Berlin, Heidelberg)

[3] Grabich M, Marichal J-L and Pap E 2009 Aggregation Functions Series: Encyclopedia of Mathematics and its Applications 127 (Cambridge University Press)

[4] Shibzukhov Z M 2017 About principle of empirical risk minimization based on averaging aggregation functions Doklady Mathematics 96 494-497

[5] Shibzukhov Z M, Kazakov M A and Dimitrichenko D P 2018 Minimisation of robust estimates of the sums of parametrised functions CEUR Workshop Proceedings 2212 10-16

[6] Calvo T and Beliakov G 2010 Aggregation functions based on penalties Fuzzy Sets and Systems 161(10) 1420-1436

[7] Beliakov G, Sola H and Calvo T 2016 Practical Guide to Averaging Functions (Springer)

[8] Bezdek J C 1981 Pattern Recognition with Fuzzy Objective Function Algorithms (Plenum Press, N. York)

[9] Duda R O, Hart P E and Stork D G 2001 Pattern Classification (John Wiley & Sons, Inc.)

[10] Rose K A, Gurewitz E and Fox C G 1990 deterministic annealing approach to clustering
Pattern Recognition Letters 11(9) 589-594

[11] Banerjee A, Merugu S, Dhillon I S and Ghosh J 2005 Clustering with Bregman Divergences Journal of Machine Learning Research 6 1705-1749

[12] Rycarev I A, Kirsh D V and Kupriyanov A V 2018 Clustering of media content from social networks using bigdata technology Computer Optics 42(5) 921-927 DOI: 10.18287/2412-6179-2018-42-5-921-927

Acknowledgments
This work was supported by the RFBR grant 18-01-00050.