Consiglio Nazionale delle Ricerche
Istituto di Calcolo e Reti ad Alte Prestazioni

Clustering by Spheres: Preliminary Results

Annabella Astorino[1], Matteo Avolio[2],
Annamaria Canino[2], Teresa Crupi[2],
Antonio Fuduli[2]

RT-ICAR-CS-22-01 **Febbraio 2022**

1) ICAR-CNR, Rende, Italy
2) Department of Mathematics and Computer Science, University of Calabria, Rende, Italy

Consiglio Nazionale delle Ricerche, Istituto di Calcolo e Reti ad Alte Prestazioni (ICAR)
– Sede di Cosenza, Via P. Bucci 8-9C, 87036 Rende, Italy, URL: *www.icar.cnr.it*
– Sezione di Napoli, Via P. Castellino 111, 80131 Napoli, URL: *www.icar.cnr.it*
– Sezione di Palermo, Via Ugo La Malfa, 153, 90146 Palermo, URL: *www.icar.cnr.it*

Clustering by spheres: preliminary results

Annabella Astorino[1], Matteo Avolio[2], Annamaria Canino[2], Teresa Crupi[2], and Antonio Fuduli[2]

[1] *ICAR - National Research Council - 87036, Rende, Italy*
[2] *Department of Mathematics and Computer Science - University of Calabria - 87036, Rende, Italy*

Abstract

In this work we propose to extend the spherical separation approach, amply used in supervised classification, to the unsupervised case, by assigning each datum to a suitable sphere. Since such spheres are computable by facing a nonsmooth nonconvex optimization problem difficult to solve, our main idea consists in designing a heuristics approach based on solving successive linear programs aimed at providing the radii of the clustering spheres, whereas the centers are fixed in advance as the barycenters of each current cluster, similarly to the strategy adopted by the well known K-Means algorithm.

Preliminary numerical results on some clustering problems drawn from the literature are reported.

Keywords: Unsupervised Classification, Clustering, Spherical Separation, K-Means

1 Introduction

Nowadays, we produce and collect huge amounts of data, for example via social media, online transactions and network sensors. This exponential growth of available data makes the entire decision-making process very complex starting from the data representation itself. Indeed, there is a real need to analyze data efficiently for discovering regular patterns in them and making better decisions. To answer this demand, it is important to develop new computational approaches and tools in data analysis.

Optimization plays a crucial role in this field as many mathematical programs have been proposed to model data mining problems, where the aim is to discover and extract previously unknown information from data [21, 26, 27]. The data mining process involves several distinct activities such as data representation, supervised data classification, association rules, feature selection and extraction, regression analysis and clustering.

In particular, in this work we focus on clustering. It is an unsupervised classification technique, where the aim is to partition a data set into subsets (clusters) of similar objects in order to recognize patterns in data. The idea is that each cluster consists of objects that are similar amongst themselves and dissimilar compared to objects of other clusters. Clustering analysis has many applications in various fields such as in medical sciences [5, 31], computer sciences [12] and economics [16], and consequently there are different approaches and algorithms for it in function of the problems to be solved. A possible classification is the following:

- partitioning clustering;
- hierarchical clustering;
- density-based clustering;
- grid-based clustering;
- model-based clustering;
- constraint-based clustering.

In the partitional clustering methods, the objective is to find a partition of the objects that optimizes some predefined clustering criterion [14, 23]. These methods can be divided into two subclasses: hard partitional clustering, where each object belongs to only one cluster and soft partitional clustering, where each object may belong to more than one cluster.

The hierarchical clustering methods create a hierarchical decomposition of the given data set [14, 15, 20]. Based on how the hierarchical decomposition is formed, these methods are distinguished in agglomerative and divisive approaches. The agglomerative clustering [8] is a bottom-up approach that starts with each object forming a cluster of its own. It keeps on merging the clusters that are close each to other, as one moves up the hierarchy until all the groups are merged into one or until a termination condition holds. On the other hand, divisive clustering [6] is a top-down approach. Initially, all the objects belong to one cluster and the division is performed recursively, as one moves down the hierarchy until each object is a cluster or a termination condition occurs.

Density-based clustering methods focus on the notion of density [18, 24]. They identify distinctive clusters, based on the idea that a cluster in a data space is a contiguous region of high point density, separated each from the other by contiguous regions of low point density. The data points in the separating regions of low point density are typically considered noise or outliers.

The grid-based clustering approach differs from the conventional clustering algorithms since it concerns not the objects but the space where the objects are located [7, 25, 32]. A typical grid-based clustering algorithm starts by partitioning the object space into a finite number of cells, i.e. it creates a grid structure on which all the operations for clustering are implemented. The benefit of this type of methods is the quick processing time, which is generally independent of the number of data objects.

Model-based clustering is a statistical approach [10, 11, 28]. The observed data are assumed to have been generated from a finite number of component models, i.e. probability distributions. Each component, that is initially unknown, determines the cluster to which the generated observation belongs and the key learning task is to calculate the parameters of its distribution.

Constraint-based clustering finds clusters that satisfy certain conditions, for example user expectations or given properties [19, 30]. In such sense the constraints provide an interactive communication way between the user and the clustering process.

In this work we present an hard partitional clustering technique based on spherical separation, which is a common strategy used in supervised classification.

Throughout the paper, we will use the following notation. Given a vector $x \in \mathbb{R}^n$, we indicate by $\|x\|$ its Euclidean norm and given any set $\mathcal{X}$, we denote by $|\mathcal{X}|$ its cardinality.

2 The model

Let

$$Y \triangleq \{y_1, \ldots, y_m\}$$

be a finite set of m points, such that $y_i \in \mathbb{R}^n$, $i = 1, \ldots, m$, and let k be an integer number such that $k < m$. The clustering problem consists of automatically partitioning the m points of the set Y into k subsets (clusters) Y_j, $j = 1, \ldots, k$, on the basis of their similarities.

For this purpose we propose to use a multisphere separation approach, based on constructing k spheres of the type:

$$S_j(x_j, R_j) = \{x \in \mathbb{R}^n \mid \|x - x_j\|^2 = R_j^2\}, \quad j = 1, \ldots, k,$$

where $x_j \in \mathbb{R}^n$ is the center of the sphere S_j and $R_j \in \mathbb{R}^+$ is the corresponding radius.

In particular, the criterion we propose is to cluster the points on the basis of their distance with respect to each sphere. In other words, a point $y_i \in Y$ belongs to the cluster Y_j if

$$\|y_i - x_j\|^2 - R_j^2 \leq \|y_i - x_r\|^2 - R_r^2 \quad \text{for any } r = 1, \ldots, k \text{ and } r \neq j,$$

i.e. if

$$\|y_i - x_j\|^2 - R_j^2 \leq \min_{\substack{1 \leq r \leq k \\ r \neq j}} \left\{ \|y_i - x_r\|^2 - R_r^2 \right\}. \tag{1}$$

Then, letting

$$d_{ij} \triangleq \|y_i - x_j\|^2, \quad i = 1, \ldots, m \text{ and } j = 1, \ldots, k$$

be the squared distance between the point y_i and the center of the sphere S_j, and assuming (1) as the clustering criterion for y_i, we say that the point y_i does not belong to the cluster Y_j if

$$d_{ij} - R_j^2 > \min_{\substack{1 \leq r \leq k \\ r \neq j}} \left\{ d_{ir} - R_r^2 \right\}.$$

As a consequence the error function, say e_{ij}, of y_i with respect to the cluster Y_j is defined as follows

$$\begin{aligned} e_{ij}(x_1, \ldots, x_k; R_1, \ldots, R_k) &\triangleq \max \left\{ 0, d_{ij} - R_j^2 - \min_{\substack{1 \leq r \leq k \\ r \neq j}} \{ d_{ir} - R_r^2 \} \right\} \\ &= \max \left\{ 0, d_{ij} - R_j^2 + \max_{\substack{1 \leq r \leq k \\ r \neq j}} \{ R_r^2 - d_{ir} \} \right\} \\ &= \max_{\substack{1 \leq r \leq k \\ r \neq j}} \left\{ 0, d_{ij} + R_r^2 - (d_{ir} + R_j^2) \right\}. \end{aligned} \tag{2}$$

Summing up, we obtain the following final clustering error function:

$$e(x_1, \ldots, x_k; R_1, \ldots, R_k) \triangleq \sum_{i=1}^{m} \sum_{j=1}^{k} e_{ij}(x_1, \ldots, x_k; R_1, \ldots, R_k),$$

which takes into account the clustering error of all the points of set Y with respect to every cluster.

To strengthen the clustering process, in addition to the minimization of the above function e, we propose to compute the clustering spheres by minimizing also the respective volumes, which is a common strategy used in supervised classification (see for example [3, 1, 2, 17]). Then we come out with the following optimization problem:

$$\min_{x_1, \ldots, x_k; R_1, \ldots, R_k} f(x_1, \ldots, x_k; R_1, \ldots, R_k), \tag{3}$$

where

$$f(x_1, \ldots, x_k; R_1, \ldots, R_k) \triangleq e(x_1, \ldots, x_k; R_1, \ldots, R_k) + C \sum_{j=1}^{k} R_j^2, \tag{4}$$

with C being a positive parameter, tuning the trade-off between the two above different objectives: the minimization of the clustering error and the minimization of the radii of the spheres.

Problem (3) is a difficult problem, due to the nonconvexity and the nonsmoothness of the objective function f, involving the nonsmooth and nonconvex functions e_{ij}s defined by (2).

3 A heuristic algorithm

In this section we propose a heuristic approach for solving problem (3). The underlying idea is only in part similar to the well known K-Means approach (see for example the survey [13]), since the computation of the centers x_js of the spheres are performed in a similar way as K-Means updates the centers of the clusters. In addition, differently from K-Means, in our approach an important role is played by the computation of the radii of the spheres (see relationship (1)), that in K-Means does not intervene.

In particular, the proposed overall algorithm is a variant of the Block Coordinate Descent approach [29], consisting in the computation of the optimal values of the centers x_js, when the radii R_js are fixed, and, vice-versa, of the optimal values of the radii when the centers are fixed. Unfortunately, tackling problem (3) by using BCD in its original form is not a practicable way from the computational point of view, because, even when all the radii R_js are fixed, problem (3) is still a difficult problem since it remains nonsmooth and nonconvex. Then, to overcome this difficulty, we have chosen to update, at each iteration of the algorithm, the centers of the spheres as the barycenters of the points currently belonging to the corresponding clusters, on the basis of the criterion expressed by relationship (1), which, in case $R_j = 0$ for any $j = 1, \ldots, k$, coincides with the clustering criterion used by K-Means.

On the other hand, whenever all the centers x_j, for $j = 1, \ldots, k$, are fixed, it is easy to see that problem (3) reduces to a linear program. In fact, in such case, introducing the auxiliary variables z_j, for $j = 1, \ldots, k$, and letting

$$z_j \stackrel{\triangle}{=} R_j^2 \geq 0 \quad j = 1, \ldots, k,$$

problem (3) becomes:

$$\left\{ \begin{array}{ll} \min\limits_{z} & \sum\limits_{i=1}^{m} \sum\limits_{j=1}^{k} \max\limits_{\substack{1 \leq r \leq k \\ r \neq j}} \{0, d_{ij} + z_r - (d_{ir} + z_j)\} + C \sum\limits_{j=1}^{k} z_j \\ & z_j \geq 0 \quad j = 1, \ldots, k, \end{array} \right. \tag{5}$$

which is equivalent to the following linear program:

$$\left\{ \begin{array}{lll} \min\limits_{z,\xi} & \sum\limits_{i=1}^{m} \sum\limits_{j=1}^{k} \xi_{ij} + C \sum\limits_{j=1}^{k} z_j & \\ & & \\ & \xi_{ij} \geq d_{ij} + z_r - (d_{ir} + z_j) & i = 1, \ldots, m \quad j = 1 \ldots, k \\ & & r = 1, \ldots, k \quad r \neq j \\ & z_j \geq 0 & j = 1, \ldots, k \\ & \xi_{ij} \geq 0 & i = 1, \ldots, m \quad j = 1 \ldots, k. \end{array} \right. \tag{6}$$

We come out with the following heuristic algorithm, where by I_j, $j = 1, \ldots, k$, we indicate the index set corresponding to the cluster Y_j such that:

$$Y_j = \{y_i \in Y \mid i \in I_j\}.$$

Moreover we indicate by $\bar{x}_j$ and $\bar{z}_j$, $j = 1, \ldots, k$, the center and the radius of the sphere S_j, available at the current iteration.

Algorithm 1: Sph-Clust^0

Input: $y_1, \ldots, y_m$; $C \geq 0$; $\bar{x}_1, \ldots, \bar{x}_k$
Output: $I_1, \ldots, I_k$

▷Initialization
1 **for** $j \leftarrow 1, \ldots, k$ **do**
2 $I_j \leftarrow \emptyset$
3 **repeat**
  ▷Computing the squared distances
4 **for** $j \leftarrow 1, \ldots, k$ **do**
5 **for** $i \leftarrow 1, \ldots, m$ **do**
6 $d_{ij} \leftarrow \|y_i - \bar{x}_j\|^2$
  ▷Computing the squared radii of the spheres
7 Solve problem (6) to compute $\bar{z}_j$, $j = 1, \ldots, k$
  ▷Updating the clusters
8 **for** $i \leftarrow 1, \ldots, m$ **do**
9 $j_i^* \leftarrow \arg\min_{j=1\ldots,k} \{d_{ij} - \bar{z}_j\}$
10 $I_{j_i^*} \leftarrow I_{j_i^*} \cup \{i\}$
  ▷Updating the centers of the spheres
11 **for** $j \leftarrow 1, \ldots, k$ **do**
12 $\bar{x}_j \leftarrow \frac{1}{|I_j|} \sum_{i \in I_j} y_i$
13 **until** *all the sets I_js, $j = 1, \ldots, k$, do not change*

Algorithm 1 (Sph-Clust^0) deserves some comments. First of all, it is easy to see that the algorithmic scheme is very similar to K-Means, apart from step 7, where the linear program (6) is solved to obtain the radii of the spheres (once the centers x_js have been fixed) and lines 8-10, where the clusters are formed on the basis of the criterion (1), involving the computation of the radii. Moreover, a crucial question is the choice of the initial centers (as in K-Means) together with to the choice of C, the latter tuning the trade-off between the clustering error and the minimization of the volume of the spheres. Note that, in case C is very large (take for example $C > m$, Theorem 1), Algorithm 1 coincides exactly with K-Means, since the linear program provides null radii for all the spheres, coming out with the same clustering criterion used by K-Means.

Since the objective function of problem (5) is nonnegative and the null solution is feasible, it is trivial to see that problem (5) admit an optimal solution, as formalized by the following lemma concerning the equivalent linear program (6).

Lemma 1. *Problem (6) admits an optimal solution.*

Proof. Since $\xi_{ij} \geq 0$, for any $i = 1, \ldots, m$ and $j = 1 \ldots, k$, and $z_j \geq 0$ for any $j = 1 \ldots, k$, taking into account the positivity of C, the objective function is bounded. Moreover the problem cannot be infeasible, since a feasible solution is:

$$\begin{cases} z_j = 0 & j = 1, \ldots, k; \\ \xi_{ij} = \max\{0, d_{ij} - \min\limits_{\substack{1 \leq r \leq k \\ r \neq j}} d_{ir}\} & i = 1, \ldots, m \quad j = 1 \ldots, k. \end{cases}$$

□

3.1 Choosing the weighting parameter

In this subsection we show some results based on the duality theory and concerning the choice of C, which plays a relevant role in Algorithm 1 since it is the main characterization with respect the standard K-Means algorithm.

The dual of problem (6) is the following:

$$\begin{cases} \max\limits_{\alpha} & \sum\limits_{i=1}^{m} \sum\limits_{j=1}^{k} \sum\limits_{\substack{r=1 \\ r \neq j}}^{k} (d_{ij} - d_{ir}) \alpha_{ijr} & & \\ & \sum\limits_{\substack{r=1 \\ r \neq j}}^{k} \alpha_{ijr} \leq 1 & i = 1, \ldots, m & j = 1 \ldots, k \\ & \sum\limits_{i=1}^{m} \sum\limits_{\substack{r=1 \\ r \neq j}}^{k} \alpha_{ijr} - \sum\limits_{i=1}^{m} \sum\limits_{\substack{r=1 \\ r \neq j}}^{k} \alpha_{irj} \leq C & j = 1, \ldots, k & \\ & \alpha_{ijr} \geq 0 & i = 1, \ldots, m & j = 1 \ldots, k \\ & & r = 1, \ldots, k & r \neq j. \end{cases} \tag{7}$$

For sake of the simplicity, in what follows we refer to the second group of constraints of the dual problem as the *C-constraints* and we introduce the following definition.

Definition 1 (Degenerate problem)**.** *The linear program (6) (or equivalently its dual (7)) is degenerate if there exist an index $i \in \{1, \ldots, m\}$ and a couple of indexes (j, r), with $j \in \{1, \ldots, k\}$, $r \in \{1, \ldots, k\}$ and $r \neq j$, such that $d_{ij} = d_{ir}$.*

Note that, when at the current iteration the problem is degenerate, applying an iteration of the standard K-Means would result in a non-unequivocal assignment of at least a point y_i to a cluster.

In the next theorems we provide two different conditions, respectively sufficient and necessary, for having $z_j = 0$, for any $j = 1, \ldots, k$, in the optimal solution to problem (6).

Theorem 1. *If $C > m$, any optimal solution to problem (6) is such that $z_j = 0$ for any $j = 1, \ldots, k$.*

Proof. By contradiction, if there exists an index $j \in \{1, \ldots, k\}$ such that $z_j > 0$, then by the complementary slackness, the jth C-constraint is satisfied as equality, i.e.:

$$C = \sum_{i=1}^{m} \sum_{\substack{r=1 \\ r \neq j}}^{k} \alpha_{ijr} - \sum_{i=1}^{m} \sum_{\substack{r=1 \\ r \neq j}}^{k} \alpha_{irj}. \tag{8}$$

Summing up over i the first group of the dual constraints, we have:

$$\sum_{i=1}^{m} \sum_{\substack{r=1 \\ r \neq j}}^{k} \alpha_{ijr} \leq m \quad j = 1 \ldots, k. \tag{9}$$

Combining (8) and (9) and taking into account the positivity of the variables α_{irj}, we obtain $C \leq m$, which is a contradiction.

□

On the basis of the above theorem, whenever $C > m$, Algorithm 1 reduces to the K-Means algorithm: in fact the assignment criterion of the points to the clusters (lines 8-10) coincides, at each iteration, with the assignment criterion adopted by the K-Means, since all the variables z_j, for $j = 1, \ldots, k$, are null.

Theorem 2. *If the linear program (6) is not degenerate and an optimal solution is such that $z_j = 0$ for any $j = 1, \ldots, k$, then*

$$C \geq \max_{1 \leq j \leq k} \{|\bar{D}_j| - (k-1)|D_j|\}, \tag{10}$$

where, for any $j = 1, \ldots, k$,

$$D_j \stackrel{\triangle}{=} \left\{ i \in \{1, \ldots, m\} \,|\, d_{ij} = \min_{1 \leq r \leq k} d_{ir} \right\}$$

and $\bar{D}_j$ is the complement of D_j.

Proof. If $z_j = 0$ for any $j = 1, \ldots, k$, by the equivalence between problems (5) and (6), for any couple of indexes (i, j), with $i = 1, \ldots, m$ and $j = 1 \ldots, k$, we have

$$\xi_{ij} = \max_{\substack{1 \leq r \leq k \\ r \neq j}} \{0, d_{ij} - d_{ir}\}.$$

Let the index i be fixed for sake of the simplicity. Since the problem is not degenerate, there exists a unique index p such that $\xi_{ip} = 0$, with $d_{ip} = \min_{1 \leq r \leq k} d_{ir}$ and such that

$$\xi_{ip} > d_{ip} - d_{ir}, \text{ for any } r \neq p.$$

Taking into account the above inequality, by the complementary slackness, we have

$$\alpha_{ipr} = 0 \text{ for any } r \neq p. \tag{11}$$

On the other hand, for any $q \neq p$, it holds $\xi_{iq} = d_{iq} - d_{ip} > 0$, which, by the complementary slackness, implies

$$\sum_{r \neq q} \alpha_{iqr} = 1. \tag{12}$$

Moreover we have

$$\xi_{iq} > d_{iq} - d_{ir} \text{ for any } r \neq p \text{ and for any } r \neq q,$$

which, by the complementary slackness, implies

$$\alpha_{iqr} = 0 \text{ for any } r \neq p \text{ and for any } r \neq q. \tag{13}$$

Combining (12) and (13), we have necessarily $\alpha_{iqp} = 1$.

In summary, we have shown that, in corresponding to any fixed index $i \in \{1, \ldots, m\}$, since q can assume $k-1$ values different from p, we have exactly $k-1$ variables α_{iqp} equal to 1, whereas by (11) and (13) the remaining $(k-1)^2$ dual variables are equal to zero.

For any $i = 1, \ldots, m$, each variable $\alpha_{iqp} = 1$ corresponds to $i \in D_p$ (since $\xi_{ip} = 0$) and to $i \notin D_q$ (since $\xi_{iq} > 0$) and acts with a positive sign on the qth C-constraint and with a negative sign on the pth C-constraint. Then, substituting the values of all the dual variables $\alpha_{iqp} = 1$ into the C-constraints, we obtain

$$C \geq |\bar{D}_j| - (k-1)|D_j| \text{ for any } j = 1, \ldots, k,$$

i.e.

$$C \geq \max_{1 \leq j \leq k} \{|\bar{D}_j| - (k-1)|D_j|\}.$$

□

On the basis of the result provided by Theorem 2, whenever at the current iteration of Algorithm 1 the linear program (6) is not degenerate, in order to guarantee an optimal solution characterized by at least a variable $z_j > 0$, with $j \in \{1, \ldots, k\}$, it is sufficient to choose C such that:

$$C < \max_{1 \leq j \leq k} \{|\bar{D}_j| - (k-1)|D_j|\}. \tag{14}$$

4 Preliminary numerical results

Algorithm 1, named Sph-Clust^0 (Spherical Clustering), has been implemented in Matlab (version R2018b) and has been run on a Windows 10 system, characterized by 16 GB of RAM and a 2.30 GHz Intel Core i7 processor.

Data set	k	m	n
Iris	3	150	4
Seeds	3	210	7
Wine	3	178	13
Thyroid	3	215	5
Breast	6	106	9
Accent	6	329	12
Glass	6	214	9
Vertebral Column	3	310	6
ECG5000	5	500	140
Lymphography	4	148	18

Table 1: Data sets

The code has been tested on ten multiclass problems, listed in Table 1. All the data sets have been download from the UCI repository [22], except EGC5000 which has been drawn from the UEA & UCR Time Series Classification Repository [4]. We remind that k denotes the number of clusters, m is the number of data (called also instances or points) and n is the number of features. Of course, even if in all the used test problems the class label of each point is known, this information has been exploited only to evaluate the results provided by the clustering algorithm.

As discussed in the previous section, a crucial point for Algorithm 1 (Sph-Clust^0) is to choose a suitable value of C in order to have at least one radius of the k spheres different from zero. Since condition (14) depends on the single iteration, in order to maintain the same value of C along the overall running of the algorithm, we have chosen C in the grid $\{1, 10, 100\}$, reporting in Table 2 the best result obtained for each data set in terms of purity [9].

About the choice of the initial centers $\bar{x}_1, \ldots, \bar{x}_k$ of the k spheres, we have set $\bar{x}_i = y_i$, for any $i = 1, \ldots, k$.

In Table 2 we compare our results with those ones provided by the K-Means algorithm, for which we have considered two different versions: the first one (Algorithm K-Means^0) has been run using the same starting point used for Algorithm Sph-Clust^0, whereas in the second version (Algorithm K-Means) the initial centers of the clusters have been randomly generated.

For the sake of completeness, in correspondence to the results provided by Algorithm Sph-Clust^0, we report also the number of iterations coinciding with the number of solved linear programs of the type (6).

Looking at the overall results, we can observe that Algorithm Sph-Clust^0 overcomes the standard K-Means on all the data sets, except ECG5000 where the behaviours are quite comparable. In particular our approach outperforms significantly on Breast and Glass (more than four percentage points), but also on Iris and Wine (more than two percentage points).

Data set	Sph-Clust^0		K-Means^0	K-Means
	%	iter	%	%
Iris	**92.00**	5	88.67	89.33
Seeds	**90.48**	4	89.05	89.52
Wine	**73.03**	5	68.54	70.22
Thyroid	**82.33**	12	80.47	75.35
Breast	**53.77**	10	43.40	45.28
Accent	**53.19**	21	52.89	52.28
Glass	**63.08**	14	58.41	55.61
Vertebral Column	**73.55**	9	72.90	72.90
ECG5000	92.80	25	**93.00**	92.60
Limphography	**71.62**	8	70.27	70.27

Table 2: Numerical results in terms of purity

References

[1] A. Astorino, A. Fuduli, and M. Gaudioso. DC models for spherical separation. *Journal of Global Optimization*, 48(4):657–669, 2010.

[2] A. Astorino, A. Fuduli, and M. Gaudioso. Margin maximization in spherical separation. *Computational Optimization and Applications*, 53(2):301–322, 2012.

[3] A. Astorino and M. Gaudioso. A fixed-center spherical separation algorithm with kernel transformations for classification problems. *Computational Management Science*, 6(3):357–372, 2009.

[4] A. Bagnall, A. Bostrom, , J. Large, J. Lines, E. Keogh, and M. Middlehurst. UEA & UCR time series classification repository. In *www.ics.uci.edu/∼mlearn/MLRepository.html*, 2018.

[5] A. Ben-Dor, R. Shamir, and Z. Yakhini. Clustering gene expression patterns. *Journal of Computational Biology*, 6(3-4):281–297, 1999.

[6] M. Chavent, Y. Ding, L. Fu, H. Stolowy, and H. Wang. Disclosure and determinants studies: An extension using the divisive clustering method (div). *European Accounting Review*, 15(2):181–218, 2006.

[7] M. Cheng, T. Ma, L. Ma, J. Yuan, and Q. Yan. Adaptive grid-based forest-like clustering algorithm. *Neurocomputing*, 481:168–181, 2022.

[8] W.H.E. Day and H. Edelsbrunner. Efficient algorithms for agglomerative hierarchical clustering methods. *Journal of Classification*, 1(1):7–24, 1984.

[9] Inderjit S. Dhillon, Yuqiang Guan, and J. Fan. Efficient clustering of very large document collections. In C. Kamath, V. Kumar, R. Grossman, and

R Namburu, editors, *Data Mining for Scientific and Engineering Applications*, page 357–381. Kluwer Academic Publishers, 2001.

[10] C. Fraley and A.E. Raftery. How many clusters? which clustering method? answers via model-based cluster analysis. *Computer Journal*, 41(8):586–588, 1998.

[11] C. Fraley and A.E. Raftery. Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association*, 97(458):611–631, 2002.

[12] J. Hou, W. Liu, X. E, and H. Cui. Towards parameter-independent data clustering and image segmentation. *Pattern Recognition*, 60:25–36, 2016.

[13] A. K. Jain. Data clustering: 50 years beyond k-means. *Pattern Recognition Letters*, 31(8):651–666, 2010.

[14] A.K. Jain, M.N. Murty, and P.J. Flynn. Data clustering: A review. *ACM Computing Surveys*, 31(3):264–323, 1999.

[15] L. Kaufman and P.J. Rousseeuw. Finding groups in data: An introduction to cluster analysis. *Biometrics*, (2), 1990.

[16] R.J. Kuo, L.M. Ho, and C.M. Hu. Cluster analysis in industrial market segmentation through artificial neural network. *Computers and Industrial Engineering*, 42(2-4):391–399, 2002.

[17] H. A. Le Thi, L. H. Minh, T. Pham Dinh, and V. H. Ngai. Binary classification via spherical separator by DC programming and DCA. *Journal of Global Optimization*, 56:1393–1407, 2013.

[18] H. Li, J. Liu, K. Wu, Z. Yang, R.W. Liu, and N. Xiong. Spatio-temporal vessel trajectory clustering based on data mapping and density. *IEEE Access*, 6:58939–58954, 2018.

[19] R. Liu, W. Huang, Z. Fei, K. Wang, and J. Liang. Constraint-based clustering by fast search and find of density peaks. *Neurocomputing*, 330:223–237, 2019.

[20] X. Min, D. Pfoser, A. Züfle, and Y. Sheng. A hierarchical spatial network index for arbitrarily distributed spatial objects. *ISPRS International Journal of Geo-Information*, 10(12), 2021.

[21] B. Mirkin. Mathematical classification and clustering. *Mathematical Classification and Clustering*, 1996.

[22] P. M. Murphy and D. W. Aha. UCI repository of machine learning databases. In *http://www.timeseriesclassification.com*, 1992.

[23] G. Nagy. State of the art in pattern recognition. *Proceedings of the IEEE*, 56(5):836–857, 1968.

[24] J. Sander, M. Ester, H.-P. Kriegel, and X. Xu. Density-based clustering in spatial databases: The algorithm gdbscan and its applications. *Data Mining and Knowledge Discovery*, 2(2):169–194, 1998.

[25] E. Schikuta and M. Erhart. The bang–clustering system: Grid–based data analysis. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 1280:513–524, 1997.

[26] H. Spath. *Cluster Analysis Algorithms for Data Reduction and Classification of Objects*, 1980.

[27] M. Teboulle. A unified continuous optimization framework for center-based clustering methods. *Journal of Machine Learning Research*, 8:65–102, 2007.

[28] S.D. Tomarchio, L. Bagnato, and A. Punzo. Model-based clustering via new parsimonious mixtures of heavy-tailed distributions. *AStA Advances in Statistical Analysis*, 2022.

[29] P. Tseng. Convergence of a block coordinate descent method for nondifferentiable minimization. *Journal of Optimization Theory and Applications*, 109(3):475–494, 2001.

[30] A.K.H. Tung, J. Han, L.V.S. Lakshmanan, and R.T. Ng. Constraint-based clustering in large databases. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 1973:405–419, 2001.

[31] R. Veloso, F. Portela, M.F. Santos, Á. Silva, F. Rua, A. Abelha, and J. Machado. A clustering approach for predicting readmissions in intensive medicine. *Procedia Technol*, 16:1307–1316, 2014.

[32] W. Wang, J. Yang, and R. Muntz. Sting: A statistical information grid approach to spatial data mining. pages 186–195, 1997.