Adaptive Possibilistic Clustering

[illegible]

[illegible]

Abstract [illegible] [1]),

the fuzzy [2] and the possibilistic clustering methods [3].

In this work we deal with the popular case where the data vectors of the data set under study, $X \subset$ [illegible] form compact and hyperspherically shaped clusters. The most well known algorithms from the above categories that deal with this case are the k-means, the fuzzy c-means (FCM) and the possibilistc c-means (PCM), respectively [1]. In all these algorithms each cluster is represented by an l-dimensional cluster representative vector. All such algorithms start with some initial estimates of these vectors and (in principle) they move the cluster representatives to regions that are "dense" in data, via the optimization of appropriate cost functions. However, their philosophical background is quite different. In the k-means, each data vector belongs exclusively on a single cluster, while in the FCM a single data vector may belong to more than one clusters simultaneously, up to a certain degree, under the constraint that all these degrees sum up to one. Last, in the PCM

[illegible] [4].

Although PCM deals well with noisy points and outliers, compared to k-means and FCM, whose estimates for the representatives are significantly affected by the noise, it requires good initialization of both the cluster representatives and the η parameters that involves. Recall that, [3], each cluster is associated with an η parameter which, loosely speaking, is a measure of how the [illegible] of a cluster is spread around its representative. In practice, η's are estimated, based on the clustering results obtained by FCM which needs to be executed first [3] and are kept fixed during the execution of PCM. However, since a prerequisite for the FCM to provide good clustering results is the accurate knowledge of the number of clusters (which is not the case in practice), the estimates for η's are (in most cases) not very accurate. Consequently, this usually leads PCM to poor results. Some variants of PCM that try to address the problems of the original PCM have been proposed in [5], [6], [7].

the algorithm to estimate the (unknown in practice) true number of natural clusters. More specifically, as it will be explained below in detail, if the algorithm starts with a crude overestimate of the number of natural clusters, it gradually reduces this number as it progresses and finally provides the actual number of clusters. Obviously, this is a very important property that improves significantly clustering performance, as corroborated by extensive simulation results.

The rest of the paper is organized as follows. In Section 2, the FCM and PCM clustering algorithms are briefly reviewed. In Section 3, the new Adaptive PCM (APCM) clustering algorithm is presented. In Section 4, the performance of APCM is tested against the k-means, the FCM and the PCM algorithms. Finally, concluding remarks are provided in Section 5.

2. A BRIEF REVIEW OF FCM AND PCM

Let $X = \{\mathbf{x}_i \in$ [illegible] $i = 1, \ldots, N\}$ be a set of N, dimensional data vectors. Let also $\Theta = \{\boldsymbol{\theta}_j \in$ [illegible] $j = 1, \ldots, m\}$ be a set of m vectors that will be used for the representation of the clusters formed in X. In what follows, we consider only Euclidean distances.

2.1. Fuzzy c-means clustering (FCM)

Let $U = [u_{ij}]$, $i = 1, \ldots, N$, $j = 1, \ldots, m$ be an $N \times m$ matrix whose (i,j) element stands for the so called membership grade of $\mathbf{x}_i$ to the jth cluster, denoted by u_{ij} and represented by the vector $\boldsymbol{\theta}_j$. Let also $\mathbf{u}_i^T = [u_{i1}, \ldots, u_{im}]$ be the vector containing the elements of the ith row of U. Fuzzy c-means clustering (FCM) is based on the optimization of the following objective function

$$J_{FCM}(\Theta, U) = \sum_{i=1}^{N} \sum_{j=1}^{m} u_{ij}^q \|\mathbf{x}_i - \boldsymbol{\theta}_j\|^2$$

subject to the constraints (a) $u_{ij} \in [0,1]$ and (b) $\sum_{j=1}^{m} u_{ij} = 1$, where $\|\cdot\|$ denotes the Euclidean distance between vectors. The parameter q, called fuzzier, takes values greater than 1. The membership grade, u_{ij}, of the data point $\mathbf{x}_i$ to the jth cluster on the one hand and the cluster representatives on the other hand are iteratively estimated as

$$u_{ij} = 1 \Big/ \sum_{k=1}^{m} \left(\frac{\|\mathbf{x}_i - \boldsymbol{\theta}_j\|^2}{\|\mathbf{x}_i - \boldsymbol{\theta}_k\|^2} \right)^{\frac{1}{q-1}} \quad \text{and} \quad \boldsymbol{\theta}_j = \frac{\sum_{i=1}^{N} u_{ij}^q \mathbf{x}_i}{\sum_{i=1}^{N} u_{ij}^q},$$

respectively.

2.2. Possibilistic c-means clustering (PCM)

In possibilistic c-means clustering (PCM) the sum-to-one constraint is not imposed on the rows of U. Now, for u_{ij}'s we have that $u_{ij} \in [0,1]$, $i = 1, \ldots, N$, $j = 1, \ldots, m$, [illegible] $\max_j u_{ij} > 0$ and $0 < \sum_{i=1}^{N} u_{ij} < N$ [3]. In this framework, u_{ij} is interpreted as the degree of compatibility of $\mathbf{x}_i$ with $\boldsymbol{\theta}_j$. The aim of the PCM is to move the vectors $\boldsymbol{\theta}_j$ to regions that are dense in data points of X. This is carried out via the minimization of, among others, the following objective function [8] (note that this function is associated with the proposed algorithm presented in the next section):

$$J_{PCM}(\Theta, U) = \sum_{i=1}^{N} \sum_{j=1}^{m} u_{ij}^q \|\mathbf{x}_i - \boldsymbol{\theta}_j\|^2 + \sum_{j=1}^{m} \eta_j \sum_{i=1}^{N} (1 - u_{ij})^q$$

In this cost function, η_j's are positive parameters, each one associated with a cluster, which are set by the user and kept fixed during the execution of the algorithm. Loosely speaking, each one of them indicates how the influence of a cluster spreads around its representative; the smaller the value of η_j, the smaller the influence of the jth cluster around $\boldsymbol{\theta}_j$. The parameter η_j has a major impact on the clustering results and therefore, its accurate estimation is quite important. A recommended formula for estimating η_j is given in [3], is $\eta_j = \alpha \frac{\sum_{i=1}^{N} u_{ij}^q \|\mathbf{x}_i - \boldsymbol{\theta}_j\|^2}{\sum_{i=1}^{N} u_{ij}^q}$, where α is an appropriate threshold. Other such strategies have also been recommended (e.g. [3]).

Minimization of J_{PCM} with respect to u_{ij}'s and $\boldsymbol{\theta}_j$'s leads to the Algorithm 1. The termination criterion may be the q-th distance between $\boldsymbol{\theta}(t)$ and $\boldsymbol{\theta}(t-1)$ to become less than a prespecified threshold ε, where $\boldsymbol{\theta}$ is an $ml \times 1$ vector containing all $\boldsymbol{\theta}_j$'s.

Algorithm 1 The PCM algorithm

```
Fix: η_j, j = 1, ..., m
Initialize: θ_j ≡ θ_j(0)
t = 0
Repeat:
    Updating U
    for i = 1 to N
        for j = 1 to m
            u_ij = exp(−‖x_i − θ_j‖² / η_j)
        end for
    end for
    t = t + 1
    Updating Θ
    for j = 1 to m
        θ_j(t) = [illegible]
    end for
Until: a termination criterion is met
```

Note from the updating equations for $\boldsymbol{\theta}_j$'s, that all data vectors contribute to the estimation of each one of the representatives. However, the farthest ones contribute less, since

the corresponding u_{ij}'s are smaller for these vectors, as the updating equation for u_{ij}'s indicates. Nevertheless, while the update of u_{ij}'s is still in the early stages during which their estimates are not good enough yet, the estimates of $\boldsymbol{\theta}_j$'s in each iteration may not be accurate enough at these stages, affecting thus their final estimation. In addition, it can be seen that u_{ij}'s update is highly dependent from the η_j parameter.

3. THE ADAPTIVE PCM (APCM)

In this section, we will describe the modifications that need to be imposed on the PCM algorithm in order to implement the adaptation of both η's and the number of clusters. Note that the number of clusters m has to be set to a value greater - but not *much* greater - than the true number of the actual clusters formed in X. Potentially, the algorithm reduces m to the true number of clusters, by leading the cluster representatives in dense regions. Note, also, that even if the number of clusters m is initially underestimated, APCM will manage to uncover at least some of the dense regions, formed naturally in X.

3.1. Parameter initialization

As it has already been mentioned, the initialization of $\boldsymbol{\theta}_j$ and η_j is a crucial part of the algorithm, as it dramatically affects the final clustering result. In the proposed clustering scheme, the initialization of $\boldsymbol{\theta}_j$ is carried out using the Max-Min algorithm proposed in [9]. Assuming that the data vectors form cohesive clusters, this algorithm will, in principle, return at least one vector from each cluster ([9]), provided that m is sufficiently large. Max-Min algorithm determines m points of X that will be used as initial estimates of the representatives, by first determining the two most distant points of X[1]. Then the Max-Min algorithm proceeds by determining the remaining $m-2$ data points, as described in the sequel and assigning them to X_{re}, which is the set containing the points that will be used as initial cluster representatives. It determines for each $\mathbf{x} \in X - X_{re}$ its minimum distance $d_{min}(\mathbf{x})$ from all the points in X_{re}. Then, it identifies the point in $X - X_{re}$ which has the maximum value among all $d_{min}(\mathbf{x})$, $\mathbf{x} \in X - X_{re}$ and assigns it to X_{re}. The algorithm terminates when X_{re} contains m vectors.

After the initialization of the $\boldsymbol{\theta}_j$'s, we propose η_j's to be initialized as follows (see also Algorithm 2). The distance of each $\boldsymbol{\theta}_j \in X_{re}$ from its closest $\boldsymbol{\theta}_s \in X_{re} - \{\boldsymbol{\theta}_j\}$, denoted by $d_{min}(\boldsymbol{\theta}_j)$, is determined first and then the algorithm sets $\eta_j = \frac{d_{min}(\boldsymbol{\theta}_j)/2}{-\log \beta}$, where $\beta \in (0,1)$ is an appropriately chosen parameter. To unravel the rationale of the last equation, we solve it with respect to β, obtaining

$$\beta = \exp\left(-\frac{d_{min}(\boldsymbol{\theta}_j)/2}{\eta_j}\right).$$

[1] [illegible]

Algorithm 2 Initialization of η_j's

for $j = 1$ to m
  Determine: $d_{min}(\boldsymbol{\theta}_j) = \min_{\boldsymbol{\theta}_s \in X_{re}-\{\boldsymbol{\theta}_j\}} \|\boldsymbol{\theta}_j - \boldsymbol{\theta}_s\|^2$
  Set: $\eta_j = \frac{d_{min}(\boldsymbol{\theta}_j)/2}{-\log \beta}$
end for

Comparing this with the updating equation for u_{ij}'s of the PCM algorithm, one can deduce that η_j is chosen such that, for a given data point $\mathbf{x}_i$ that lies at distance $d_{min}(\boldsymbol{\theta}_j)/2$ from $\boldsymbol{\theta}_j$, its degree of compatibility (u_{ij}) with the jth cluster equals to β. As it has been verified experimentally, typical values for β that lead to satisfactory results are in the range [illegible]. The experiments showed also that β depends on how densely the natural clusters are located; low density requires smaller values of β, while high density requires larger values of β.

3.2. Parameter adaptation

Parameter adaptation in the proposed APCM clustering algorithm refers to the adjustment of the number of clusters and the adaptation of η's, which are two interrelated processes. These processes can be attached in the main while-loop of the classic possibilistic algorithm and the whole APCM becomes as shown in Algorithm 3. Here, $label$ is a N-dimensional vector, whose ith component contains the index of the cluster which is most compatible with the data vector $\mathbf{x}_i$ (i.e. the cluster C_j for which $u_{ij} = \max_{r=1,\ldots,m} u_{ir}$), n_j denotes the number of the data points $\mathbf{x}_i$, that are most compatible with the jth cluster and $\boldsymbol{\mu}_j$ is the mean vector of these data points. In words, the "Possible cluster elimination" part of APCM examines if the index j of a cluster C_j appears in the vector $label$. If this is the case (i.e. if there exists at least one vector $\mathbf{x}_i$ that is most compatible with C_j), C_j is not eliminated. Otherwise, C_j is eliminated. Also, η_j is estimated as the variance of the most compatible with C_j points, around their mean $\boldsymbol{\mu}_j$.

Let us first comment on the proposed updating mechanism of η_j's. Note that, although their updating formula resembles to that used in the classical PCM, as well as many of its variants, it differs from them in two distinctive points. First η_j's are updated taking into account *only* the data vectors that are most compatible to C_j. Second, the distances involved in the formula are between a data vector and the mean vector $\boldsymbol{\mu}_j(t)$, *not* the representative $\boldsymbol{\theta}_j(t)$. This allows more accurate estimates for η_j's. It is also noted that, in the case where there are two or more clusters, that are equally compatible with a specific $\mathbf{x}_i$, then $\mathbf{x}_i$ will contribute to the determination of the η parameter of *only* one (arbitrarily chosen) of them.

Let us now comment on the immunity of the proposed algorithm to overestimates on the number of clusters. In such a scenario and taking into account that a possibilistic type algorithm moves the representatives in dense regions, the prob-

Algorithm 3 The APCM algorithm

Initialize: $\theta^{(0)}$ (as in Algorithm 1)
Initialize: $\Theta^{(0)}$ (as in Algorithm 2)
$t = 0$
Repeat:
Update: U (as in Algorithm 1)
$t = t + 1$
Update: Θ (as in Algorithm 1)
%Possible cluster elimination

```
for i = 1 to N
   Determine: label(i) = j, 1 ≤ j ≤ m
   Set: l a b e l r(i) = label(i)
end for
for j = 1 to m
   If j ∉ label then
      Remove θ_j
      m = m − 1
   end if
end for
```

%Adaptation of η

```
for j = 1 to m
   η_j(t) = (1/n_j(t)) Σ_{x_i : u_ij(t) = max_{r=1,...,m} u_ir(t)} ||x_i − θ_j(t)||
end for
```

Until: a termination criterion is met

ability to select as representative at least one point in each dense region is increased, due to the way the representatives are initially selected (via the Max-Min Algorithm). Then, due to the possibilistic nature of APCM, it is guaranteed, in principle, that the number of the representatives which move to a specific dense region will be reduced to a single one.

In order to get some further insight to the way the algorithm works, assume that two cluster representatives θ_r, θ_s almost coincide but let say $\eta_r > \eta_s$. Then, for a given data point $\mathbf{x}_i$, it is $\frac{\|\mathbf{x}_i - \theta_r\|}{\eta_r} < \frac{\|\mathbf{x}_i - \theta_s\|}{\eta_s}$, which implies that $u_{ir} > u_{is}$, considering the updating equation of u_{ij} in the possibilistic algorithm. Loosely speaking, between two cluster representatives with different η parameters, the one with the greater η has a stronger influence around it. Thus, if θ_r, θ_s almost coincide, the influence of the one with the smaller η will be vanished by the influence of the one with the greater η in the sense that $u_{ij} \approx 0$, for all data points $\mathbf{x}_i$.

4. EXPERIMENTAL RESULTS

In this section, we test the proposed method through a synthetic data set and two real data sets. Moreover, we compare the results with those obtained from k-means, FCM and PCM. Note that, because the PCM algorithm leads, in some cases, to coincident clusters, the clustering result is extracted taking into account only the truly "different" clusters. In addition, in order to make a fair comparison between all clustering algorithms, the representatives $\theta_j^{(0)}$ are initialized based on the Max-Min scheme, in all of them.

Table 1. The results of the synthetic data set

	$m_{initial}$	m_{final}	Rand Measure
k-means	3	3	97.12%
k-means	5	5	82.57%
FCM	3	3	96.58%
FCM	5	5	82.20%
PCM	3	2	83.98%
PCM	5	2	83.98%
PCM	10	2	83.98%
APCM	5	3	96.79%
APCM	10	3	96.79%

4.1. A synthetic data set experiment

Let us consider a two-dimensional data set consisting of $N = 1000$ points that form three clusters. Each cluster is modelled by a normal distribution. Their means are $E_1 = [1.07, 1.35]^T$, $E_2 = [7.24, 8.07]^T$ and $E_3 = [5.54, 1.34]^T$, respectively, while their (common) covariance matrix is $0.8 \cdot I_2$, where I_2 is the 2×2 identity matrix. A number of 400 points have been generated by each one of the first and the third distribution, while 200 points have been generated by the second one. Note that the second and the third clusters are close enough to each other, while they are far away from the first one. Each data point is assigned to a cluster, utilizing the U matrix, as follows: $\mathbf{x}_i$ is assigned to cluster θ_j if $j = \arg\max_{r=1,\ldots,m} u_{ir}$. Figs. 1 (a), (b) show the clustering results obtained using the k-means algorithm with $m = 3$ and $m = 5$, respectively. Similarly, in Figs. 1 (c), (d) we present the corresponding results for FCM. Fig. 1 (e) depicts the performance of PCM for $m = 5$, while in addition, it shows the circled regions, centered at each θ_j and having radius equal to η_j, in which θ_j has increased influence. Finally, Fig. 1 (f) shows the results of APCM with $m = 5$ and $K = 0.1$.

In order to compare a clustering with the true data label information, we use the Rand Measure described in [1]. Table 1 shows the results of the previously mentioned algorithms for the synthetic data set, where $m_{initial}$ and m_{final} denote the initial and the final number of clusters, respectively. It presents the Rand Measure of each algorithm, which results from the comparison of the $label$ vector, and the true cluster identity of the vectors.

As it is deduced from Fig. 1 and Table 1, when k-means and FCM are initialized by the (unknown in practice) true number of clusters ($m = 3$), the provided results are very satisfactory. However, any deviation from this value causes a significant degradation to the obtained clustering quality. On the other hand, the classical PCM fails to unravel the under-

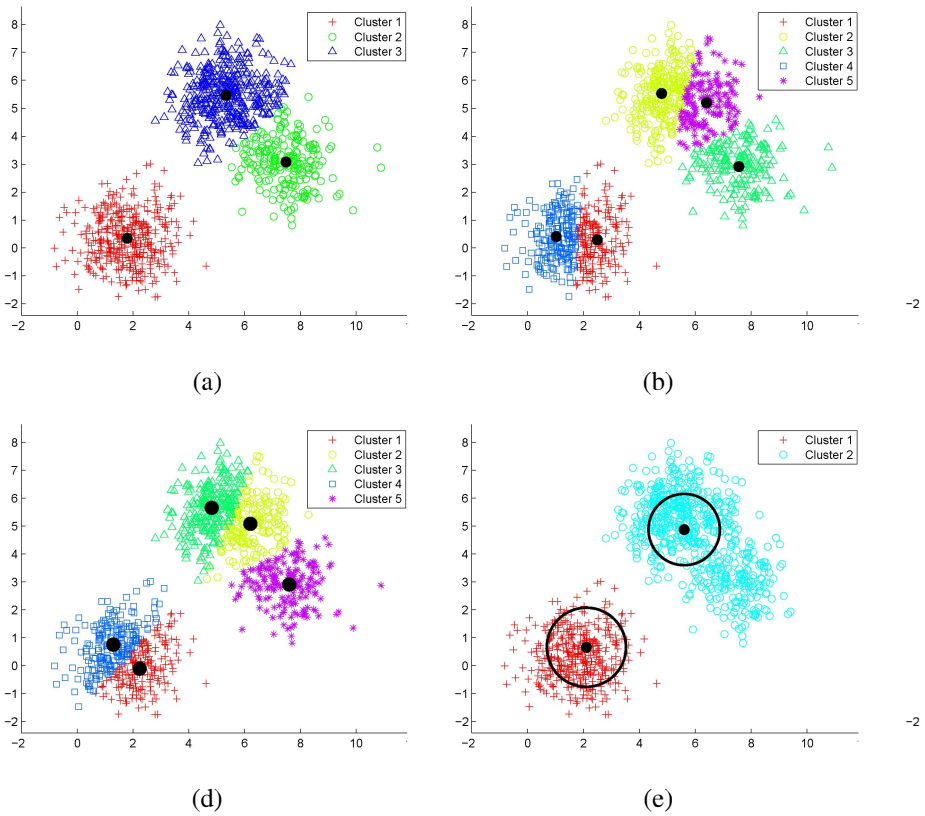

Fig. 1. Clustering results of k-means with, (a) $m = 3$ and (b) $m = 5$, FCM with, (c) $m = 3$ and (d) $m = 5$, (e) PCM with $m = 5$ and (f) APCM with $m = 5$. Note that in PCM the clustering result is extracted taking into account only the truly "different" clusters. Bolded dots represent the final clusters' representatives.

ying clustering structure, due to the fact that two clusters are close enough to each other and the algorithm does not have the ability to adapt η_j's in order to cope with this situation. Finally, the proposed APCM constantly produces very accurate results for various initial values of m.

4.2. Real data set experiments

In this sub-section, the Iris data set and the Wine data set from UCI Library database [11] are considered. In order to efficiently utilize all the information of the l features of these real data sets, we shall normalize them as

$$\hat{x}_{ip} = \frac{x_{ip} - \frac{1}{N}\sum_{r=1}^{N} x_{rp}}{\sqrt{\frac{1}{N-1}\sum_{r=1}^{N}(x_{rp} - \frac{1}{N}\sum_{r=1}^{N} x_{rp})^2}},$$

where $i = 1, \ldots, N$, $p = 1, \ldots, l$.

Iris data set: This set consists of $N = 150$ 4-dimensional data points that form three classes, each one having 50 points. In this data set, two classes are overlapped thus one can argue whether the number of clusters m is 2 or 3.

As it is shown in Table 2, APCM ends up with $m_{final} = 2$ clusters independently of the initial number of clusters $m_{initial}$. Since algorithms following the possibilistic philosophy detect dense regions, the APCM ends with $m_{final} = 2$ clusters, which is an acceptable result, due to the nature of the data set. On the other hand, k-means and FCM perform better provided that they have been supplied with the true number of the underlying classes. However, this situation changes as we deviate from this value. In addition, it is interesting to note that for $m_{initial} = 2$, both k-means and FCM provide the same clustering with that produced by APCM. This is an indication that APCM provides constantly the best possible two-cluster clustering for all initial values for m and $\beta = 0.1$. Finally, the original PCM needs a significantly greater than 3 initial value for m in order to be competitive with the other schemes.

Wine data set: This set consists of $N = 178$, 13-dimensional data points that stem from three classes, the first with 59 points, the second with 71 and the third one with 48 points.

In the Wine data set, an exhaustive feature selection procedure preceded the clustering stage, due to the high dimensionality of this data set in relation to the small number of data

Table 2 The results of the real Iris data set

t t t]zfl	m ht]zfl	Rand Measure
k-means	2	2	77.63%
k-means	3	3	83.22%
k-means	5	5	76.61%
k-means	8	8	75.38%
FCM	2	2	77.63%
FCM	3	3	83.68%
FCM	5	5	77.43%
FCM	8	8	75.81%
PCM	2	1	32.89%
PCM	3	1	32.89%
PCM	5	1	32.89%
PCM	8	2	74.98%
APCM	4	2	77.63%
APCM	5	2	77.63%
APCM	8	2	77.63%
APCM	10	2	77.63%

Table 3. The results of the real Wine data set

t t t]zfl	m ht]zfl	Rand Measure
k-means	3	3	99.25%
k-means	5	5	89.19%
k-means	8	8	82.30%
FCM	3	3	99.25%
FCM	5	5	83.75%
FCM	8	8	78.22%
PCM	3	1	33.80%
PCM	5	1	33.80%
PCM	8	1	33.80%
APCM	4	3	98.37%
APCM	5	3	98.37%
APCM	8	3	98.37%

points. In order to make a fair comparison between the clustering algorithms, as presented in Table. 3, we considered the optimal combination of features for each algorithm. It turned out that the best result for each algorithm was obtained using different combinations of eight features. As before, Table 3 shows that APCM (β 0 1) results in ~ht]zfl 3 clusters independently of the initialization of the number of clusters ~t t t]zfl while the other clustering algorithms require the knowledge of the actual number of clusters, in order to extract satisfactory results.

5. CONCLUSIONS

In this paper a novel possibilistic clustering algorithm (APCM) has been proposed. The algorithm encompasses a proper initialization and a new updating mechanism for the η parameters and is immune to overestimates on the actual number of existing clusters. These features make the algorithm very flexible in tracking the clustering environment under study. The performance of the proposed algorithm against k-means, FCM and the original PCM has been assessed using both synthetic and real data sets. In all these experiments, it is shown that APCM has a steadily good performance irrespective of the initial number of clusters, which is not the case with k-means and FCM. In addition, APCM constantly provides better results compared to the original PCM.

6. REFERENCES

[1] S. Theodoridis and K. Koutroumbas, Pattern Recognition, 4th edition, Academic Press, 2009.

[2] J. C. Bezdek, Pattern Recognition with Fuzzy Objective Function Algorithms, Plenum, 1981.

[3] R. Krishnapuram and J. M. Keller, "A possibilistic approach to clustering," IEEE Transactions on Fuzzy Systems, vol. I, pp. 98–110, 1993.

[4] M. Barni, V. Cappellini, and A. Mecocci, "A possibilistic approach to clustering," IEEE Transactions on Fuzzy Systems, vol. IV, pp. 393–396, 1996.

[5] M.-S Yang and K.-L. Wu, "Unsupervised possibilistic clustering," Pattern Recognition, vol. 39, pp. 5–21, 2006.

[6] K. Treerattanapitak and C. Jaruskulchai, "Possibilistic exponential fuzzy clustering," Journal of computer science, vol. 28, pp. 311–321, 2013.

[7] J.-S. Zhang and Y.-W. Leung, "Improved possibilistic c-means clustering algorithms," IEEE Transactions on Fuzzy Systems, vol. 12, pp. 209–217, 2004.

[8] R. Krishnapuram and J. M. Keller, "The possibilistic c-means algorithm: insights and recommendations," IEEE Transaction on Fuzzy Systems, vol. IV, pp. 385–393, 1996.

[9] B. Mirkin, Clustering for Data Mining: A Data Recover Approach, Chapman Hall, 2005.

[10] O. Egecioglu and B. Kalantari, "Approximating the diameter of a set of points in the euclidean space," Information Processing Letters, vol. 32, pp. 205–211, 1989.

[11] UCI Library database, http://archive.ics.uci.edu/ml/datasets.html.