

Fuzzy Clustering Using Automatic Particle Swarm Optimization

Min Chen and Simone A. Ludwig

Department of Computer Science

North Dakota State University

Fargo, ND, USA

min.chen@my.ndsu.edu, simone.ludwig@ndsu.edu

Abstract—Fuzzy clustering is a popular unsupervised learning method used in cluster analysis which allows a data point to belong to two or more clusters. Fuzzy c-means is one of the most well-known and used methods, however, the number of clusters need to be defined in advance. This paper proposes a clustering approach based on Particle Swarm Optimization. This approach automatically determines the optimal number of clusters using a threshold vector that is added to the particle. The algorithm starts by partitioning the data set randomly within a preset maximum number of clusters in order to overcome the fuzzy c-means shortcoming of the predefined cluster count. A reconstruction criterion is applied to evaluate the performance of the clustering results of the proposed algorithm. The experiments conducted show that the proposed algorithm can automatically find the optimal number of clusters.

Index Terms—Fuzzy Clustering, Particle Swarm Optimization (PSO), data mining

I. INTRODUCTION

Data mining is called exploratory data analysis, among other things. It is an analytic process designed to explore data. Data mining aims to search for consistent patterns or systematic relationships between variables. It then validates the findings by applying the detected patterns to new subsets of data [1]. It is a statistical analysis process which can identify the clusters along with collection of data. Data mining can be achieved by classification, association, prediction, sequential pattern, similar time sequences and clustering [2].

Clustering analysis is one of the popular approaches and has been widely used in data mining. Clustering analysis is a process to identify groups or clusters based on some similarity measures. Most clustering algorithms can be categorized into two popular techniques known as hierarchical and partitional clustering. The output of the hierarchical clustering is a tree showing a sequence of clusters with each cluster being a partition of the data set. Hierarchical clustering does not specify the number of clusters, and the output is independent of the initial condition. However, the hierarchical clustering is static, i.e., the data points assigned to a cluster cannot be reassigned to another cluster. In addition, it will fail to separate overlapping clusters due to the lack of information regarding the global shape or size of the clusters. On the other hand, partitioned clustering requires a fixed number of clusters to be specified a priori. Objective functions such as square error function are used as a criteria in the optimization process

of data partitioning. Partitioned clustering uses an iterative process to optimize the cluster centers, as well as the number of clusters. However, it is a challenge to find the “optimum” number of clusters since it always requires prior knowledge about the data. The advantages of hierarchical algorithms are the disadvantages of the partitional algorithms and vice versa.

The goal of clustering involves the task of dividing data points into homogeneous groups such that the data points in the same group are as similar as possible and data points in different groups are as dissimilar as possible [3], [2]. The importance of clustering is documented in pattern recognition [4], machine learning, image analysis [5], information retrieval, etc. Depending on whether a data point belongs to a single cluster or several clusters with different membership degrees, clustering methods can be categorized as either hard clustering [6], [7] or fuzzy clustering [8]. Each data point of the data set belongs to exactly one cluster in hard clustering. Fuzzy set theory which was proposed by Zadeh [9] in 1965 is used to describe the membership degrees in fuzzy cluster analysis. Therefore, each data point of the data set belongs to two or more clusters with a membership degree between 0 and 1. Due to the capacity of handling uncertainty and vagueness, the potential of fuzzy clustering to reveal the underlying structures in data with regard to similarities or dissimilarities among them can be exploited [10].

One of the widely used methods in fuzzy clustering is Fuzzy C-Means clustering (FCM) [11]. The FCM method attempts to partition a data set into a collection of c fuzzy groups. It finds a cluster center in each group such that the intra-distance within the group is minimized and the inter-distance between each group is maximized. All of the fuzzy clustering methods that have been applied recently mostly use an extension of the FCM algorithm. As we have discussed before, partitional clustering suffers from the following two drawbacks:

- The number of clusters needs to be specified in advance. Furthermore, it requires prior knowledge or ground truth of the data.
- In most cases data points in overlapping areas can not be categorized correctly.

In order to overcome these two drawbacks, we proposed a fuzzy C-means clustering approach using automated Particle Swarm Optimization (PSO) for clustering analysis. The rest of

this paper is organized as follows. In Section II, fuzzy c-means and PSO are introduced. The proposed algorithm is described in Section III. A list of validity indices is given in Section IV. The experimental results and analysis is described in Section V. We finally conclude this paper in Section VI.

II. RELATED WORK

FCM was first developed by [12] in 1973 and was extended by [11] in 1981. Since then, FCM is one of the most promising fuzzy clustering methods. Many variants of FCM have been introduced. For example, the Gustafson-Kessel (GK) algorithm [7] is a fuzzy clustering technique which can estimate local covariance to partition data into subsets that can be well fitted with linear sub-models. However, considering a general structure of the covariance matrix can have substantial effect on the modeling approach, thus the Gath-Geva algorithm [13] was proposed. The fuzzy c-varieties (FCV) [14] clustering algorithm is a fuzzy clustering algorithm where the prototype of each cluster is a multi-dimensional linear vector. It is similar to cluster analysis, however, it uses the statistical method of principal component analysis. A generalized FCM algorithm is presented in [15]. The algorithm proposes an approach for setting the algorithm parameters.

With regards to PSO approaches, two methods called PSO-V and PSOU are introduced in [16]. A reformulated objective function of fuzzy c-means is minimized by PSO for cluster analysis. A PSO-based fuzzy clustering algorithm is introduced to overcome the shortcomings of FCM [17]. An ant colony clustering algorithm is applied for solving clustering problem in [18]. The algorithm employs the global pheromone updating and the heuristic information to find clustering solutions. In [19], a genetic fuzzy K-modes algorithm for clustering categorical data is proposed. The fuzzy K-modes clustering is treated as an optimization problem and a genetic algorithm is used to obtain the global optimal solution. A hybrid data clustering algorithm which makes use of the merits of PSO and KHM is proposed in [20]. The proposed algorithm helps the KHM to escape from local optima. In addition, it overcomes the problem of slow convergence of the PSO algorithm. A hybrid evolutionary algorithm called FAPSO-ACO-K is discussed in [21]. The hybrid algorithm is based on PSO, ACO and k-means applied to cluster analysis. A new method for dynamic parameter adaptation in PSO is proposed in [22]. The proposed algorithm uses fuzzy logic to improve the convergence and diversity of the swarm in PSO.

Although PSO techniques do eventually locate the desired solution, the high computational cost and the slow convergence rate severely limit the use of PSO on clustering analysis. For these reasons, a chaotic map PSO with an accelerated convergence rate strategy is introduced in [23]. The algorithm adopts chaotic maps and adaptive action to avoid entrapment of the PSO in a local optimum. A hybrid fuzzy clustering method based on FCM and FPSO is proposed to overcome the shortcomings of PSO in [24]. A modified version of PSO, known as Multi-Elitist PSO (MEPSO), is proposed in [25]. This approach solves the hard clustering problem which can

automatically determine the optimal number of clusters. This approach shows that PSO is guaranteed to solve clustering problem automatically.

III. FUZZY C-MEANS AND PARTICLE SWARM OPTIMIZATION

A. Fuzzy C-means Clustering

Fuzzy clustering is a method of clustering which allows one piece of data to belong to two or more clusters. The FCM algorithm is an iterative partition clustering technique which was first introduced by Dunn [12] and was extended by Bezdek [11]. FCM is a pretty standard least squared error model that generalizes an earlier and very popular non-fuzzy c-means model that produces hard clusters of the data. An optimal c partition is produced iteratively by minimizing the weighted within group sum of squared error objective function:

$$J = \sum_{i=1}^n \sum_{j=1}^c (u_{ij})^m d^2(y_i, c_j) \quad (1)$$

where $Y = [y_1, y_2, \dots, y_n]$ is the data set in a d -dimensional vector space. n is the number of data items. c is the number of clusters which is defined by the user where $2 \leq c \leq n$. u_{ij} is the degree of membership of y_i in the j^{th} cluster. m is a weighted exponent on each fuzzy membership. c_j is the center of cluster j . $d^2(x_i, c_j)$ is a square distance measure between object y_i and cluster c_j . An optimal solution with c partitions can be obtained via an iterative process which is as follows:

- 1) Input($c, m, \epsilon, \text{data}$)
- 2) Initialize the fuzzy partition matrix $U = [u_{ij}]$
- 3) Iteration starts and set $t=1$
- 4) Calculate the c cluster centers with U^t :

$$c_i = \frac{\sum_{i=1}^n (u_{ij})^m y_i}{\sum_{i=1}^n (u_{ij})^m} \quad (2)$$

- 5) Calculate the membership U^{t+1} using:

$$u_{ij} = \frac{1}{\sum_{k=1}^c \left(\frac{d_{ij}}{d_{kj}} \right)^{\frac{2}{m-1}}} \quad (3)$$

- 6) If the stopping criteria is not met, $t = t + 1$ and go to Step 4)

B. Particle Swarm Optimization

PSO was originally designed and introduced by Eberhart and Kennedy [26]. The PSO is a population search algorithm which intends to simulate the choreography of a bird flock. Each individual, called particle, within the swarm is represented by a vector in a multidimensional search space. A velocity vector is assigned to each particle to determine the next movement of the particle. Each particle updates its velocity based on the current velocity, best personal position it has explored so far and the global best position explored by the swarm:

The velocity and position of the particle at the next iteration is updated as:

$$V_i(t+1) = wV_i(t) + c_1r_1(X_i^l(t) - X_i(t)) + c_2r_2(X^g - X_i(t)) \quad (4)$$

$$X_i(t+1) = X_i(t) + V_i(t+1) \quad (5)$$

for the i^{th} particle, where w is the inertia weight, $V_i(t)$ is the previous velocity in iteration t of i^{th} particle. c_1 and c_2 are coefficients. Generally, r_1 and r_2 are random numbers between 0 and 1. $(X_i^l(t) - X_i(t))$ is the difference between the local best X_i^l of the i^{th} particle and the previous position $X_i(t)$. Similarly, $(X^g - X_i(t))$ is the difference between the global best X^g and the previous position $X_i(t)$.

IV. PROPOSED APPROACH

The proposed algorithm is based on PSO and FCM. The particle encoding, velocity encoding, decoding and clustering validation is described separately. The procedures of the proposed algorithm are presented at the end of this section.

A. Particle Encoding

A particle is a $2 \times k$ matrix, where k is the maximum number of clusters that is predefined. The first row represents the centers. Each value in the second row controls the activation of each center in the first row.

$$X_i = \begin{pmatrix} x_{1,1}^i & x_{1,2}^i & \dots & x_{1,k}^i \\ t_{2,1}^i & t_{2,2}^i & \dots & t_{2,k}^i \end{pmatrix} \quad (6)$$

where $x_{1,k}^i$ represents the i^{th} particle's position in cluster k . $x_{1,k}^i$ should be in the range of $[x_{min}, x_{max}]$. $t_{2,k}^i$ is the i^{th} particle's threshold value in the range of $[0, 1]$. If the threshold value is greater than 0.5, the center is activated. Otherwise, it is deactivated.

B. Velocity Encoding

The velocity matrix should have the same dimension as the position matrix with a range. Suppose we set the range as $[v_{min}, v_{max}]$, all values of the velocity matrix should be between v_{min} and v_{max} . Thus, the i^{th} velocity is denoted as:

$$V_i = \begin{pmatrix} v_{x1,1}^i & v_{x1,2}^i & \dots & v_{x1,k}^i \\ v_{t2,1}^i & v_{t2,2}^i & \dots & v_{t2,k}^i \end{pmatrix} \quad (7)$$

Similarly, k is the maximum number of clusters. The first row is the velocity of the centers, and the second row is the velocity of the threshold values.

C. Decoding

$Y = (y_1, y_2, \dots, y_n)$ is the data set with d dimensions. The cluster centers can be decoded as $C = (c_1, c_2, \dots, c_k)$ using Equation 2.

D. Clustering Validation Techniques

The aim of clustering validation is to evaluate the clustering results to find the best partition that fits the underlying data. Thus, cluster validity is used to quantitatively evaluate the result of clustering algorithms. Compactness and separation are considered as two widely used criteria in measuring the quality of partitioning a data set into a number of clusters. Conventional approaches run the algorithm iteratively using different input values and select the best validity measure to determine the "optimum" number of clusters. A collection of validity indices in fuzzy clustering is listed below.

1) *Least Squared Error (SE) Index*: The weighted within cluster sum of squared error function is used [27]:

$$J_m = \sum_{i=1}^n \sum_{j=1}^c u_{ij}^m \|y_i - c_j\|^2 \quad (8)$$

where y_i is the i^{th} data point with d dimensions. c_j is the value of the j^{th} cluster, and $\|y_i - c_j\|$ is the Euclidean distance between y_i and c_j . J_m takes its minimum value when the cluster structure is best.

2) *Partition Coefficient (PC) Index*: The partition coefficient (PC) is defined as [11]:

$$PC = \frac{1}{n} \sum_{i=1}^n \sum_{j=1}^c u_{ij}^2 \quad (9)$$

PC obtains its maximum value when the cluster structure is optimal.

3) *Partition Entropy (PE) Index*: The partition entropy was defined as [14]:

$$PE = -\frac{1}{n} \sum_{i=1}^n \sum_{j=1}^c u_{ij} \log_b(u_{ij}) \quad (10)$$

where b is the logarithmic base. PE gets its minimum value when the cluster structure is optimal.

4) *Modified Partition Coefficient (MPC) Index*: Modification of the PC index, which can reduce the monotonic tendency, is proposed by Dave in 1996 [28].

$$MPC = 1 - \frac{c}{c-1}(1 - PC) \quad (11)$$

where c is the number of cluster. An optimal cluster number is found by maximizing MPC to produce a best clustering performance for a data set.

5) *Fukuyama and Sugeno (FS) Index*: Fukuyama and Sugeno proposed a validity function in 1989 [29]. It is defined as:

$$FS = \sum_{i=1}^n \sum_{j=1}^c \mu_{ij}^m \|x_i - c_j\| - \sum_{i=1}^n \sum_{j=1}^c \mu_{ij}^m \|c_j - \bar{c}\| \quad (12)$$

where $\bar{c} = \sum_{j=1}^c c_j / c$. It measures the separation. The first term equals to J_m which is the least squared error. It measures the compactness. The best clustering performance for a data set is found by maximizing the value of FS.

6) *Xie-Beni (XB) Index*: Xie and Beni proposed a validity function in 1991 [30], and later it was modified by Bezdek in 1995 [31].

$$XB = \frac{J_m}{n \times \min_{i \neq j} \|z_i - z_j\|^2} \quad (13)$$

XB reaches its minimum value when the cluster structure is optimal.

7) *Partition Coefficient and Exponential Separation (PCAES) Index*: The partition coefficient and exponential separation (PCAES) index [32] is defined as:

$$PCAES = \sum_{i=1}^n \sum_{j=1}^c \frac{(u_{ij})^2}{u_M} - \sum_{k=1}^c \exp(-\min_{k \neq i} \|z_i - z_k\|^2 / \beta_T) \quad (14)$$

where $u_M = \min_{1 \leq j \leq c} \{\sum_{i=1}^n u_{ij}^2\}$ and $\beta_T = (\sum_{j=1}^c \|z_j - \bar{z}\|^2) / c$. $\bar{z} = \sum_{i=1}^n (y_i / n)$. PCAES takes its maximum value when the cluster structure is optimal.

8) *Weighted Inter-Intra (Wint) Index*: The weighted inter-intra (Wint) measure is introduced by Strehl [33] in 2002. It compares the compactness of the data to its separation.

$$Wint = (1 - \frac{2c}{n}) \cdot (1 - \frac{\sum_i \frac{1}{n-|c_i|} \sum_{j \neq i} inter(c_i, c_j)}{\sum_i \frac{2}{|c_i|-1} intra(c_i)}) \quad (15)$$

where $intra(c_i)$ is the average intra-distance within cluster i . $inter(c_i, c_j)$ is the average inter-distance between cluster i and cluster j . Wint obtains its maximum value when the cluster structure is optimal.

The procedure of the proposed algorithm is listed below:

Input: data set $Y = [y_1, y_2, \dots, y_n]$, number of cluster c , fuzzification coefficient m .

Output: a $n \times c$ partition matrix U and corresponding centers.

- 1) Randomly initialize a swarm
- 2) Iteration starts and set $t=1$
- 3) Update the velocity of each particle using Equation 4
- 4) Update the position of each particle using Equation 5
- 5) Update the personal best and global best
- 6) Calculate the partition matrix U
- 7) If the stopping criterion is not met, $t = t + 1$ and go back to Step 3)
- 8) The partition matrix U of the global best is used to reconstruct the original data
- 9) Calculate the reconstruction error. In order to use a consistent method to evaluate the eight different indices, the reconstruction criterion (RC) [34] is used. The reconstruction criterion uses the cluster prototypes and partition matrix to “reconstruct” the original data vectors. The reconstructed version of the original data vectors, $\hat{Y} = [\hat{y}_1, \hat{y}_2, \dots, \hat{y}_n]$, is calculated as:

$$\hat{y}_i = \frac{\sum_{j=1}^c u_{ij}^m c_j}{\sum_{j=1}^c u_{ij}^m} \quad (16)$$

Once the reconstruction has been finished, the squared error of the reconstruction vectors and original vectors are evaluated using Equation 17.

$$E = \sum_{i=1}^n \|\hat{y}_i - y_i\|^2 \quad (17)$$

- 10) Select the partition matrix and centers corresponding to the minimum reconstruction error.

V. EXPERIMENTS AND RESULTS

In this section, the experimental setup, datasets and experimental study are described.

A. Experimental Setup

The experiments are implemented and evaluated on an ASUS desktop (Intel(R) Dual Core I3 CPU @3.07 GHz, 3.07 GHz) Matlab Version 7.13. All measurements of the proposed algorithm are tested 30 times. The parameters required for the proposed algorithm are listed in Table I.

Parameter	Value
Maximum number of cluster	10
Maximum iteration	50
Swarm size	25
Maximum run	30
Fuzzification coefficient (m)	2

TABLE I: Parameters and their values of the proposed algorithm.

B. Datasets

The experiments are conducted on a number of datasets taken from the UCI repository [35], and one synthetic data set was generated in Matlab. The datasets are described in Table II.

TABLE II: Datasets used for the experiments.

Data Set	Dimensions	Instances	Classes
Pinwheel	2	1000	2
Transfusion	4	748	2
Haberman	3	306	2
Breast-W	9	699	2
Jain	2	373	2
Thyroid	5	215	2
Iris	4	150	3

C. Experimental Study

1) *Use of Synthetic Data*: In order to investigate the clustering performance with different numbers of clusters, we use a synthetic data set to test the clustering performance using K-means, FCM and the proposed algorithm (FPSO). K-means and FCM are the default methods in Matlab. The data generated is shown in Figure 1. A toolbox for estimating the number of clusters implemented by Kaijun Wang is adopted. Nine traditional indices, Rand [36], Mirkin [37], Hubert [38], Silhouette [39], Davies-Bouldin [40], Calinski-Harabasz [41], Krzanowski-Lai [42], Hartigan [43] and Homogeneity-Separation [44] are implemented as used by Wang.

Figure 2 lists the performance of the synthetic data set using K-means. The Rand index measures an agreement of the clustering results. As the number of clusters increases, the

agreement of the clustering results decrease. As the number of clusters increases, the performance using Mirkin distance increases. Hubert index has the same performance as Rand index. As the number of clusters increases, the performance using Silhouette improves. The optimal cluster numbers found by Davies-Bouldin, Calinski-Harabasz, Krzanowski-Lai, Hartigan and Homogeneity-Separation are 2, 6, 9, 1, and 2, respectively.

Figure 3 shows the performance results using FCM on the same synthetic data set. As shown in the figure, Rand, Mirkin, Hubert and Silhouette have the same performance as K-means. The optimal number of clusters found by Davies-Bouldin, Calinski-Harabasz, Krzanowski-Lai, Hartigan and Homogeneity-Separation are 9, 7, 4, 1, and 2, respectively.

Due to the stochastic nature of PSO, the performance of the proposed algorithm fluctuates using the Rand, Mirkin, Hubert and Silhouette indices as seen in Figure 4. The optimal number of clusters found by Davies-Bouldin, Calinski-Harabasz, Krzanowski-Lai, Hartigan and Homogeneity-Separation are 9, 7, 9, 1, and 2, respectively.

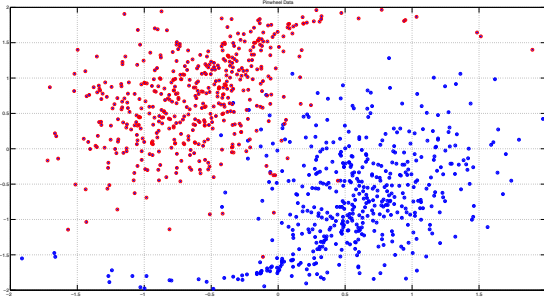


Fig. 1: Synthetic data set

2) *Use of Real-World Data:* In this section, we investigate the behavior of the clustering results using eight different validity measures listed in Equations 8-15.

In Table III, the reconstruction errors of the transfusion data set, where c ranges from 2 to 9, have been calculated using the proposed algorithm by applying Equations 8-15. As shown by the results, the bold values are minimum reconstruction errors with different cluster numbers for each measure. 6 out of 8 cases show that $c = 2$ is the optimal number of clusters.

TABLE III: Reconstruction error with varying c using transfusion data set.

	$c=2$	3	4	5	6	7	8	9
SE	23.6	25.6	26.5	23.6	28.7	33.7	24.3	25.9
PC	16.3	29.6	34.2	81.8	22.7	75.1	31.1	21.4
PE	26.3	85.0	26.0	57.0	24.9	39.3	16.9	30.3
MPC	17.0	44.8	24.6	60.4	93.3	12.2	166.7	12.5
FS	23.6	23.6	23.6	25.6	29.5	24.9	33.6	23.7
XB	23.6	43.0	25.1	36.1	27.1	61.3	68.8	63.1
PCAES	38.9	81.7	82.5	48.1	53.8	57.5	97.2	93.4
Wint	23.6	29.3	25.4	47.4	51.5	68.8	42.5	36.5

Due to the stochastic nature of our proposed algorithm, we tested the proposed algorithm on 30 runs and calculated the

average optimal number of clusters as listed in Table IV. The values in parenthesis are the standard deviation. The results show that the proposed algorithm can obtain an optimal cluster number except for the Breast data set which returns $c = 3$ using the MPC index.

TABLE IV: Eight different indices using the proposed algorithm.

	Haberman	Breast	Jain	Thyroid	Iris
SE	2.17 (0.44)	2.27 (0.56)	2.37 (0.60)	2.20 (0.54)	3.20 (1.15)
PC	2.30 (0.76)	2.40 (0.89)	2.30 (0.86)	2.53 (0.95)	3.50 (1.57)
PE	2.33 (0.51)	2.43 (0.57)	2.40 (0.55)	2.50 (0.61)	3.83 (1.72)
MPC	2.43 (0.89)	3.61 (1.13)	2.73 (1.18)	2.97 (1.31)	3.33 (1.55)
FS	2.23 (0.46)	2.30 (0.49)	2.23 (0.46)	2.47 (0.59)	3.23 (1.20)
XB	2.40 (0.55)	2.47 (0.59)	2.33 (0.51)	2.57 (0.66)	3.47 (1.40)
PCAES	2.47 (0.59)	2.53 (0.64)	2.27 (0.48)	2.70 (0.76)	3.43 (1.37)
Wint	2.33 (0.51)	2.47 (0.59)	2.30 (0.49)	2.40 (0.55)	3.33 (1.28)
True Class labels	2	2	2	2	3

VI. CONCLUSION

In this study, we proposed an algorithm based on PSO and FCM in order to overcome the drawbacks of traditional partition clustering. The proposed algorithm uses a threshold vector to control the number of clusters and solves the clustering problem via an iterative fuzzy partition process. We generated a synthetic dataset and used 6 datasets from the UCI repository. The results show that the proposed algorithm can automatically find the optimal number of clusters.

However, due to the slow convergence and the stochastic nature of the PSO algorithm, the prediction results of a single run fluctuates and is thus hard to make predictions. Unlike K-means and FCM, the proposed algorithm need to be tested repeatedly in order to find the optimal solution. In addition, the maximum number of clusters is predefined, and the recursive increase of the number of clusters is computational expensive.

As for future work, it would be interesting to improve the proposed algorithm to achieve more stable predictions with fewer runs. Moreover, we are planning to explore the proposed algorithm with big datasets, and therefore parallelization techniques are necessary.

REFERENCES

- [1] H. A. Edelstein, "Introduction to data mining and knowledge discovery (3rd ed)," Potomac, MD: Two Crows Corp. 1999.
- [2] B. Mirkin, "Clustering: A Data Recovery Approach," Second Edition (Chapman & Hall/CRC Computer Science & Data Analysis).
- [3] R. B. Cattell, "The description of personality: Basic traits resolved into clusters," Journal of Abnormal and Social Psychology, 38, 476-506, 1943.
- [4] S. Theodoridis, and K. Koutroubas, "Pattern Recognition," Academic Press, 1999.
- [5] T. N. Pappas, "An adaptive clustering algorithm for image segmentation," IEEE Trans. Signal process, vol. 40, pp.901-914, 1992
- [6] A. Likas, N. Vlassis, and J. Verbeek, "The global k-means clustering algorithm (Technical Report)," Computer Science Institute, University of Amsterdam, The Netherlands. ISA-UVA-01-02. 2001.

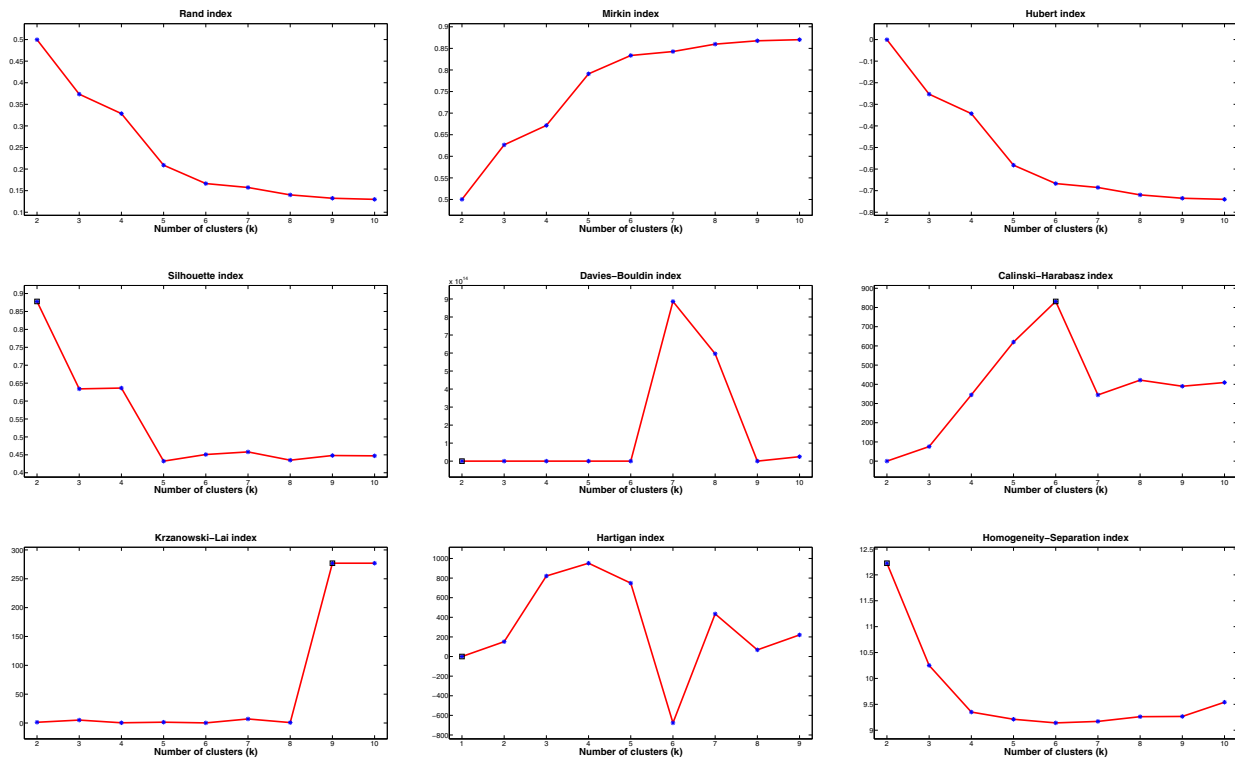


Fig. 2: Kmeans using 9 different validity indices

- [7] V. P. Guerrero-Bote, et al., "Comparison of neural models for document clustering," *Int. Journal of Approximate Reasoning*, vol. 34, pp.287-305, 2003.
- [8] G. L. Carl, "A fuzzy clustering and fuzzy merging algorithm, Technical Report," CS-UNR-101, 1999.
- [9] L. A. Zadeh, *Fuzzy sets, Information and Control*, Vol. 8, pp. 338-353, 1965.
- [10] R. Babuska, "Fuzzy Modelling for Control," Kluwer Academic, USA, 1998.
- [11] J. C. Bezdek, "Pattern Recognition with Fuzzy Objective Function Algorithms". ISBN 0-306-40671-3, 1981.
- [12] J. C. Dunn, "A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters," *Journal of Cybernetics* 3: 32-57, 1973.
- [13] I. Gath and A. B. Geva, "Unsupervised optimal fuzzy clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol 11(7), pp 773-781, 1989.
- [14] J. C. Bezdek, C. Coray, R. Gunderson and J. Watson, "Detection and characterization of cluster substructure i. linear structure: Fuzzy c-lines." *SIAM Journal on Applied Mathematics*, 40(2), 339-357.
- [15] J. Yu, and M. S. Yang. "A generalized fuzzy clustering regularization model with optimality tests and model complexity analysis." *Fuzzy Systems, IEEE Transactions on* 15.5 (2007): 904-915.
- [16] T. A. Runkler, and C. Katz. "Fuzzy clustering by particle swarm optimization." *Fuzzy Systems, 2006 IEEE International Conference on*. IEEE, 2006.
- [17] H. C. Liu, J. M. Yih, D. B. Wu, S. W. Liu. "Fuzzy C-mean clustering algorithms based on Picard iteration and particle swarm optimization." *Education Technology and Training, 2008. and 2008 International Workshop on Geoscience and Remote Sensing. ETT and GRS 2008. International Workshop on*. Vol. 2. IEEE, 2008.
- [18] B. J. Zhao. "An ant colony clustering algorithm." *Machine Learning and Cybernetics, 2007 International Conference on*. Vol. 7. IEEE, 2007.
- [19] G. Gan, J. Wu, and Z. Yang. "A genetic fuzzy k-Modes algorithm for clustering categorical data." *Expert Systems with Applications* 36.2 (2009): 1615-1620.
- [20] F. Yang, T. Sun, and C. Zhang. "An efficient hybrid data clustering method based on K-harmonic means and Particle Swarm Optimization." *Expert Systems with Applications* 36.6 (2009): 9847-9852.
- [21] T. Niknam, B. Amiri. "An efficient hybrid approach based on PSO, ACO and k-means for cluster analysis." *Applied Soft Computing*, 10(1), 183-197.
- [22] P. Melin, F. Olivas, O. Castillo, F. Valdez, J. Soria, and M. Valdez. "Optimal design of fuzzy classification systems using PSO with dynamic parameter adaptation through fuzzy logic." *Expert Systems with Applications* 40, no. 8 (2013): 3196-3206.
- [23] L. Y. Chuang, C. J. Hsiao, and C. H. Yang. "Chaotic particle swarm optimization for data clustering." *Expert systems with Applications* 38.12 (2011): 14555-14563.
- [24] H. Izakian, and A. Abraham. "Fuzzy C-means and fuzzy swarm for fuzzy clustering problem." *Expert Systems with Applications* 38.3 (2011): 1835-1838.
- [25] S. Das, A. Abraham, and A. Konar, "Automatic kernel clustering with a multi-elitist particle swarm optimization algorithm." *Pattern Recognition Letters*, 29(5), 688-699, 2008.
- [26] R. C. Eberhart and J. Kennedy, A New Optimizer using Particle Swarm Theory, In *Proc. 6th Symp. Micro Machine and Human Science*, Nagoya, Japan 1995, 29-43.
- [27] J. C. Bezdek. "Cluster validity with fuzzy sets." (1973): 58-73
- [28] R. N. Dave, "Validating fuzzy partitions obtained through c-shells clustering." *Pattern Recognition Letters*, 17(6), 613-623, 1996.
- [29] Y. Fukuyama and M. Sugeno, "A new method of choosing the number of clusters for the fuzzy c-means method." *Proceeding of fifth fuzzy Syst. Sympo.*, pp.247-250, 1989.
- [30] X. L. Xie, and G. Beni, "A validity measure for fuzzy clustering. *Pattern Analysis and Machine Intelligence*," *IEEE Transactions on*, 13(8), 841-847, 1991.
- [31] N. R. Pal, and J. C. Bezdek, "On cluster validity for the fuzzy c-means model." *Fuzzy Systems, IEEE Transactions on*, 3(3), 370-379, 1995.
- [32] K. L. Wu, and M. S. Yang, "A cluster validity index for fuzzy clustering." *Pattern Recognition Letters*, 26(9), 1275-1291, 2005.
- [33] A. Strehl, "Relationship-based clustering and cluster ensembles for high-dimensional data mining," 2002.
- [34] W. Pedrycz and J. V. de Oliveira, "A development of fuzzy encoding

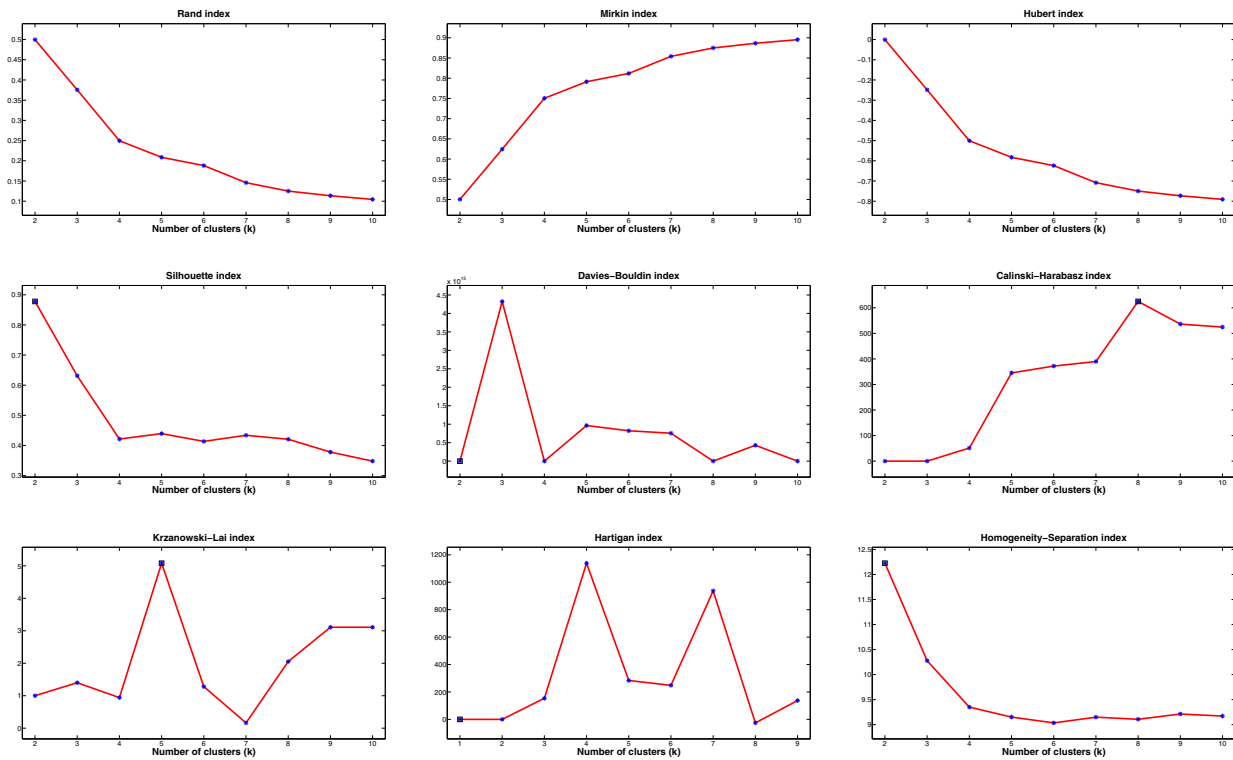


Fig. 3: FCM using 9 different validity indices

- and decoding through fuzzy clustering,” IEEE Trans. Instrum. Meas., vol. 57, no. 4, pp. 829837, Apr. 2008.
- [35] A. Frank & A. Asuncion, UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science, 2010.
- [36] W. M. Rand, “Objective criteria for the evaluation of clustering methods”. Journal of the American Statistical Association (American Statistical Association) 66 (336): 846850, 1971
- [37] B. Mirkin, “Mathematical classification and clustering: From how to what and why (pp. 172-181).” Springer Berlin Heidelberg, 1998.
- [38] L. J. Hubert, and J. R. Levin, “A general statistical framework for assessing categorical clustering in free recall,” Psychological Bulletin, 83(6), 1072, 1976.
- [39] P. J. Rousseeuw, “Silhouettes: a graphical aid to the interpretation and validation of cluster analysis,” Journal of computational and applied mathematics, 20, 53-65, 1987.
- [40] D. L. Davies and D. W. Bouldin, “A cluster separation measure. Pattern Analysis and Machine Intelligence,” IEEE Transactions on, (2), 224-227, 1979.
- [41] R. B. Calinski and J. Harabasz, “A Dendrite Method for Cluster Analysis,”Comm. in Statistics, vol. 3, pp. 1-27, 1974.
- [42] R. J. Bolton, and W. J. Krzanowski, “Projection pursuit clustering for exploratory data analysis.” Journal of Computational and Graphical Statistics, 12(1), 2003.
- [43] J. A. Hartigan, and M. A. Wong, “Algorithm AS 136: A k-means clustering algorithm.” Journal of the Royal Statistical Society. Series C (Applied Statistics), 28(1), 100-108, 1979.
- [44] G. Chen, S. A. Jaradat, N. Banerjee, T. S. Tanaka, M. S. Ko, and M. Q. Zhang, “Evaluation and comparison of clustering algorithms in analyzing ES cell gene expression data.” Statistica Sinica, 12(1), 241-262, 2002.

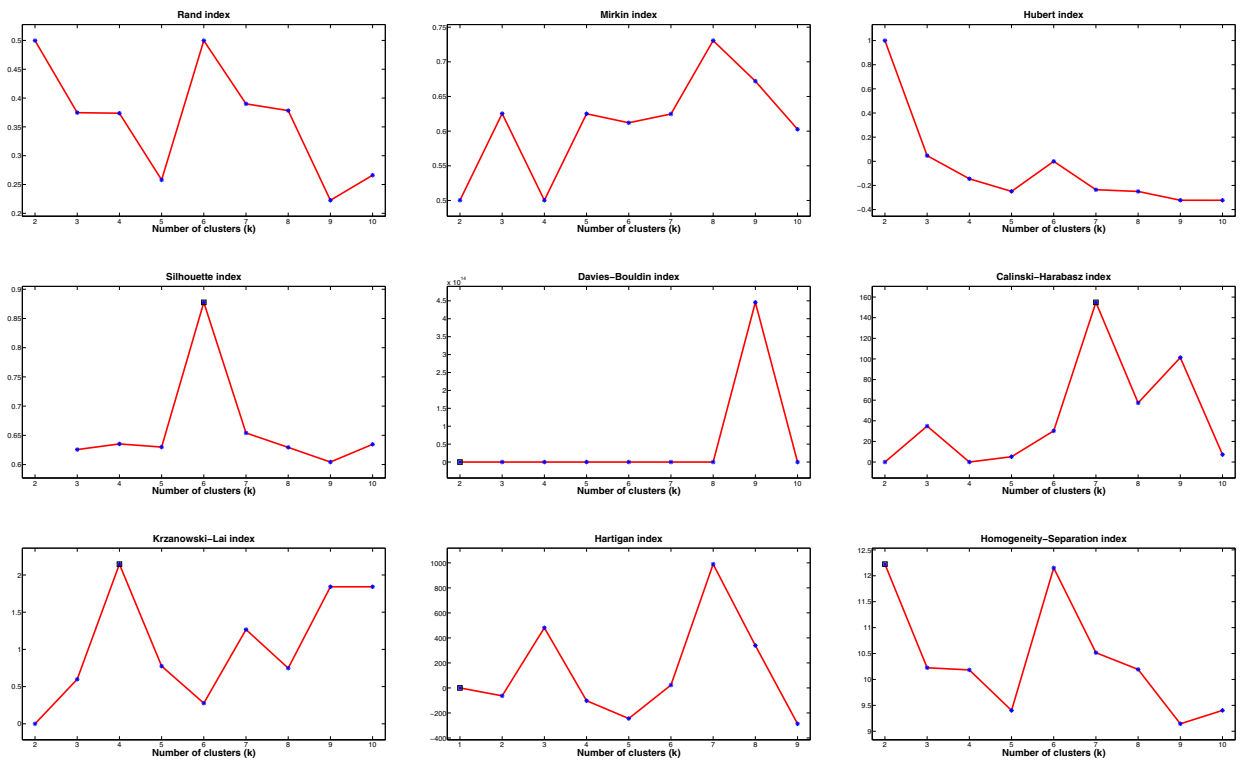


Fig. 4: FPSO using 9 different validity indices