α-Clusterable Sets

Gerasimos S. Antzoulatos and Michael N. Vrahatis

Computational Intelligence Laboratory (CILAB)
Department of Mathematics
University of Patras Artificial Intelligence Research Center (UPAIRC)
University of Patras, GR-26110 Patras, Greece
antzoulatos@upatras.gr, vrahatis@math.upatras.gr

Abstract. In spite of the increasing interest into clustering research within the last decades, a unified clustering theory that is independent of a particular algorithm, or underlying the data structure and even the objective function has not be formulated so far. In the paper at hand, we take the first steps towards a theoretical foundation of clustering, by proposing a new notion of "*clusterability*" of data sets based on the density of the data within a specific region. Specifically, we give a formal definition of what we call "*α-clusterable*" set and we utilize this notion to prove that the principles proposed in Kleinberg's impossibility theorem for clustering [25], are consistent. We further propose an unsupervised clustering algorithm which is based on the notion of α-clusterable set. The proposed algorithm exploits the ability of the well known and widely used particle swarm optimization [31] to maximize the recently proposed window density function [38]. The obtained clustering quality is compared favorably to the corresponding clustering quality of various other well-known clustering algorithms.

1 Introduction

Cluster analysis is an important human process associated with the human ability to distinguish between different classes of objects. Furthermore, clustering is a fundamental aspect of data mining and knowledge discovery. It is the process of detecting homogenous groups of objects without any priori knowledge about the clusters. A cluster is a group of objects or data that are similar to one another within the particular cluster and are dissimilar to the objects that belong to another cluster [9, 19, 20].

The last decades, there exists an increasing scientific interest in clustering and numerous applications, in different scientific fields have appeared, including statistics [7], bioinformatics [37], text mining [43], marketing and finance [10, 26, 33], image segmentation and computer vision [21] as well as pattern recognition [39], among others. Many clustering algorithms have been proposed in the literature, which can be categorised into two major categories, hierarchical and partitioning [9, 22].

Partitioning algorithms consider the clustering as an optimization problem. There are two directions. The first one discovers clusters through optimizing a goodness criterion based on the distance of the dataset's points. Such algorithms are *k*-means [27], ISODATA [8] and fuzzy *c*-means [11]. The second one utilizes the notion of density and considers clusters as high-density regions. The most characteristic algorithms of this approach are DBSCAN [18], CLARANS [28] and *k*-windows [41].

D. Gunopulos et al. (Eds.): ECML PKDD 2011, Part I, LNAI 6911, pp. 108–123, 2011.

Recent approaches for clustering apply population based globalized search algorithms exploiting the capacity (cognitive and social behaviour) of the swarms and the ability of an organism to survive and adjust in a dynamically changing and competitive environment [1, 6, 12, 13, 14, 29, 32]. Evolutionary Computation (EC) refers to the computer-based methods that simulate the evolution process. Genetic algorithms (GA), Differential Evolution (DE) and Particle Swarm Optimization (PSO) are the main algorithms of EC [16]. The principal issues of these methods consist of the representation of the solution of the problem and the choice of the objective function.

Despite of the considerably progress and innovations that the last decades have been occurred, there is a gap between practical and theoretical clustering foundation [2, 3, 25, 30]. The problem is getting worse due to the lack of a unified definition of what a cluster is, which will be independent of the measure of similarity/ dissimilarity or the algorithm of clustering. Going a step further, it is difficult to answer questions such as how many clusters exist in a dataset, without having any priori knowledge for the underlying structure of the data, or whether a k-clustering of a dataset is meaningful.

All these weaknesses led to the development of the study of theoretical background of clustering aiming to develop a general theory. Thus, Puzicha *et al.* [35], considered the proximity-based data clustering as a combinatorial optimisation problem and moreover their proposed theory aimed to face two fundamental problems: (i) the specification of suitable objective functions, and (ii) the derivation of efficient optimisation algorithms.

In 2002 Kleinberg [25] developed an axiomatic framework for clustering and showed that there is no clustering function that could satisfy simultaneously three simple properties, the *scale–invariance*, the *richness* and the *consistency* condition. Kleinberg's goal was to develop a theory of clustering that would not be dependent on any particular algorithm, cost function or data model. To accomplish that, a set of axioms was set up, aiming to define what the clustering function is. Kleinberg's result was that there is no clustering function satisfying all three requirements.

After some years, Ackerman and Ben-David [2] disagreed with Kleinberg's impossibility theorem claiming that Kleinberg's result, was to a large extent, the outcome of a specific formalism rather than being an inherent feature of clustering. They focused on the clustering-quality framework rather than to attempt to define what a clustering function is. They developed a formalism and consistent axioms of the quality of a given data clustering. This lead to a further investigation of interesting measures of clusterability of data sets [3]. Clusterability is a measure of clustered structure in a data set. Although, in the literature, several notions of clusterability [17, 35] have been proposed and in addition they share the same intuitive concept, however these notions are pairwise incompatible, as Ackerman *et al.*, have proved in [3]. Furthermore, they concluded that the finding a *close-to-optimal* clustering for well clusterable data set is computationally easy task comparing with the common clustering task which is NP-hard [3].

Contribution: All the aforementioned theoretical approaches refer to the distance-based clustering and implicitly mention that the dissimilarity measure is a *distance function*. Thus, the concept of clusterability is inherent in the concept of a distance. In the paper at hand, we try to investigate if the notion of clusterability could be extended in density-based notion of clusters. To attain this goal, we introduce the notion of *α-clusterable set*, that is based on the *window density function* [38]. We aim to capture the dense regions of points in the data set, given an arbitrary parameter α, which presents the size of a D-range, where D is the dimensionality of the data set. Intuitively, a cluster can be considered as a dense area of data points, which is separated from other clusters with sparse areas of data or areas without any data point. Under this consideration, a cluster can be seen as an α-clusterable set or as an union of all intersecting α-clusterable sets. Then, a clustering, called *α-clustering*, will be comprised of the set of all the clusters. In this theoretical framework, we are able to show that the properties of Kleinberg's impossibility theorem are satisfied. Particularly, we prove that in the class of window density functions there exist clustering functions satisfying the properties of scale-invariance, richness and consistency. Furthermore, a clustering algorithm can be found utilising the theoretical framework and having as the goal to detect the α-clusterable sets.

Thus, we propose an unsupervised clustering algorithm that exploits the benefits of a population-based algorithm, known as particle swarm optimisation, in order to detect the centres of the dense regions of data points. These regions are actually what we call *α-clusterable sets*. When all the *α-clusterable sets* have been identified, the merging procedure is executed in order to merge the regions that have an overlap each other. After this process, the final clusters will have been formed and the α-clustering will has been detected.

The rest of the paper is organized as follows. In the next section we briefly present the background work that our theoretical framework, which is analysed in Section 3, is based on. In more detail, we present and analyse the proposed definitions of α-clusterable set and α-clustering, and futhermore we show that, using these concepts the conditions of Kleinberg's impossibility theorem for clustering are hold and are consistent. Section 4 gives a detailed analysis of the experimental framework and the proposed algorithm. In Section 5 the experimental results are demonstrated. Finally, the paper ends in Section 6 with conclusions.

2 Background Material

For completeness purposes, let us briefly describe the Kleingberg's axioms [25] as well as the window density functions [38].

2.1 Kleinberg's Axioms

As we have already mentioned above, Kleingberg, in [25], proposed three axioms for clustering functions and claimed that this set of axioms is inconsistent, meaning that lack of clustering function that satisfies all the three axioms. Let $X = \{x_1, x_2, \ldots, x_N\}$ be a data set with cardinality N and let $d: X \times X \rightarrow \mathbb{R}$ be a *distance function* over X, that means $\forall x_i,\ x_j \in X,\ d(x_i, x_j) > 0$ if and only if $x_i \neq x_j$ and $d(x_i, x_j) = d(x_j, x_i)$ otherwise. It is worth observing that the triangle inequality is not necessary to be fulfilled, i.e. distance function should not be considered as a metric function. Furthermore, a *clustering function* is a function f which, given a distance function d, separates the data set X into a set of Γ clusters.

The first axiom, ***scale-invariance***, is concern with the requirement that the clustering function have to be invariant to changes in the units of a distance measure. Formally, for any distance function d and any $\lambda > 0$, a clustering function f is *scale-invariant* if $f(d) = f(\lambda d)$.

The second property, called ***richness***, deals with the outcome of the clustering function, and it requires that every possible partition of the data set can be obtained. Typically, a function f is ***rich*** if for each partition Γ of X, there exist a distance function d over X such that $f(d) = \Gamma$.

The ***consistency*** property requires that if the distances between the points laid in the same cluster are decreased and the distances between points laid in a different clusters are increased, then the clustering result does not change. Kleinberg gave the following definition:

Definition 1. *Let Γ be a partition of X and d, d' are two distance functions on X. Then, a distance function d' is a **Γ-transformation** of d if (a) $\forall\, x_i,\ x_j \ \in X$ belonging to the same cluster of Γ, it holds $d'(x_i,x_j) \leqslant d(x_i,x_j)$ and (b) $\forall\, x_i,\ x_j \ \in X$ belonging to different clusters of Γ, it holds $d'(x_i,x_j) \geqslant d(x_i,x_j)$. Furthermore, a function f is **consistent** if $f(d) = f(d')$, whenever a distance function d' is a Γ-transformation of d.*

Using the above axioms, Kleinberg stated the impossibility theorem [25]:

Theorem 1 (Impossibility Theorem). *For each $N \geqslant 2$, there is no clustering function f that satisfies scale-invariance, richness and consistency.*

2.2 Window Density Function

In [38] the authors proposed a window density function as an objective function, so as to discover the optimum clustering. Assume that the data set comprise a set $X = \{x_1, x_2, \ldots, x_N\}$, where x_j is a data point in the D–dimensional Euclidean space $\mathbb{R}^D$. Then we give the following definition:

Definition 2 (Window Density Function). *Let a D-range of size $\alpha \in \mathbb{R}$ and center $z \in \mathbb{R}^D$ be the orthogonal range $[z_1 - \alpha, z_1 + \alpha] \times \cdots \times [z_D - \alpha, z_D + \alpha]$. Assume further, that the set $S_{\alpha,z}$, with respect to the set X, is defined as:*

$$S_{\alpha,z} = \{y \in X :\ z_i - \alpha \leqslant y_i \leqslant z_i + \alpha,\ \forall\, i = 1,2,\ldots,D\} \,.$$

Then the Window Density Function (WDF) for the set X, with respect to a given size $\alpha \in \mathbb{R}$ is defined as:

$$\mathrm{WDF}_\alpha(z) = |S_{\alpha,z}| \,, \tag{1}$$

where $|\cdot|$ indicates the cardinality of the set $S_{\alpha,z}$.

WDF is a non-negative function that expresses the density of the region (orthogonal range) around the point. The points that are included in this region can be effectively estimated using Computational Geometry methods [5, 34]. For a given α, the value of WDF increases continuously as the density of the region within the window increases. Furthermore, for low values of α, WDF has many local maxima. While the value of α

increases, WDF reveals the number of local maxima that corresponds to the number of clusters. However for higher values of the parameter, WDF becomes smoother and the clusters are not distinguished.

Thus, it is obvious, that the determination of the dense region depends on the size of the window. Actually, the parameter α captures our inherent view for the size of the dense regions that there exist in the data set. To illustrate the effect of parameter α, we employ the following dataset *Dset1* which contains 1600 data points in the 2-dimensional Euclidean space (Fig. 1(a)).

In the following figures the behaviour of WDF function is exhibited over distinct values of the α parameter. As we can conclude, when the value of parameter α is increasing more dense and smooth regions of data points is detected. When $\alpha = 0.05$ or $\alpha = 0.075$ there are many maxima inside the real clusters of data points, Fig. 1(b), Fig. 1(c) respectively. As α increases there is a clear improvement on the formation of groups, namely the dense regions are more distinct and separate, so between the values $\alpha = 0.1$ and $\alpha = 0.25$ we can detect the four real clusters of data points, Fig 1(d), Fig. 1(e) respectively. If the parameter α continue to grow, then the four maximum of the WDF function corresponding to the four clusters of data points, which were detected previously, merge into one single maximum leading to the formation of one cluster, Fig 1(f).

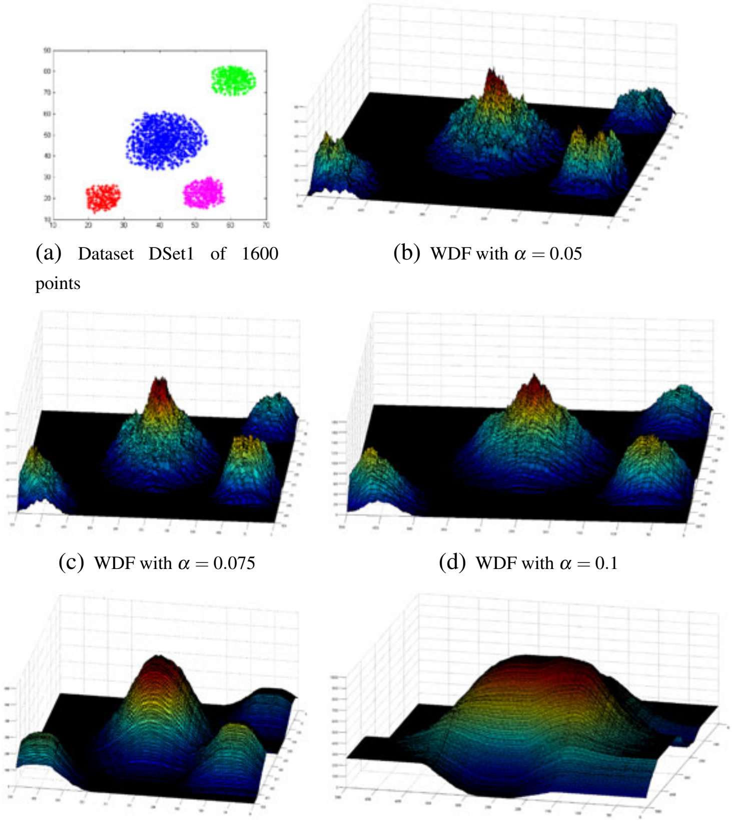
3 Proposed Theoretical Framework

In this section, we give the definitions needed to support the proposed theoretical framework for clustering. Based on the observation that a good clustering is one that separates the points of all data in high-density areas, which are separated by areas of sparse points or areas with no points, we define the notion of an α–clusterable set as well as the notion of α–clustering. To do this, we exploit the benefits of window density function and its ability to find local dense regions of data points without investigate the whole dataset.

Definition 3 **(α–Clusterable Set).** *Let X be the data set that is comprised of the set of points $\{x_1, x_2, \dots, x_N\}$. A set of data points $x_m \in X$ is defined as an α–**clusterable set** if there exist a positive real value $\alpha \in \mathbb{R}$, a hyper–rectangle $\mathscr{H}_\alpha$ of size α and a point $z \in \mathscr{H}_\alpha$ in which the window density function centered at z is unimodal. Formally,*

$$C_{\alpha,z} = \Big\{ x_m \mid x_m \in X \ \wedge \ \exists\, z \in \mathscr{H}_\alpha \ : \ \mathrm{WDF}_\alpha(z) \geqslant \mathrm{WDF}_\alpha(y), \ \forall\, y \in \mathscr{H}_\alpha \Big\}. \qquad (2)$$

Remark 1. It is worth to mention that although the points y and z are laid in the hyper–rectangle $\mathscr{H}_\alpha$, however it is not necessary to be points of the data set. Also, the hyper–rectangle $\mathscr{H}_\alpha$ is a bounding box of the data set X and a set $C_{\alpha,z}$ is a subset of X. In addition, the α–clusterable set is a highly dense region due to the fact that the value of WDF function is maximised. Furthermore, the point z could be considered as the centre of the α–clusterable set. Thus, given an α and a sequence of points $z_i \in \mathscr{H}_\alpha$, a set that comprises of a number of α–clusterable sets could be considered as a *close to optimal* clustering of X.



(a) Dataset DSet1 of 1600 points

(b) WDF with $\alpha = 0.05$

(c) WDF with $\alpha = 0.075$

(d) WDF with $\alpha = 0.1$

(e) WDF with $\alpha = 0.25$

(f) WDF with $\alpha = 0.5$

Fig. 1. WDF with different values of parameter α

Definition 4 **(α–Clustering).** *Given a real value α, an α–**clustering** of a data set X is a partition of X, that is a set of k disjoint α–clusterable sets of X such that their union is X. Formally, an α–clustering is a set:*

$$\mathscr{C} = \left\{C_{\alpha,z_1}, C_{\alpha,z_2}, \ldots, C_{\alpha,z_k}\right\},$$

where $z_i \in \mathscr{H}_\alpha \subset \mathbb{R}^D$, $i = 1, 2, \ldots, k$ are the centres of the dense regions C_{α,z_i}.

We explain the above notions by given an example. Let X be the dataset of 1000 random data points that drawn from the normal (Gaussian) distribution (Figure 2). The four clusters have the same cardinality thus each one of them contains 250 points. As we can notice, there exist a proper value for the parameter α, $\alpha = 0.2$, so as the hyper–rectangles $\mathscr{H}_\alpha$ captures the whole clusters of points. These hyper–rectangles can be considered as the α–clusterable sets. Also, it is worth to mention that there is only one point z inside the α–clusterable set, such that the window density function is unimodal.

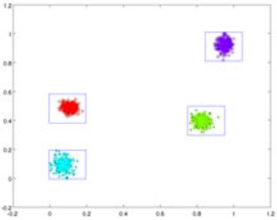


Fig. 2. Dataset of 1000 points. Parameter value is $\alpha = 0.2$

Furthermore, we define an *α–clustering function* for a data set X, that takes a window density function, with respect to a given size α, on X and returns a partition $\mathscr{C}$ of α–clusterable sets of X.

Definition 5 (α–Clustering Function). *A function $f_\alpha(\mathrm{WDF}_\alpha, X)$ is an α–**clustering function** if for a given window density function, with respect to a real value parameter α, returns a clustering $\mathscr{C}$ of X, such as each cluster of $\mathscr{C}$ is an α–clusterable set of X.*

Next, we prove that the clustering function f_α fulfills the properties of scale-invariance, consistency and richness. Intuitively, the scale-invariance property describes that in any uniform change in the scale of the domain space of the data, the high-density areas will be maintained and furthermore they will be separated by sparse regions of points. Richness means that there exist a parameter α and points z, such that an α-clustering function f can be constructed, with the property of partitioning the dataset X into α-clusterable sets. Finally, the consistency means that if we shrink the dense areas, α-clusterable sets, and simultaneously expand the sparse areas between the dense areas, then we can get the same clustering solution.

Lemma 1 (Scale-Invariance). *Every α–clustering function is scale-invariant.*

Proof. According to the definition of scale-invariance, every clustering function has this property if for every distance measure dist and any $\lambda > 0$ it holds that $f(\mathrm{dist}) = f(\lambda \mathrm{dist})$. Thus, in our case an α–clustering function, f_α, is scale-invariant since it holds that:

$$f_\alpha(\mathrm{WDF}_\alpha(z), X) = f_{\lambda\alpha}(\mathrm{WDF}_{\lambda\alpha}(\lambda z), X),$$

for every positive number λ. This is so because if a data set X is scaled by a factor $\lambda > 0$, then the window density function of each point will be remain the same. Indeed, if a uniform scale is applied to the dataset, then we can find a scale factor λ, such that a scaled window, with size $\lambda\alpha$, contains the same amount of points as the window of size α. More specifically, for each data point $y \in X$ that belongs to a window which has center the point z and size α, it holds that:

$$z - \alpha \leqslant y \leqslant z + \alpha \quad \Leftrightarrow \quad \lambda z - \lambda\alpha \leqslant \lambda y \leqslant \lambda z + \lambda\alpha\,.$$

So, if the point $y \in X$ belongs to the window of size α and center z, then the point $y' = \lambda y$, $y' \in X'$ will belong to the scaled window, which has size $\lambda\alpha$ and center the point $z' = \lambda z$. Thus the lemma is proved. □

Lemma 2 (Richness). *Every α–clustering function satisfies the richness property.*

Proof. It is obvious that for each non-trivial α–clustering $\mathscr{C}$ of X, there exist a window density function for the set X, with respect to a size α, such that:

$$f(\mathrm{WDF}_\alpha(z), X) = \mathscr{C}.$$

In other words, given a data set of points X we can find a WDF and a size α, such that each window with size α and center the point z will be an α–clusterable set. Thus the lemma is proved. □

Lemma 3 (Consistency). *Every α–clustering function is consistent.*

Proof. Suppose that f_α is an α–clustering function. By definition, there exist α–clusterable sets of X that constitute a set

$$\mathscr{C} = \{C_{\alpha,z_1}, C_{\alpha,z_2}, \dots, C_{\alpha,z_k}\},$$

where each $z_i \in \mathscr{H}_\alpha$, $i = 1, 2, \dots, k$ is the centre of each α–clusterable set, C_{α,z_i}. According to the definition of α–clusterable set, the window density function is unimodal for each set C_{α,z_i}. Thus, for each $y \in \mathscr{H}_\alpha$ it holds that $\mathrm{WDF}_\alpha(z) \geqslant \mathrm{WDF}_\alpha(y)$.

Furthermore, if we reduce the value of window density function by decreasing the value of parameter α to a smaller value α', then for the set C_{α',z_i}, where $\alpha' < \alpha$, the WDF is also unimodal centered at the point z_i. Assume that there exists another point $z_i' \neq z_i$ such that $\mathrm{WDF}_{\alpha'}(z_i') \geqslant \mathrm{WDF}_\alpha(z_i)$, then the WDF function would be a multimodal function for the set C_{α,z_i}, implies that the set C_{α,z_i} is not an α–clusterable set, which is in contrary to our assumption. So, C_{α,z_i} is an α–clusterable set for each value $\alpha' < \alpha$, that means

$$f_{\alpha'}(\mathrm{WDF}_{\alpha'}(z_i'), X) = \mathscr{C},$$

which implies that f_α is consistent. Thus the lemma is proved. □

In the contrary of the general framework of Kleinberg's impossibility theorem, we obtain the following theorem:

Theorem 2. *For each $N \geqslant 2$ there is an α–clustering function that satisfies the properties of scale-invariance, richness and consistency.*

Proof. The proof follows using Lemmata 1, 2 and 3. □

4 Experimental Framework

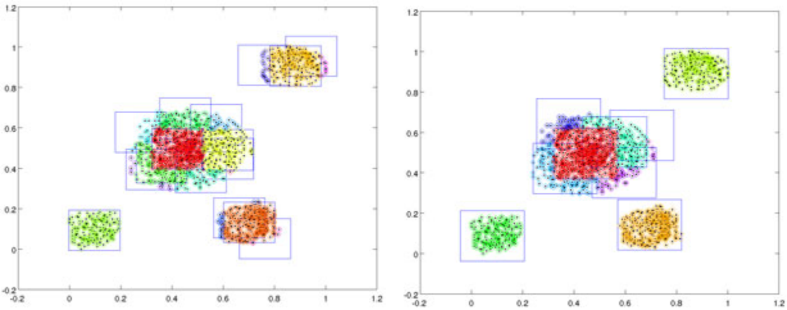
In this section we propose an unsupervised algorithm, in the sense that it doesn't require a predefined number of clusters in order to detect the α–clusterable sets laiding in the dataset X. Define the correct number of clusters is a critical open issue in cluster analysis, Dubes refer to it as "*the fundamental problem of cluster analysis*" [15], because the number of clusters is often tough to determine or, even worse, impossible to define.

Thus, the main goal of the algorithm is to identify the dense regions of points, in which the window density function is unimodal. These regions constitute the α–clusterable sets that enclose the real clusters of the dataset. The algorithm runs iteratively identifying the centre of the α–clusterable set, removing the data points that lie within it. The above process continues until no data points left in the dataset. In order to detect the centre of the dense regions we utilised a well-known population-based optimisation algorithm, called Particle Swarm Optimisation (PSO) [23]. PSO is inspired by swarm behaviour, such as flocking birds collaboately searching for food. In the last decades there has been a rapid increase of the scientific interest around Swarm Intelligence and particularly around Particle Swarm Optimization and numerous approaches have been proposed in many application fields [16, 24, 31]. Recently, Swarm Intelligence and especially Particle Swarm Optimisation have been utilised in Data Mining and Knowledge Discovery, producing promising results [1, 40].

In [6] an algorithm, called IUC, has been proposed, which utilises as objective function the window density function and Differential Evolution algorithm in order to evolve the clustering solution of the data set reaching the best position of the data set. Also, they use an enlargment procedure in order to detect all the points that laying in the same cluster. In the paper at hand, we exploit the benifits of the Particle Swarm Optimisation algorithm to search the space of potential solutions efficiently, so as to find the global optimum of a window density function. Each particle presents the centre of a dense region of the dataset, so the particles are flying through the seach space forming folks around peaks of window density function. Thus, the algorithm detects the centre of the α–clusterable set one each time.

It is worth to say that the choice of the value of the parameter α seems to play an important role of the identifcation of the real number of clusters and depends on several factors. For instance, if the value of the parameter α is too small so the hyper-rectangle is not able to capture the whole cluster, or if the data points shape dense regions with various cardinality, then again the hyper-rectangles with constant size α are difficult to capture the whole clusters of the datasets. The following figures describe the above cases more clearly. We conclude that the small choice of parameter α leads to the detection of small dense regions that are the α–clusterable sets. However, as we can be noticed, even for the detection of small clusters of data points, it needs more than one α–clusterable set (Fig. 3(a)). On the other hand, increasing α causes the detection of small clusters of the data sets by using only one α–clusterable set. However, the detection of the big cluster needs more α–clusterable sets, the union of them describes the whole cluster. It has to mentioned here, that the union of overlapping α–clusterable sets is still an α-clusterable set, hence we can find a point z which will be the centre of the set and its window density function value is maximum, in a hyper-rectangle size $\alpha' > \alpha$, means that the $\mathrm{WDF}_{\alpha'}(z)$ is unimodal.

In order to avoid the above situations, we propose and implement a merging procedure that merges the overlapping α–clusterable sets, so that the outcome of the algorithm represents the real number of clusters in the data set. Specifically, two dense regions (α–clusterable sets) are going to merge if and only if the overlap between them contains at least one data point.



(a) Effect of parameter value $\alpha = 0.2$ (b) Effect of parameter value $\alpha = 0.25$

Fig. 3. Dataset WDF with different values of parameter α

Subsequently, we summarise the above analysis and we propose the new clustering algorithm. It is worth to refer that the detection α–clusterable sets, which are highly density regions of the datasets, through the window density function is a maximization problem, however Particle Swarm Optimisation is a minimization algorithm, hence $-\mathrm{WDF}_\alpha(z)$ is utilised as the fitness function.

Algorithm 1. PSO for the Unsupervised Detection α–Clusterable Sets

```
repeat
    Create a data structure that holds all unclustered points
    Perform the PSO algorithm returning the center $z$ of an $\alpha$–clusterable set
    Mark the points that lie in the window $w$ as clustered
    Remove the clustered points from the dataset
until no left unclustered points
Mark the points that lie in overlapping windows as members of the same cluster and merge
these windows to form the clusters.
```

It needs be stressed that the proposed algorithm clusters a dataset in an unsupervised manner, since it detects the clusters without a priori knowledge of their number. It is based solely on the density of a region. Still for the execution of the algorithm a user must determine the parameter α, this user-defined parameter is easily regulated, in contrast with the number of clusters that is an invariant feature characterising the underlying structure of the dataset and furthermore it is difficult to define. Also, Particle Swarm Optimization's search space dimension is fixed to the dimensionality of the dataset, in contrast to the majority of other approaches in the literature that increase the dimensionality of the optimisation problem by a factor of the maximum number of estimated clusters.

5 Experimental Results

The objective of the conducted experiments was three-fold. First, we want to investigate the behaviour of the algorithm regarding the resizing of the window. Second, we compare the proposed algorithm with well-known partitioning clustering algorithms, k-means, DBSCAN, k-windows, DEUC and IUC. Third, we want to examine the scalability of the proposed algorithm.

In order to evaluate the performance of the clustering algorithms the Entropy and Purity measures are utilised. The Entropy function [43] represents the dissimilarity of the points lying in a cluster. Higher homogeneity means that entropy's values converge to zero. However, for the usage of the entropy function, the knowledge of the real classification/categorization of the points is required. Let, $\mathscr{C} = \{C_1, C_2, \ldots, C_k\}$ be a clustering provided by a clustering algorithm and $\mathscr{L} = \{L_1, L_2, \ldots, L_m\}$ be the target classification of the patterns, then the entropy of each cluster C_i is defined as $H_i = -\sum_{j=1}^{m} P(x \in L_j | x \in C_i) \log P(x \in L_j | x \in C_i)$. For a given set of n patterns, the entropy of the entire clustering is the weighted average of the entropy of each cluster. The Purity is defined as $r = \frac{1}{n} \sum_{i=1}^{k} \alpha_i$, where k denotes the number of clusters found in the dataset and α_i represents the number of patterns of the class to which the majority of points in cluster i belongs to it. The larger the values of purity, the better the clustering solution is [42].

5.1 Investigate the Effect of the Parameter α

The aim of these experiments is to investigate the effect of the parameter α to the performance of the proposed clustering algorithm. To do this, we utilised three 2-dimenstional artificial datasets (Fig. 4). The first one, $Dset_1$ has 1600 points that form four spherical clusters each one with different size. The second one, $Dset_2$ has 2761 points grouping into four arbitrary shape clusters, three of them are convex and one is non-convex. The final dataset, $Dset_3$ contain 5000 points contains 2 randmly created clusters whereby one cluster is located at the centre area of a quadratic grid and the other surrounds it, as described in [1].

The following three plots Fig. 5 present the entropy and the purity of the clustering plotted against the increase of the window size. As we can conlude, the clustering quality is getting worse while the parameter α takes higher values. This is rational due to the fact that higher values of parameter α, lead to the creation of sets that contains data from different groups.

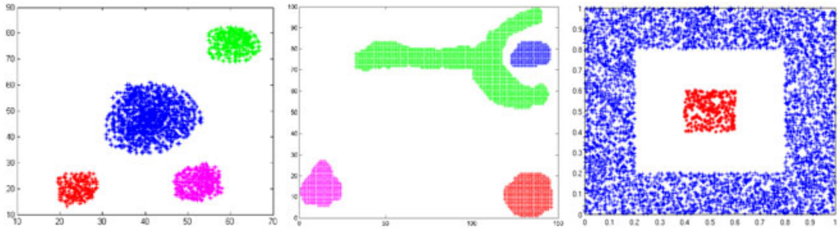
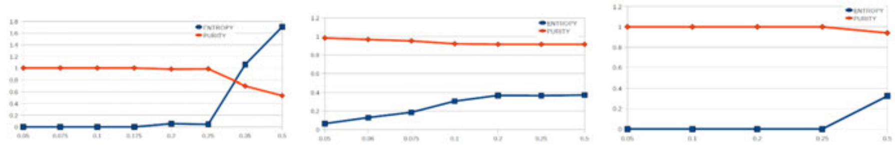


Fig. 4. Datasets $Dset_1$, $Dset_2$ and $Dset_3$



(a) Entropy and Purity vs Window Size α for the $Dset_1$

(b) Entropy and Purity vs Window Size α for the $Dset_2$

(c) Entropy and Purity vs Window Size α for the $Dset_3$

Fig. 5. Entropy and Purity vs Window Size α

5.2 Comparing Proposed Algorithm against Well-Known Clustering Algorithms

In this series of experiments, we investigate the performance of the proposed algorithm versus the performance of other well known clustering algorithms, such as k-means [27], DBSCAN [18], k-windows [41] and two evolutionary clustering algorithms called DEUC [38] and IUC [6]. All algorithms are implemented using the C++ programming language on the Linux operating system. For each dataset, the algorithms are executed 100 times, except DBSCAN that due to its deterministic nature was executed once. For the k-means algorithm the parameter k is set equal to the real number of clusters in each dataset. For the other algorithms, their parameters were determined heuristically. Finally, for the algorithms DEUC and IUC all the mutation operators are utilized in order to investigate their effects on clustering quality.

In this series of experiments, apart from the two datasets $Dset_1$ and $Dset_2$, we utilise two more datasets of 15000 points each one and which are randomly generated from multivariate normal distributions. The first of them, denoted as $Dset_4$, was created as described in [32], with unary covariance matrix and different mean vectors, forming six clusters with various cardinality. To form the latter dataset ($Dset_5$) we utilised random parameters based on [36]. It contains eight clusters. All the datasets are normalized in the $[0,1]^D$ range.

The experimental results (Table 1) for the datasets show that the proposed algorithm, called PSOα-Cl, attains to find a good clustering solution in the majority of the experiments, as the average entropy tends to be zero and the average purity tends to 100%.

5.3 Investigate the Scalability of the Algorithm

In order to examine the scalability of the proposed algorithm, we created artificial datasets which are randomly generated from multivariate normal distribution with different mean value vectors and convariance matrices. The data of each one of these datasets are clustered into eight groups with various cardinality. Also, the dimensionality of the datasets vary between 3 to 10 dimensions. All the datasets contain 15000 points and are normalized in the $[0,1]^D$ range. We tested the performance of the proposed algorithm in different values of parameter α and in each case we calculate the entropy and the purity measures. Observing the results (Table 2), we can conclude that the proposed algorithm exhibits good scalability properties since the entropy tends to zero and the purity tends to one when the dimensionality and the cardinality of the datasets increase. Moreover, it is worth to note that for the higher dimensional datasets

Table 1. The mean values and standard deviation of entropy and purity for each algorithm over the four datasets

	Dset$_1$		**Dset$_2$**	
	Entropy	*Purity*	*Entropy*	*Purity*
IUC DE1	8.55e-3(0.06)	99.7%(0.02)	4.54e-2(0.11)	98.9%(0.03)
IUC DE2	1.80e-2(0.1)	99.4%(0.03)	3.08e-2(0.09)	*99.2%*(0.03)
IUC DE3	*1.94e-4*(0.002)	*100%*(0.0)	7.16e-2(0.13)	98.2%(0.03)
IUC DE4	6.01e-3(0.06)	99.8%(0.01)	4.21e-2(0.10)	99.0%(0.02)
IUC DE5	2.46e-2(0.01)	99.2%(0.03)	6.95e-2(0.13)	98.3%(0.03)
DEUC DE1	1.70e-1(0.1)	91.0%(0.05)	3.39e-2(0.02)	90.5%(0.01)
DEUC DE2	1.36e-1(0.09)	92.3%(0.05)	3.22e-2(0.02)	90.3%(0.01)
DEUC DE3	1.66e-1(0.09)	90.4%(0.05)	2.90e-2(0.02)	90.8%(0.01)
DEUC DE4	1.45e-1(0.09)	91.1%(0.04)	3.16e-2(0.02)	90.4%(0.01)
DEUC DE5	1.39e-1(0.1)	92.9%(0.05)	2.88e-2(0.02)	90.6%(0.01)
k-means	1.10e-1(0.21)	96.7%(0.06)	3.45e-1(0.06)	90.5%(0.03)
k-windows	*0.00e-0*(0.0)	99.2%(0.02)	*2.20e-2*(0.08)	95.4%(0.01)
DBSCAN	*0.00e-0*(—)	*100%*(—)	3.74e-1(—)	*100.0%*(—)
PSO 0.05-Cl	**0.00e-0**(0.0)	**100%**(0.0)	6.44e-2(0.11)	98.2%(0.03)
PSO 0.075-Cl	**0.00e-0**(0.0)	**100%**(0.0)	1.86e-1(0.16)	95.1%(0.04)
PSO 0.1-Cl	**0.00e-0**(0.0)	92.048%(0.01)	3.07e-1(0.08)	92.0%(0.0)
PSO 0.2-Cl	5.54e-2(0.17)	98.2%(0.06)	3.68e-1(0.01)	91.4%(0.0)
PSO 0.25-Cl	4.3e-2(0.15)	98.6%(0.05)	3.66e-1(0.01)	91.4%(0.0)

	Dset$_4$		**Dset$_5$**	
	Entropy	*Purity*	*Entropy*	*Purity*
IUC DE1	2.52e-3(0.02)	94.7%(0.05)	2.7e-3(0.03)	99.7%(0.01)
IUC DE2	7.59e-3(0.04)	96.0%(0.04)	7.9e-3(0.04)	99.5%(0.02)
IUC DE3	1.02e-2(0.05)	95.5%(0.04)	8.0e-3(0.04)	99.6%(0.02)
IUC DE4	*0.00e+0*(0.0)	96.6%(0.01)	1.06e-3(0.05)	99.4%(0.02)
IUC DE5	5.04e-3(0.03)	97.0%(0.01)	2.12e-3(0.07)	99.0%(0.02)
DEUC DE1	6.86e-3(0.01)	90.7%(0.02)	2.63e-3(0.21)	87.4%(0.07)
DEUC DE2	6.04e-3(0.01)	91.0%(0.02)	2.90e-3(0.19)	86.4%(0.06)
DEUC DE3	6.16e-3(0.07)	91.2%(0.01)	2.94e-3(0.21)	86.4%(0.07)
DEUC DE4	7.17e-3(0.01)	89.9%(0.02)	3.09e-3(0.24)	86.0%(0.07)
DEUC DE5	6.38e-3(0.01)	90.1%(0.02)	2.79e-3(0.22)	86.8%(0.07)
k-means	2.69e-1(0.18)	89.9%(0.07)	3.99e-3(0.25)	86.8%(0.09)
k-windows	4.18e-5(0.0)	98.3%(0.003)	*0.00e-0*(0.0)	99.7%(0.006)
DBSCAN	8.54e-4(—)	*99.2%*(—)	*0.00e-0*(0.0)	*100%*(—)
PSO 0.05-Cl	**0.00e-0**(0.0)	**99.9%**(0.0)	**0.00e-0**(0.0)	99.0%(0.0)
PSO 0.075-Cl	1.03e-2(0.05)	99.5%(0.02)	**0.00e-0**(0.0)	**100.0%**(0.0)
PSO 0.1-Cl	7.95e-2(0.12)	96.9%(0.05)	**0.00e-0**(0.0)	**100.0%**(0.0)
PSO 0.2-Cl	4.62e-1(0.15)	81.9%(0.06)	1.02e-2(0.05)	99.5%(0.02)
PSO 0.25-Cl	1.76e-0(0.17)	45.5%(0.05)	1.30e-1(0.06)	94.7%(0.06)

Table 2. The mean values and standard deviation of entropy, purity and number of clusters

D	N	Measures	Size of the window α						
			0.2	**0.25**	**0.3**	**0.35**	**0.4**	**0.45**	**0.5**
3	15K	Entropy	0.01(0.06)	0.129(0.16)	0.485(0.07)	0.890(0.21)	2.340(0.47)	2.24(0.49)	2.68(0.0)
		Purity	0.995(0.02)	0.946(0.07)	0.834(0.02)	0.703(0.06)	0.337(0.09)	0.356(0.1)	0.266(0.0)
		#Clusters	8.2(0.56)	7.6(0.56)	5.6(0.48)	3.9(0.39)	1.3(0.5)	1.45(0.5)	1.0(0.0)
5	15K	Entropy	0.0(0.0)	0.0(0.0)	0.0(0.0)	0.264(0.0)	0.380(0.02)	0.655(0.0)	0.780(0.07)
		Purity	0.999(0.0)	1.0(0.0)	1.0(0.0)	0.906(0.0)	0.874(0.006)	0.793(0.0)	0.769(0.02)
		#Clusters	8.3(0.51)	8.2(0.46)	8.0(0.17)	7.0(0.0)	6.04(0.19)	5.0(0.0)	3.9(0.02)
10	15K	Entropy	0.0(0.0)	0.0(0.0)	0.0(0.0)	0.0(0.0)	0.043(0.04)	0.078(0.03)	0.114(0.09)
		Purity	0.996(0.007)	0.999(0.0)	0.999(0.0)	0.999(0.003)	0.990(0.01)	0.982(0.007)	0.971(0.03)
		#Clusters	8.1(0.40)	8.09(0.29)	8.05(0.22)	8.03(0.22)	7.57(0.56)	7.17(0.40)	6.97(0.30)

the performance of the clustering is increased when the size of the window becomes larg. However, if the size of the window exceeds a specific value, related to the dataset, the quality of the clustering deteriorates.

The scalability of the algorithm depends on the window density function and specifically depends on the complexity of determining the points that lie in a specific window. This is the well-known *orthogonal range search* problem that have been studied and many algorithms have been proposed in the literature to address it [4, 34]. A preprocessing phase is employed so as to construct the data structure that stores the data points. For high dimensional applications data structures like Multidimensional Binary Tree [34] is preferable, while for low dimensional applications with large number of points Alevizos's approach [4] is more suitable. In this work, we utilise the Multidimensional Binary Tree so the preprocessing time is $O(DN\log N)$, while the data structure demands $O(s+DN^{1-1/D})$ time to answer it to a query [38].

6 Conlusions

Although clustering is a foundamental process to discover knowledge from data, however it still difficult to give a clear, coherent and general definition of what is a cluster, or whether a dataset is clusterable or not. Furthermore, many researches focused on practical aspect of clustering and leave almost untouched the theoretical background. In this study, we have presented a theoretical framework of clustering and we introduced a new notion of clusterability, called "α–clusterable set", which is based on the notion of window densiy function. Particularly, an α–clusterable set is considered as the dense region of points of a dataset X and also inside of this area the window density function is unimodal. The set of these α–clusterable sets forms a clustering solution, denoted as α–clustering. Moreover, we prove, in contrary to the general framework of Kleinberg's impossibility theorem, that this α–clustering solution of a data set X satisfies the properties of scale-invariance, richness and consistency. Furthermore, to validate the theoretical framework, we propose an unsupervised algorithm based on the particle swarm optimisation. The experimental results are promising since its performance is better or similar to other well-known algorithms and in addition the proposed algorithm exhibits good scalability properties.

References

[1] Abraham, A., Grosan, C., Ramos, V.: Swarm Intelligence in Data Mining. Springer, Heidelberg (2006)

[2] Ackerman, M., Ben-David, S.: Measures of clustering quality: A working set of axioms for clustering. In: Advances in Neural Information Processing Systems (NIPS), pp. 121–128. MIT Press, Cambridge (2008)

[3] Ackerman, M., Ben-David, S.: Clusterability: A theoretical study. Journal of Machine Learning Research - Proceedings Track 5, 1–8 (2009)

[4] Alevizos, P.: An algorithm for orthogonal range search in $d \geq 3$ dimensions. In: Proceedings of the 14th European Workshop on Computational Geometry (1998)

[5] Alevizos, P., Boutsinas, B., Tasoulis, D.K., Vrahatis, M.N.: Improving the orthogonal range search k-windows algorithms. In: 14th IEEE International Conference on Tools and Artificial Intelligence, pp. 239–245 (2002)

[6] Antzoulatos, G.S., Ikonomakis, F., Vrahatis, M.N.: Efficient unsupervisd clustering through intelligent optimization. In: Proceedings of the IASTED International Conference Artificial Intelligence and Soft Computing (ASC 2009), pp. 21–28 (2009)

[7] Arabie, P., Hubert, L.: An overview of combinatorial data analysis. In: Clustering and Classification, pp. 5–64. World Scientific Publishing Co., Singapore (1996)

[8] Ball, G., Hall, D.: A clustering technique for summarizing multivariate data. Behavioral Sciences 12, 153–155 (1967)

[9] Berkhin, P.: Survey of data mining techniqes. Technical report, Accrue Software (2002)

[10] Berry, M.J.A., Linoff, G.: Data mining techniques for marketing, sales and customer support. John Willey & Sons Inc., USA (1996)

[11] Bezdek, J.C.: Pattern Recognition with Fuzzy Objective Function Algorithms. Kluwer Academic Publishers, Norwell (1981)

[12] Chen, C.Y., Ye, F.: Particle swarm optimization algorithm and its application to clustering analysis. In: IEEE International Conference on Networking, Sensing and Control, vol. 2, pp. 789–794 (2004)

[13] Cohen, S.C.M., Castro, L.N.: Data clustering with particle swarms. In: IEEE Congress on Evolutionary Computation, CEC 2006, pp. 1792–1798 (2006)

[14] Das, S., Abraham, A., Konar, A.: Automatic clustering using an improved differential evolution algorithm. IEEE Transactions on Systems, Man and Cybernetics 38, 218–237 (2008)

[15] Dubes, R.: Cluster Analysis and Related Issue. In: Handbook of Pattern Recognition and Computer Vision, pp. 3–32. World Scientific, Singapore (1993)

[16] Engelbrecht, A.P.: Computational Intelligence: An Introduction. John Wiley & Sons, Ltd., Chichester (2007)

[17] Epter, S., Krishnamoorthy, M., Zaki, M.: Clusterability detection and initial sees selection in large datasets. Technical Report 99-6, Rensselaer Polytechnic Institute, Computer Science Dept. (1999)

[18] Ester, M., Kriegel, H.P., Sander, J., Xu, X.: A density-based algorithm for discovering clusters in large spatial databases with noise. In: Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining, pp. 226–231 (1996)

[19] Han, J., Kamber, M.: Data Mining: Concepts and Techniques. Morgan Kaufmann Publishers, San Francisco (2006)

[20] Jain, A.K., Dubes, R.: Algorithms for Clustering Data. Prentice-Hall, Englewood Cliffs (1988)

[21] Jain, A.K., Flynn, P.J.: Image segmentation using clustering. In: Advances in Image Understanding: A Festschrift for Azriel Rosenfeld, pp. 65–83. Willey - IEEE Computer Society Press, Singapore (1996)

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

[23] Kennedy, J., Eberhart, R.C.: Particle swarm optimization. In: Proceedings of IEEE International Conference on Neural Networks, vol. 4, pp. 1942–1948 (1995)

[24] Kennedy, J., Eberhart, R.C.: Swarm Intelligence. Morgan Kaufmann Publishers, San Francisco (2001)

[25] Kleinberg, J.: An impossibility theorem for clustering. In: Advances in Neural Information Processing Systems (NIPS), pp. 446–453. MIT Press, Cambridge (2002)

[26] Lisi, F., Corazza, M.: Clustering financial data for mutual fund managment. In: Mathematical and Statistical Methods in Insurance and Finance, pp. 157–164. Springer, Milan (2007)

[27] MacQueen, J.B.: Some methods for classification and analysis of multivariate observations. In: Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, vol. 1, pp. 281–297. University of California Press (1967)

[28] Ng, R., Han, J.: CLARANS: A method for clustering objects for spatial data mining. IEEE Transactions on Knowledge and Data Engineering 14(5), 1003–1016 (2002)

[29] Omran, M.G.H., Engelbrecht, A.P.: Self-adaptive differential evolution methods for unsupervised image classification. In: Proceedings of IEEE Conference on Cybernetics and Intelligent Systems, pp. 1–6 (2006)

[30] Ostrovsky, R., Rabani, Y., Schulman, L.J., Swamy, S.: The effectiveness of lloyd-type methods for the k-means problem. In: Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science, pp. 165–176. IEEE Computer Society, Washington, DC (2006)

[31] Parsopoulos, K.E., Vrahatis, M.N.: Particle Swarm Optimization and Intelligence: Advances and Applications. Information Science Publishing (IGI Global), Hershey (2010)

[32] Paterlini, S., Krink, T.: Differential evolution and particle swarm optimisation in partitional clustering. Computational Statistics & Data Analysis 50, 1220–1247 (2006)

[33] Pavlidis, N., Plagianakos, V.P., Tasoulis, D.K., Vrahatis, M.N.: Financial forecasting through unsupervised clustering and neural networks. Operations Research - An International Journal 6(2), 103–127 (2006)

[34] Preparata, F., Shamos, M.: Computational Geometry: An Introduction. Springer, New York (1985)

[35] Puzicha, J., Hofmann, T., Buhmann, J.: A theory of proximity based clustering: Structure detection by optimisation. Pattern Recognition 33, 617–634 (2000)

[36] Tasoulis, D.K., `http://stats.ma.ic.ac.uk/d/dtasouli/public_html`

[37] Tasoulis, D.K., Plagianakos, V.P., Vrahatis, M.N.: Unsupervised clustering in mRNA expresion profiles. Computers in Biology and Medicine 36, 1126–1142 (2006)

[38] Tasoulis, D.K., Vrahatis, M.N.: The new window density function for efficient evolutionary unsupervised clustering. In: IEEE Congress on Evolutionary Computation, CEC 2005, vol. 3, pp. 2388–2394. IEEE Press, Los Alamitos (2005)

[39] Theodoridis, S., Koutroubas, K.: Pattern Recognition. Academic Press, London (1999)

[40] van der Merwe, D.W., Engelbrecht, A.P.: Data clustering using particle swarm optimization. In: Proceedings of the 2003 IEEE Congress on Evolutionary Computation, pp. 215–220 (2003)

[41] Vrahatis, M.N., Boutsinas, B., Alevizos, P., Pavlides, G.: The new *k*-windows algorithm for improving the *k*-means clustering algorithm. Journal of Complexity 18, 375–391 (2002)

[42] Xiong, H., Wu, J., Chen, J.: K-means clustering versus validation measures: A data-distribution perspective. IEEE Transactions on Systems, Man and Cybernetics - Part B: Cybernetics 39(2), 318–331 (2009)

[43] Zhao, Y., Karypis, G.: Criterion Functions for Clustering on High-Dimensional Data. In: Grouping Multidimensional Data Recent Advances in Clustering, pp. 211–237. Springer, Heidelberg (2006)