

# A non-parametric heuristic algorithm for convex and non-convex data clustering based on equipotential surfaces

Farhad Bayat<sup>a,\*</sup>, Ehsan Adeli Mosabbeh<sup>b</sup>, Ali Akbar Jalali<sup>a</sup>, Farshad Bayat<sup>c</sup>

<sup>a</sup> Electrical Engineering Department, Iran University of Science and Technology, Tehran, Iran

<sup>b</sup> Computer Engineering Department, Iran University of Science and Technology, Tehran, Iran

<sup>c</sup> Computer Engineering Department, Azad University of Zanjan, Zanjan, Iran

## ARTICLE INFO

### Keywords:

Clustering  
Classification  
Convex  
Non-convex  
Potential functions  
Unsupervised and point location

## ABSTRACT

In this paper, using the concepts of field theory and potential functions a sub-optimal non-parametric algorithm for clustering of convex and non-convex data is proposed. For this purpose, equipotential surfaces, created by interaction of the potential functions, are applied. Equipotential surfaces are the geometric location of the points in the space on which the potential is constant. It means all points in each surface were affected the same by the field. Regarding this concept and other characteristics of equipotential surfaces, the outcome of this method will be an optimal solution for the clustering problem. But with regard to the existence of several parameters requiring to be set in the algorithm, finding the global optimal solution leads to a high computational complexity and therefore is not practical. Thus by applying some considerations and approximations, the resulting outcome will be a sub-optimal solution, while appropriate setting of the parameters causes the result to be closer to the global optimal solution. The advantage of this method is that it does not need any external parameter setting, such as number of clusters. To this end, an automatic parameter setting algorithm is suggested based on an optimal clustering index. Simulation results for a number of standard datasets, illustrate the superb performance of this method, especially for non-convexly scattered data. All mentioned characteristics of this method are widely demanded in different scientific areas. In this case it has been utilized in the well-known Point Location Problem (PLP) to reduce computational complexity.

© 2009 Elsevier Ltd. All rights reserved.

## 1. Introduction

Clustering, as an unsupervised pattern classification method, has an important role in data analysis and refinement. Classification, in general, has a wide domain of applications. It includes all from biology and biomedical imaging sciences to military and archeology applications. According to the sensitiveness and limitations of some mentioned applications, such as medical and military applications, data with certain classes for use in supervised classification algorithms, are not simply available. Therefore, obtaining powerful and reliable unsupervised classification algorithms are very important. So in the recent decades, clustering problem as a tool for pattern analysis, classification, decision making and information extraction and retrieval, has attracted the attention of many researchers. Several approaches and points of view are presented in the literature. Each of these approaches is based on a certain criterion, and has its own advantages and disadvantages. In

general, a comprehensive method and criterion for optimal clustering of any kind of data does not exist.

In Dubes et al. (1976), the comparison of different clustering algorithms was done using the criteria presented in Fisher et al. (1971). A review of the results of applying some existing limitations on data sources to enhance the clustering process performance was done by Titterton et al. (1985). Limitations applied in this method are based on the combination of an unknown number of probability density functions with multivariable Gaussian functions, in which clustering tries to extract the probability density functions and their parameters. The development of clustering applications is presented for pattern recognition by Anderberg (1973), image processing by Jain et al. (1996) and information retrieval by Salton (1991) and Rasmussen (1992).

General requirements of a data clustering system are: scalability, ability to recognize different shape, size and density clusters, robustness versus noise and disturbance, least number of input parameters, etc. Based on these criteria, efficiency and performance of clustering algorithms are determined. As a result, old hierarchical clustering algorithms are of very high computational complexity and qualitatively weak (George & et al., 1999). For instance, Complete-Link method is biased for spherical clusters and Single-Link

\* Corresponding author. Tel.: +98 2177240487; fax: +98 2177240486.

E-mail addresses: [fbayat@iust.ac.ir](mailto:fbayat@iust.ac.ir) (F. Bayat), [eadeli@iust.ac.ir](mailto:eadeli@iust.ac.ir) (E.A. Mosabbeh), [ajalali@iust.ac.ir](mailto:ajalali@iust.ac.ir) (A.A. Jalali), [farshad.bayat@gmail.com](mailto:farshad.bayat@gmail.com) (F. Bayat).

undertakes chaining (Oyang, 2001), while newer clustering techniques, combining hierarchical and partitioning methods (Gan, 2003; Gan et al., 2003), result in more quality and less computational complexity. A hybrid genetic fuzzy  $k$ -modes algorithm has been proposed by Gan, Wu, and Yang (2009). They have optimized fuzzy  $k$ -modes clustering algorithm using GA, to avoid being stuck in local optima of the  $k$ -modes clustering. Hsu and Huang (2008) have presented a learning based algorithm to cluster data with either categorical or numeric values. They have used a modified version of an adaptive resonance theory (ART) unsupervised neural network for this purpose. Wang and et al. (2009) have also presented a clustering algorithm based on the extension theory and genetic algorithm, EGA.

One of the weak sides of most clustering algorithms is the challenging case of non-convexly scattered data and the discussion of how each algorithm behaves in such a circumstance.

Different metrics for clustering analysis have been proposed in different works, such as entropy, purity and mutual information. Park and Jun (2009) have proposed a  $k$ -means-like algorithm, known as  $k$ -medoid, which is efficient in the complexity point of view. The authors have proposed a Random Index to evaluate the performance of the clustering. Wei, Lee, and Hsu (2003) have done an empirical comparison of some partition-based clustering techniques. They have introduced some characteristics of the data such as data size, number of clusters, cluster distinctness, cluster asymmetry and data randomness. They have analyzed the effects of changes in each of these parameters in the clustering results. Also Wu et al. (2009) have introduced some cluster validation measures to be used to evaluate  $k$ -means clusters. Normalized variation of information (VI), van Dongen criterion (VD) and Mirkin metric (M) are the measures used as cluster quality quantifiers. They showed that using these metrics can avoid bias in the clustering process. All these metrics and cluster quality measure used in these works are for convex data sets. Some studies (Mitra, Pal, & Siddiqi, 2003; Pal, Ghosh, & Uma Shankar, 2000) have introduced metrics also used for non-convex data clusters. Such a same metric is used in this paper and will be explained more, later.

In this paper, using the idea of potential functions and equipotential surfaces arising from fields' interaction a clustering method for both convex and non-convex data is presented. In this method, a potential function is assigned to each data sample. Then, by the fields' interactions in feature space and extraction of the equipotential surfaces the clustering procedure can be conducted. It is very important to know that most of the unsupervised classification techniques are based on the degree of similarity in data sample feature vector, such that members of a data class generally have the most similarity. Regarding this and the fundamental concepts of fields' theory in physic, applying the equipotential surfaces as the clusters discriminant boundaries the optimal solution would be achieved. This is because the equipotential surfaces introduce the geometric locations in the space on which all points have the same average membership or similarity to the class inside and outside the boundary. Thus, by proper setting of the reference potential level as the decision boundary, optimal solution could be obtained. Finally, using cluster measures a simple algorithm is presented which can set the reference potential level automatically. This approach determines the number of clusters itself. So, no parameters are left to be set externally. If one wants to determine the number of the clusters, he/she can change the reference potential by trial and error. Moreover, the proposed method enables us to construct a hierarchical non-parametric data clustering which is widely demanded in different areas. As an application, this method has been used to reduce computational complexity of the Point Location Problem (PLP) which is the most time consuming part in the Explicit Model Predictive Control (Bayat et al., 2009).

This paper is organized as follows: in Section 2 concepts and some definitions used in the subsequent sections are brought up.

Then, in Section 3 potential functions are presented. After that using the potential functions concept, the clustering algorithm is demonstrated in Section 4. Finally, Section 5 illustrates some examples pondering the performance of the proposed algorithm.

## 2. Mathematical background and definitions

As described above, the method presented in this paper is based on the concept of field, which is one of the basics in Physics and has a vast number of applications. Different kinds of fields on Physics include magnetic, electric, gravity, and nucleus power fields. Although each of these instances has own different definitions, the common concept relating them is that instead of studying the mutual interaction between components (electric particles, for example), we can use the influence of the field on the components in that working set. In what follows, we utilize this simple concept to extract an optimal method for data clustering. The following definitions are evident and used in the algorithm definition.

**Definition 1.** Space  $(D, \|\cdot\|)$  in linear vectors set  $D \in \mathfrak{R}^n$  and real function  $\|\cdot\| : D \rightarrow \mathfrak{R}_+$  are called a norm space if all following conditions fulfill:

- (i)  $\|\mathbf{X}\| \geq 0, \forall \mathbf{X} \in D,$
- (ii)  $\|\mathbf{X}\| = 0 \iff \mathbf{X} = \mathbf{0}, \forall \mathbf{X} \in D,$
- (iii)  $\|\alpha\mathbf{X}\| = |\alpha| \cdot \|\mathbf{X}\|, \forall \mathbf{X} \in D, \alpha \in \mathfrak{R},$
- (iv)  $\|\mathbf{X} + \mathbf{Y}\| \leq \|\mathbf{X}\| + \|\mathbf{Y}\|, \forall \mathbf{X}, \mathbf{Y} \in D.$

**Definition 2.** Assume  $i = 1, \dots, N, j = 1, \dots, n, D \in \mathfrak{R}^n$  and  $D = \{\mathbf{X}_i \in \mathfrak{R}^n | \mathbf{X}_i = (x_{i1}, \dots, x_{in}), x_{ij} \in \mathfrak{R}\}$ , then the norm space  $(D, \|\cdot\|_2)$  is called a limited norm space relative to  $\|\cdot\|_2$  if and only if there exists a scalar  $0 < h < \infty$  such that:

$$\|\mathbf{X}_i\|_2 \leq h \text{ for all } \mathbf{X}_i \in D \text{ where } \|\mathbf{X}_i\|_2^2 = \sum_{j=1}^n x_{ij}^2.$$

**Definition 3.** Scalar function  $V(\mathbf{X}) : \mathfrak{R}^n \rightarrow \mathfrak{R}$  is called a potential function if:

- (i)  $V(\mathbf{X})$  is a continuous smooth function in the given limited norm space (later we will find out that this space is in fact the feature vector space).
- (ii)  $V(\mathbf{X})$  is isotopic, i.e., it has symmetric behavior and characteristics in all dimensions.
- (iii) If  $V_i(\mathbf{X})$  is the potential function for component  $\mathbf{X}_i$ , increasing  $\|\mathbf{X} - \mathbf{X}_i\|_2$  should cause  $V_i(\mathbf{X})$  to decrease and  $V_i(\mathbf{X}) \rightarrow 0$  for each  $\|\mathbf{X} - \mathbf{X}_i\|_2 \rightarrow \infty$ .

In what follows, assuming that the space regarding the feature vector in the clustering problem is a limited norm space, clustering algorithm based on the potential function could be extracted.

## 3. Establishing the proper potential function

Once more assume a vector space with limited norm for the feature vector:

$$D = \{\mathbf{X}_i \in \mathfrak{R}^n | \mathbf{X}_i = (x_{i1}, \dots, x_{in}), x_{ij} \in \mathfrak{R}, i = 1, \dots, N, j = 1, \dots, n \tag{1}$$

where  $\mathbf{X}_i \in \mathfrak{R}^n$  is the feature vector for the  $i$ th sample, and  $x_{ij} \in \mathfrak{R}$  is the  $j$ th feature in the  $i$ th feature vector. Also 'N' is the number of patterns and 'n' the number of features for each pattern.

The potential function for the  $i$ th sample, considering the conditions in Definition 3 will be:

$$V_i(\mathbf{X}) = e^{-B\|\mathbf{X}-\mathbf{X}_i\|_2^2}, \quad \mathbf{X} = (x_1, \dots, x_n) \in \text{feature space} \quad (2)$$

where  $B$  is a constant positive coefficient, and  $V_i(\mathbf{X})$  is the potential arising from sample  $\mathbf{X}_i$  in location  $\mathbf{X}$  from feature space. Therefore,  $V_i(\mathbf{X})$  could be considered as the membership value of point  $\mathbf{X} = (x_1, \dots, x_n)$  from the feature space to the feature vector  $\mathbf{X}_i$ , such that its value in  $\mathbf{X} = \mathbf{X}_i$  will be the maximum value and increasing distance  $\|\mathbf{X} - \mathbf{X}_i\|_2$  causes the membership value to decrease. In Fig. 1, a diagram of a potential function with  $n = 2$ ,  $\mathbf{X}_i = (0, 0)$  and  $B = 1$  is drawn.

As far as clustering is basically putting similar data in a same class, the membership value of a sample point in feature space could be used as the similarity measure. To this end, the average membership function could be defined as:

$$\tilde{V}(\mathbf{X}) = \frac{A}{N} \sum_{i=1}^N V_i(\mathbf{X}) \quad (3)$$

The constant coefficient 'A' is chosen such that  $0 \leq \tilde{V}(\mathbf{X}) \leq 1$ . Finally, with regard to the potential function represented in 2 (a descending monotone function with a global maximum in  $\mathbf{X} = \mathbf{X}_i$  (Fig. 1),  $\tilde{V}(\mathbf{X})$  will become a limited function with several local maximums. The location and circumstance of these local maximums are determined by the density and distribution of the data in the feature space.

#### 4. Clustering algorithm

In order to extract a clustering algorithm using the average membership function presented in Eq. (3), the concept of equipotential surfaces could be utilized. Equipotential surfaces are the geometric locations of points in the space that have a same average membership value and are formed as closed islands around local maximums. Thus, considering the reference potential level ( $V_{ref}$ ), number and shape of the islands differ. Points inside the area of the equipotential surface have a average potential value more than  $V_{ref}$ . Island Boundary Potential ( $V(\mathbf{X}) > V_{ref}$ ). As a result, points inside each island share the most similarity within the feature space and therefore, each island could be considered as a cluster with its centroid on the maximum potential location.

With this definition, when  $V_{ref}$  changes from  $\tilde{V}_{min}$  to  $\tilde{V}_{max}$ , the number of islands (clusters) changes from 1 (including all samples) to 'N' (each sample a single cluster). In the following section, details and stages of the clustering algorithm are presented. First,

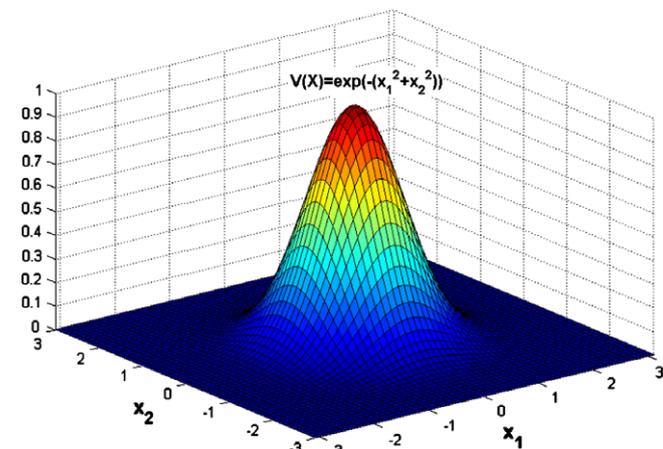


Fig. 1. Diagram of a potential function with  $n = 2$ ,  $\mathbf{X}_i = (0, 0)$  and  $B = 1$ .

the algorithm is given for the special case of  $n = 2$  (feature vector has two elements for each sample), and then the algorithm is generalized for all cases.

#### 4.1. Stages of the clustering algorithm

As far as, the case  $n = 2$  is more comprehensible because of the ease of its graphical representation, and also because the case  $n = 2$  could give the solution in lots of practical applications, the clustering algorithm here is based on the case  $n = 2$  and then is developed for general case.

**Stage 1.** Choose  $dx_1$  and  $dx_2$ , the search steps in feature space. Choosing these parameters should be done based on the expected resolution and accuracy. One good initial guess for  $dx_i$  is the half of minimum distance between any two points in the  $i$ th axis direction.

**Stage 2.** Divide the feature space considering  $dx_2, dx_1$  within the boundary  $x_{1min} \leq x_1 \leq x_{1max}$  and  $x_{2min} \leq x_2 \leq x_{2max}$ , as shown in Fig. 2.

$x_{imax}, x_{imin}$  is the variation boundary for the  $i$ th element in the feature space ( $i$ th feature) in space 'D', with assumption that 'D' is a limited norm space for sure  $x_{imax}, x_{imin}$  exist. The 'B' coefficient in Eq. (2) is a free tuning parameter. According to the authors' experience an empirical rule that generally issues a good result is:

$$B \geq \frac{N}{\sqrt[n]{\prod_{i=1}^n (x_{imax} - x_{imin})}} \quad (4)$$

Note that Eq. (4) has a root in the geometrical density of space partitions.

Assuming  $q = \left\lceil \frac{x_{2max} - x_{2min}}{dx_2} \right\rceil$ ,  $p = \left\lceil \frac{x_{1max} - x_{1min}}{dx_1} \right\rceil$ , stage 2 is the same as generating a  $q \times p$  matrix in which element  $(i, j)$  introduces vector  $(x_{1i}, x_{2j})$  from the feature space and its value equals the average potential in that point. This matrix is called the potential matrix (PM $_{q \times p}$ ) and is determined using the following equation:

$$PM(i, j) = \tilde{V}(\mathbf{X}) \quad (5)$$

where  $(i = 1, \dots, p)$ ,  $(j = 1, \dots, q)$ ,  $\mathbf{X} = (i \cdot dx_1, j \cdot dx_2)$ .

**Stage 3.** Using the PM matrix and choosing an appropriate value for  $V_{ref}$  in  $[0 \ 1]$ , matrix NPM (Normalized Potential Matrix) is defined as:

$$NPM(i, j) = \text{sign}(PM(i, j) - V_{ref}) \quad (6)$$

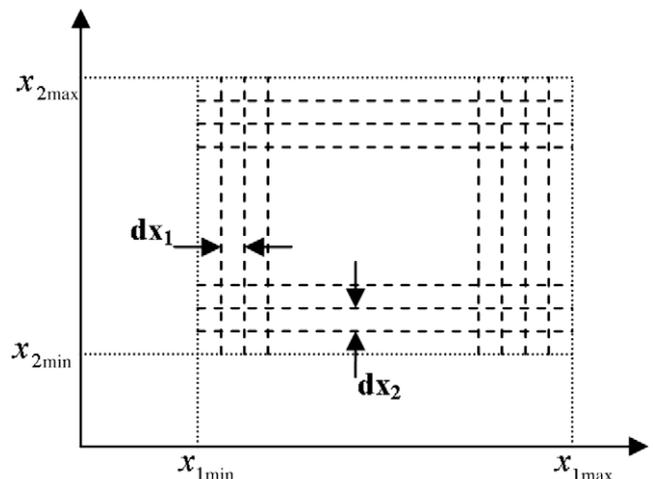


Fig. 2. The feature space division in the boundary  $x_{2min} \leq x_2 \leq x_{2max}$ ,  $x_{1min} \leq x_1 \leq x_{1max}$ .

where  $sign(\cdot)$  is defined as:

$$sign(s) = \begin{cases} 1, & s \geq 0 \\ 0, & s < 0 \end{cases}$$

**Stage 4.** As an example assume that the resulting matrix from stage 3 is as what is illustrated in Fig. 3.

In such a case, elements of the NPM matrix having a value of 1 are the points lying in the equipotential surfaces (for the chosen  $V_{ref}$ ). Now for the clustering purpose the following step should be taken into account:

- (i) Two data  $a = (x_{1i}, x_{2j})$ , and  $b = (x_{1r}, x_{2t})$ , with  $(i, r \in [1, \dots, p])$ ,  $(j, t \in [1, \dots, q])$  are members of a same cluster iff in the matrix NPM, there exists a path with non-zero elements form  $a$  to  $b$ .
- (ii) Suppose that there exists  $N_c$  different clusters in the  $L$ th iteration, if data  $c = (x_{1\alpha}, x_{2\beta})$  connected to any of the  $N_c$  isolated clusters through a non-zero element in the matrix, 'c' is a part of that cluster, otherwise in the case that 'c' is not connected to any of the  $N_c$  clusters, 'c' is considered as a new cluster.
- (iii) Considering the (i) and (ii), in order to reduce the mass of computation, we take in use this fact that there is a connection between all elements inside the each cluster. Therefore, it is sufficient to consider a single element for each cluster as a representative and when checking whether or not to join data 'c' to this cluster just a single path examination suffices.

**Stage 5.** After determining the location and number of clusters, if all the considered conditions and requirements are met, the optimal solution is obtained. Otherwise, we need to return to stage 3 and change  $V_{ref}$  to meet the required conditions.

In order to set  $V_{ref}$ , it is very important to take into account that increasing  $V_{ref}$  from 0 to 1 yields to changing the number of clusters from 1 (all data samples as a single cluster) to 'N' (each data point as a cluster for itself).

#### 4.2. Clustering algorithm: general case

Extending this algorithm for the general case is straightforward. To this end, we only need to replace the followings in the algorithm for the case  $n = 2$ :

**Stage 1:** Choosing search steps  $dx_1, \dots, dx_n$ , the same as the previous case.

**Stage 2:** Dividing the feature space based on  $dx_i$  ( $i = 1, \dots, n$ ) in interval  $x_{imin} \leq x_i \leq x_{imax}$ .

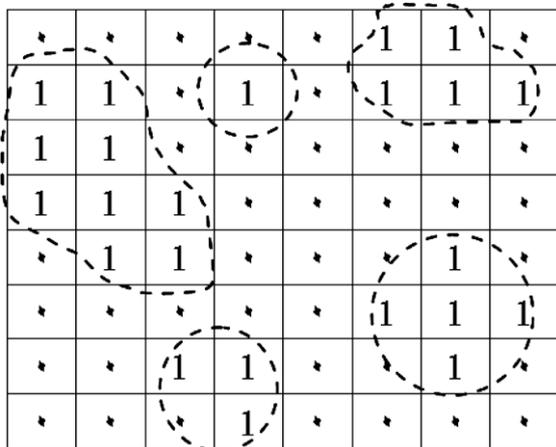


Fig. 3. NPM matrix acquired from stage 3.

**Stage 3:** Assuming  $p_i = (x_{imax} - x_{imin})/dx_i$ ,  $i = 1, \dots, n$  and choosing an appropriate  $V_{ref}$ , matrices PM and NPM with dimension  $p_1 \times p_2 \times \dots \times p_n$ , would be determined using:

$$\begin{aligned} PM(r_1, r_2, \dots, r_n) &= \tilde{V}(\mathbf{X}) \\ NPM(r_1, r_2, \dots, r_n) &= sign(PM(r_1, r_2, \dots, r_n) - V_{ref}) \\ \mathbf{X} &= (r_1 \cdot dx_1, \dots, r_n \cdot dx_n), \quad (1 \leq r_i \leq p_i), \quad (i = 1, \dots, n) \end{aligned} \tag{7}$$

**Stage 4:** For data clustering aim, first we search the NPM matrix for an element with value 1, and choose it as the representative for the first cluster. Then assume that in the  $L$ th iteration there exist  $N_c$  different clusters, if the new data  $c = (x_1, \dots, x_n)$  is connected to any of the  $N_c$  existing clusters through a non-zero path in the NPM matrix; 'c' will be joined to that cluster. Otherwise if 'c' is not connected to any of the existing clusters' representatives then it must consider as a new cluster's representative.

**Stage 5:** After determining the location and number of clusters, if the prescribed requirements and conditions are met the optimal solution is obtained. Otherwise, we need to return to the stage 3 and increase or decrease  $V_{ref}$  to meet the requirements.

As far as the algorithm is straight forward, the computational complexity of this algorithm could be simply computed. The algorithm could be summarized as filling the NPM matrix. In order to do that each dimension  $i$  is divided regarding its  $dx_i$ . This divides the feature space to relatively small cells. The number of these cells is  $\prod_{i=1}^n p_i$ , as  $p_i$ s defined earlier. Each of the  $N$  data points affect on these cells. The interaction of the potential functions of each data point assigns a potential value to each cell, after which the NPM matrix could be extracted. As a result, the computational complexity of this proposed algorithm is of  $O(N \times \prod_{i=1}^n p_i)$ . This complexity is sound and as far as is no more than polynomial with respect to  $N$  is of great interest.

#### 4.3. Setting $V_{ref}$ automatically

The algorithm's only parameter which has to be set externally is the reference potential. Pal et al. (2000) have used a cluster validation index  $\beta$ , which is the ratio of the total variation and within-cluster variation. This type of measure is widely used as a good criterion for feature selection and cluster analysis (Mitra et al., 2003; Pal et al., 2000).  $\beta$  is defined as:

$$\beta = \frac{\sum_{i=1}^k \sum_{j=1}^{n_i} (X_{ij} - \bar{X})^T (X_{ij} - \bar{X})}{\sum_{i=1}^k \sum_{j=1}^{n_i} (X_{ij} - \bar{X}_i)^T (X_{ij} - \bar{X}_i)} \tag{8}$$

where  $n_i$  is the number of points in the  $i$ th cluster ( $i = 1, 2, \dots, k$ ),  $X_{ij}$  is the feature vector of the  $j$ th pattern in cluster  $i$  ( $j = 1, 2, \dots, n_i$ ),  $\bar{X}_i$  is the mean of  $n_i$  patterns of the  $i$ th cluster,  $n$  is the total number of patterns, and  $\bar{X}$  is the mean value of the entire set of patterns.

This cluster validation index could be taken into use to determine an acceptable value for  $V_{ref}$ . A simple algorithm could do so: set the reference potential to an initial value and then try to optimize that value using a local heuristic search, like simple greedy search or hill climbing to optimize the clustering process. The more the value of  $\beta$ , the better the algorithm performs. For each  $V_{ref}$ , number of clusters is evident and the  $\beta$  index could be calculated. The search algorithm changes the value of  $V_{ref}$  and tries to maximize  $\beta$ .

#### 5. Experimental results

In this section, some examples are illustrated to demonstrate the effectiveness and performance of the proposed algorithm.

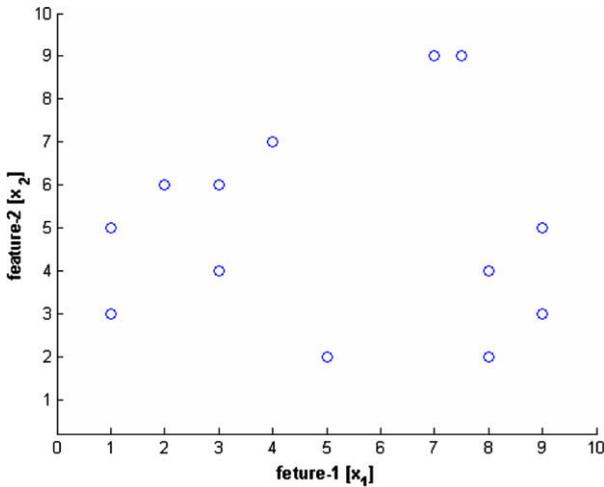


Fig. 4. Diagram of sample artificial data:  $x_{i1} = [2, 3, 4, 5, 7, 8, 1, 3, 1, 9, 8, 9, 7, 7.5]$ ,  $x_{i2} = [6, 4, 7, 2, 9, 4, 3, 6, 5, 5, 2, 3, 9, 9]$ .

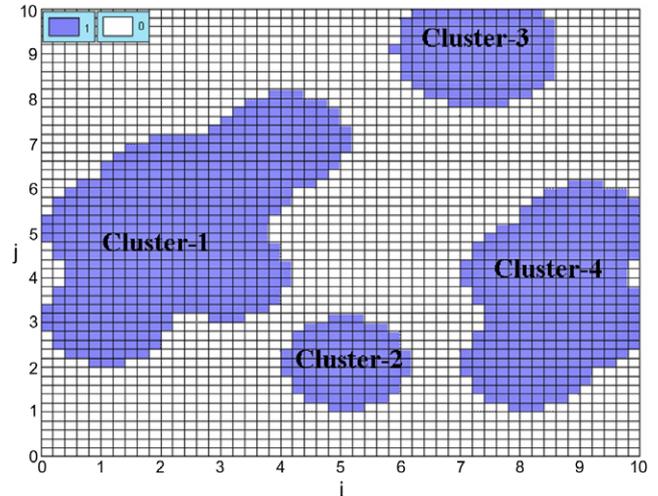


Fig. 6. NPM matrix for  $V_{ref} = 0.04$ , and the resulting clusters.

5.1. Example 1 (artificial data)

Assume that the sample data illustrated in Fig. 4 is given. The potential functions are defined as:

$$V_i(\mathbf{X}) = \exp(-B\|\mathbf{X} - \mathbf{X}_i\|_2^2) = \exp\left(-B \sum_{j=1}^2 (x_j - x_{ij})^2\right),$$

$$\mathbf{X} = (x_1, x_2), \mathbf{X}_i = (x_{i1}, x_{i2}), i = 1, 2, \dots, N = 14$$

From Eq. (4), we can conclude  $B = 4$ . Finally, the potential function for  $A = 5.221$  is as:

$$\tilde{V}(\mathbf{X}) = \frac{5.221}{14} \sum_{j=1}^{14} \exp(-2\|\mathbf{X} - \mathbf{X}_i\|_2^2)$$

As shown in Fig. 5a, average potential function has several different local maximum points in the feature space  $(x_1, x_2)$ . Fig. 5b shows the reference potential surface  $V_{ref} = 0.04$ , together with the potential function itself. The shared boundaries between these two surfaces

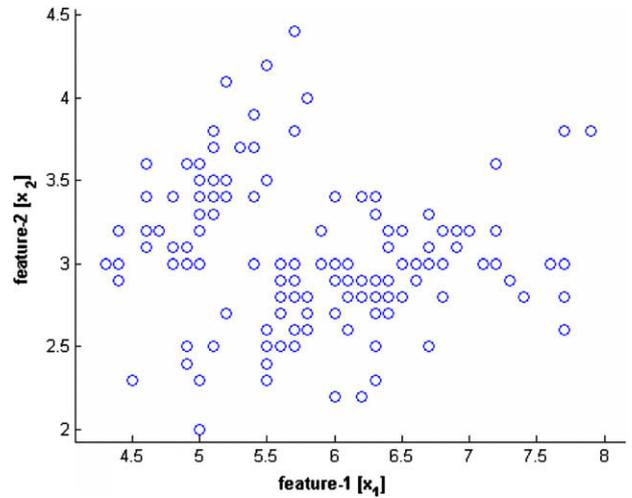


Fig. 7. Iris data dispersion diagram.

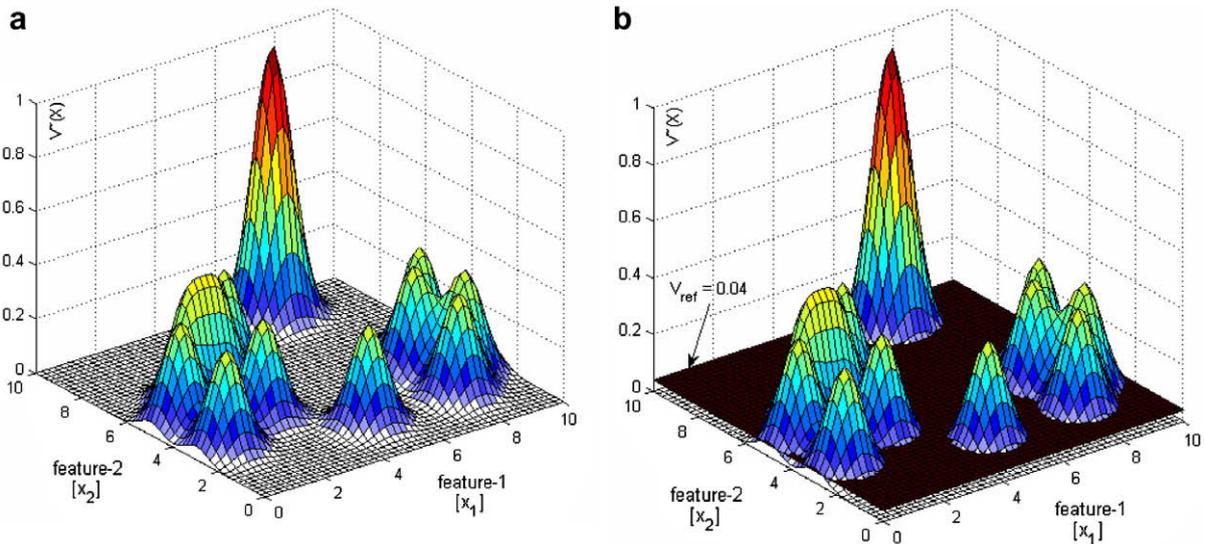


Fig. 5. Graphical representation of (a) normalized average potential function  $\tilde{V}(\mathbf{X})$ ; (b) comparisons of  $\tilde{V}(\mathbf{X})$  and  $V_{ref} = 0.04$ .

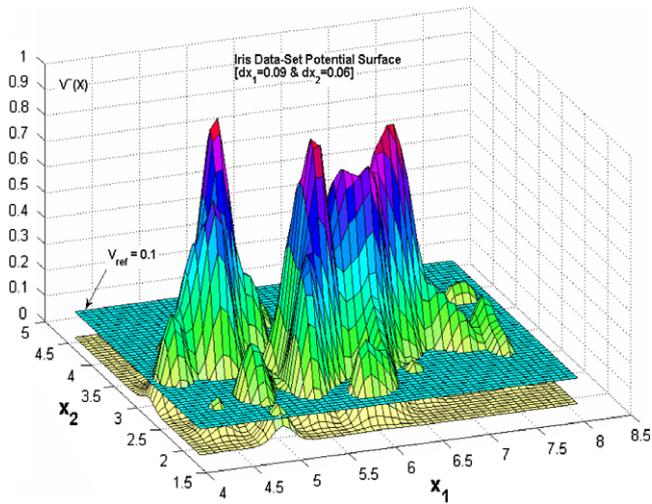


Fig. 8. Comparing normalized average potential function with  $V_{ref} = 0.1$ .

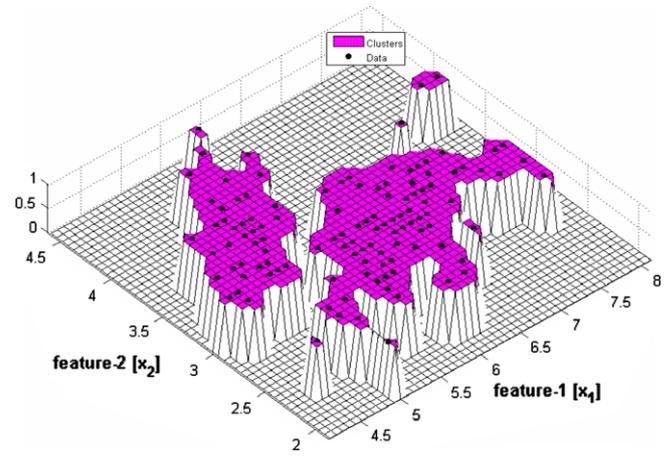


Fig. 9. Cluster areas for  $V_{ref} = 0.1$ .

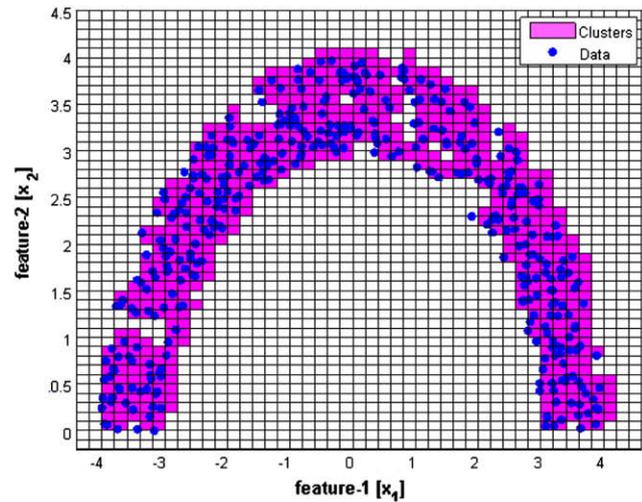
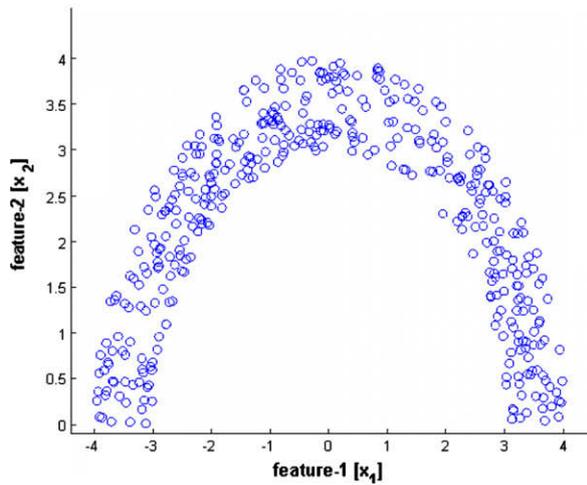


Fig. 10. (Arc cluster) Result of the clustering algorithm and representation of the generated cluster area for  $V_{ref} = 0.1$ .

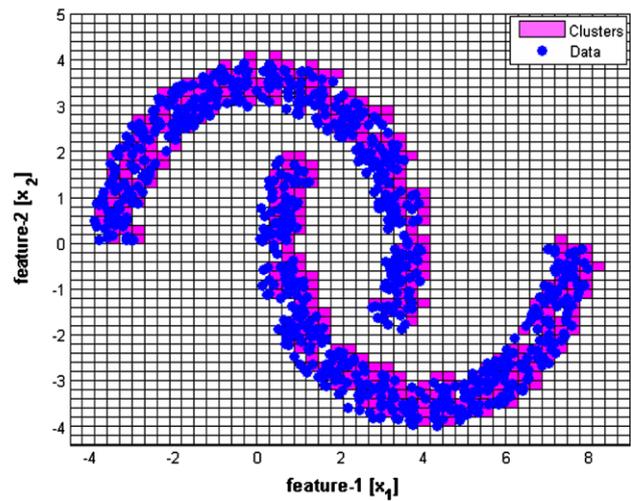
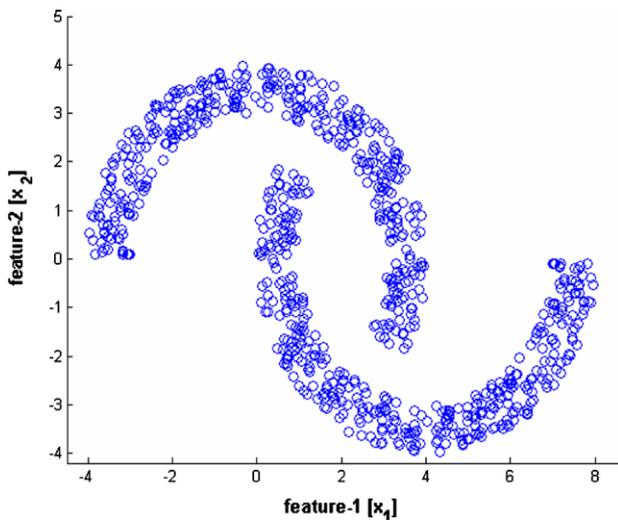


Fig. 11. (Two complex arc clusters) Result of applying the clustering algorithm for  $V_{ref} = 0.1$ .

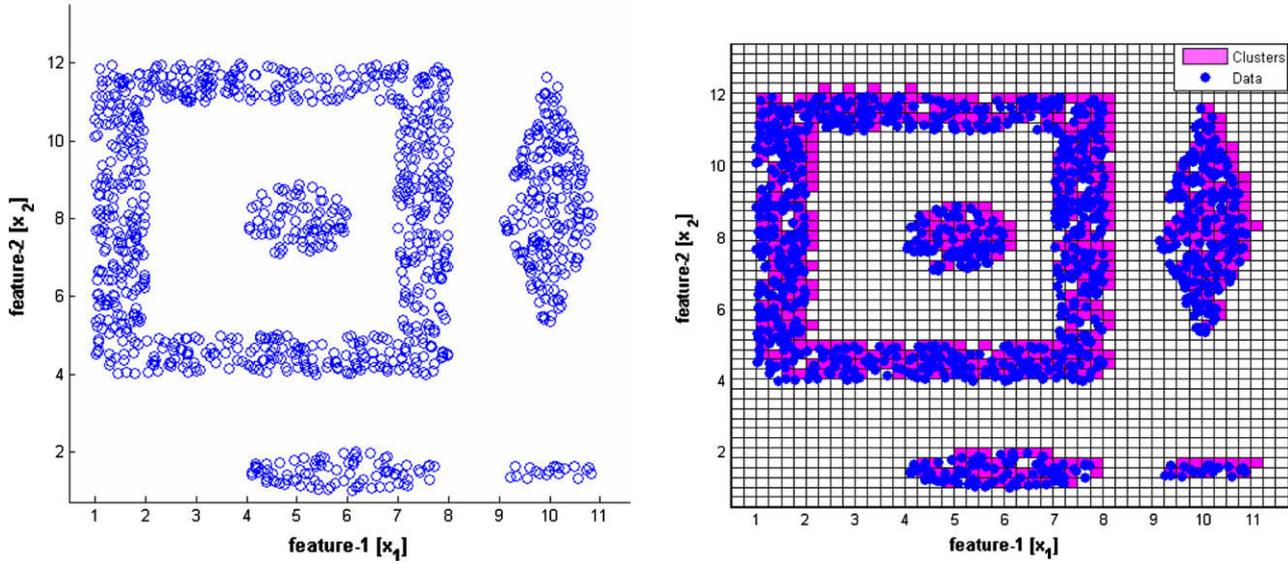


Fig. 12. (Geometric clusters) Result of applying the clustering algorithm and representation of the generated cluster area for  $V_{ref} = 0.05$ .

are, in fact, the equipotential levels and the inner area of equipotential boundaries are the generated clusters for the considered reference potential. For the next stage, choosing  $dx_1 = dx_2 = 0.2$  and comparing  $V_{ref} = 0.04$  and  $\tilde{V}(\mathbf{X})$  yields to extracting the NPM matrix from Eq. (6), as illustrated in Fig. 6. Finally, the data is clustered as:

- Cluster\_1** =  $\{(1, 3), (1, 5), (2, 6), (3, 4), (3, 6), (4, 7)\}$
- Cluster\_2** =  $\{(5, 2)\}$ , **Cluster\_3** =  $\{(7, 9), (7.5, 9)\}$
- Cluster\_4** =  $\{(8, 2), (8, 4), (9, 3), (9, 5)\}$

If the acquired results do not satisfy the problem requirements, we do the same as stage 5.

5.2. Example 2 (real Iris data with two features)

In this example, we use the Iris dataset which is frequently used in many papers to illustrate the performance of clustering/classification algorithms. Fig. 7 shows the diagram of 150 Iris data samples with two features.

The potential function is considered as explained in Eqs. (2) and (3), and the coefficient  $B$  is determined as  $B = 41$  using Eq. (4). Finally, the average potential functions for  $A = 18.423$  are as:

$$\begin{aligned} \tilde{V}(\mathbf{X}) &= \frac{A}{N} \sum_{j=1}^N \exp(-41 \|\mathbf{X} - \mathbf{X}_j\|_2^2) \\ &= \frac{18.423}{150} \sum_{j=1}^{150} \exp\left(-41 \sum_{j=1}^2 (x_j - x_{ij})^2\right) \end{aligned}$$

After some repetitions,  $V_{ref} = 0.1$  is assigned.

The reference potential plane  $V_{ref} = 0.1$  and the average potential function are shown in Fig. 8.

After that by choosing  $dx_1 = 0.09$ ,  $dx_2 = 0.06$  and comparing  $V_{ref} = 0.1$  and  $\tilde{V}(\mathbf{X})$  with regard to Eq. (6), the cluster areas are extracted as illustrated in Fig. 9.

5.3. More non-convex cases

Now it is time to test the algorithm with more difficult and highly non-convex data. These data are particularly good performance measures for clustering algorithms evaluations.

Applying different clustering algorithms to these data sets usually yields to different results. The results of applying the proposed

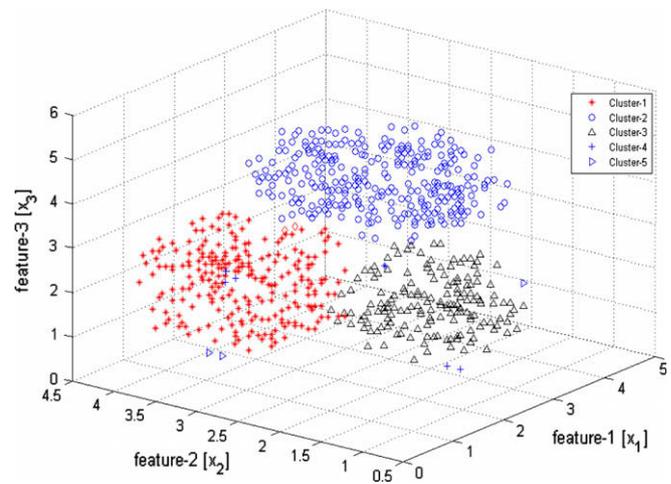


Fig. 13. (3D clusters) Result of applying the clustering algorithm and representation of the generated cluster area for  $V_{ref} = 0.05$ .

clustering algorithm are illustrated in Figs. 10–13. The resulting outcomes can prove the reliable, robust and optimal performance of this algorithm especially in non-convex cases.

6. Conclusion and future works

Unsupervised classification algorithms are generally based on similarity measure of the feature vector; therefore the elements forming a cluster are the most similar ones among the others. On this basis and relying on the potential functions and equipotential surfaces, an optimal clustering algorithm is presented that can be also used for non-convex datasets or datasets with any kind of dispersion. In this method a potential function is assigned to any of the data samples, and then interacting the fields in the feature space extracts the equipotential surfaces. Using these equipotential surfaces and the proposed algorithm the clustering process could be accomplished. Taking advantage of these equipotential boundaries, as discussed before, leads to finding the optimal clustering solution. This algorithm is not dependent to any external parameter setting.

The experimental results illustrate the desirable, robust performance of the proposed method in clustering fragmented, complex, non-convex data with different shape, size and densities. On the other hand, this method has a reasonable computational complexity comparing with other clustering methods. The main reason of the computational complexity reduction is figuring the total potential function out once at the beginning and using in it the subsequent stages with no need of recalculations. This later feature is more noticeable with increasing the number of data samples and the dimensions of the feature vector.

For the future work, in order to decrease the execution time of the algorithm some heuristics could be taken into account. Using non-uniform dividing of the feature space could be one. Instead of using a same  $dx$  for a dimension in the feature space, cells with different sizes could be used. The distribution of the data points can affect the choice of cell sizes. Or a simple pre-refinement phase can be added to refine the data points, and for example merge very close data points to a single representative with a bit of higher impacts in the potential function..

## References

- Anderberg, M. R. (1973). *Cluster analysis for applications*. New York: Academic Press Inc..
- Bayat, F., Jalali, A. A., & Jahed Motlag, M. R. (2009). Utilizing multi-parametric programming for optimal control of linear constrained systems. *Iranian Journal of Control*, 3(1), 12–22.
- Dubes, R. C. et al. (1976). Clustering techniques: The user's dilemma. *Pattern Recognition*, 8, 247–260.
- Fisher, L. et al. (1971). Admissible clustering procedures. *Biometrika*, 58, 91–104.
- Gan, W. Y. (2003). *Study on clustering problem for data mining foundations*. PhD thesis, University of Science and Technology, Nanjing.
- Gan, W. Y. et al. (2003). Optimal choice of parameters for a density-based clustering algorithm. In *Proceedings of the 19th international conference on rough sets, fuzzy sets, data mining* (pp. 603–606).
- Gan, G., Wu, J., & Yang, Z. (2009). A genetic fuzzy  $k$ -modes algorithm for clustering categorical data. *Expert Systems with Applications*, 36, 1615–1620.
- George, K. et al. (1999). Chameleon: A hierarchical clustering algorithm using dynamic modeling. *IEEE Computer*, 27(3), 229–241.
- Hsu, C.-C., & Huang, Y.-P. (2008). Incremental clustering of mixed data based on distance hierarchy. *Expert Systems with Applications*, 35, 1177–1185.
- Jain, A. K. et al. (1996). Image segmentation using clustering. *Advances in image understanding: A Festschrift for Azriel Rosenfeld*. Piscataway, NJ: IEEE Press (pp. 65–83).
- Mitra, P., Pal, S. K., & Siddiqi, M. A. (2003). Non-convex clustering using expectation maximization algorithm with rough set initialization. *Pattern Recognition Letters*, 24, 863–873.
- Oyang, Y. J. et al. (2001). A study on hierarchical data clustering algorithm based on gravity theory. In *Proceeding of fifth European conference on principles and practice of knowledge* (pp. 350–361).
- Pal, S. K., Ghosh, A., & Uma Shankar, B. (2000). Segmentation of remotely sensed images with fuzzy thresholding, and quantitative evaluation. *International Journal of Remote Sensing*, 21(11), 2269–2300.
- Park, H.-S., & Jun, C.-H. (2009). A simple and fast algorithm for  $k$ -medoids clustering. *Expert Systems with Applications*, 36, 3336–3341.
- Rasmussen, E. (1992). *Clustering algorithms information retrieval: Data structures and algorithms* (pp. 419–442). Upper Saddle River, NJ: Prentice-Hall Inc.
- Salton, G. (1991). Developments in automatic text retrieval. *Science*, 253, 974–980.
- Titterton, D. M. et al. (1985). *Statistical analysis of finite mixture distributions*. New York: John Wiley & Sons Inc..
- Wang, M.-H. et al. (2009). A novel clustering algorithm based on the extension theory and genetic algorithm. *Expert Systems with Applications*, 36(4), 8269–8276.
- Wei, C.-P., Lee, Y.-H., & Hsu, C.-M. (2003). Empirical comparison of fast partitioning-based clustering algorithms for large data sets. *Expert Systems with Applications*, 24, 351–363.
- Wu, J. et al. (2009). External validation measures for  $k$ -means clustering: A data distribution perspective. *Expert Systems with Applications*, 36(3P2), 6050–6061.