A Method for Finding the Appropriate Number of Clusters

Huan Doan and Dinh Nguyen
Department of Information System, University of Information Technology, Vietnam

Abstract: *Drawback of almost partition based clustering algorithms is the requirement for the number of clusters specified at the beginning. Identifying the true number of clusters at the beginning is a difficult problem. So far, there were some works studied on this issue but no method is perfect in every case. This paper proposes a method to find the appropriate number of clusters in the clustering process by making an index indicated the appropriate number of clusters. This index is built from the intra-cluster coefficient and inter-cluster coefficient. The intra-cluster coefficient reflects intra-distortion of the cluster. The inter-cluster coefficient reflects the distance among clusters. Those coefficients are made only by extremely marginal objects of clusters. The looking for the extremely marginal objects and the building of the index are integrated in a weighted FCM algorithm and it is calculated suitably while the weighted Fuzzy C-Means (FCM) is processing. The Extended weighted FCM algorithm integrated this index is called Fuzzy C-Means-Extended (FCM-E). Not only does the FCM-E seek the clusters, but it also finds the appropriate number of clusters. The authors experiment with the FCM-E on some data sets of University of California, Irvine (UCI): Iris, Wine, Breast Cancer Wisconsin, and Glass and compare the results of the proposed method with the results of the other methods. The results of proposed method obtained are encouraging.*

Keywords: *Method for finding the number of clusters, appropriate a number of clusters, fuzzy c-means, clustering algorithm.*

Received December 18, 2014; accepted March 3, 2016

1. Introduction

Clustering algorithms play important role in data mining. They are widely used in practice such as marketing, information retrieval, image processing [6], etc. The output of a clustering algorithm is clusters so that intra-cluster similarity is maximized and the inter-cluster similarity is minimized. The number of clusters has to specify at the beginning, therefore intra-cluster similarity is dependent on a number of clusters selected. Evidently, if it chooses a small number of clusters, it will make big clusters. The object's similarity in big clusters is not high. In contrast, if it chooses the big number of clusters, it will make many small clusters. In this case, the clustering result is not good also. Here, a difficult problem of clustering algorithms arises that how many clusters are optimal and it is how to specify. Is it possible for us to build an algorithm that has both capabilities of seeking clusters and finding the appropriate number of clusters? This paper proposes a method to answer this question.

Authors make two coefficients $\bar{\alpha}$, $\bar{\beta}$ with an index $\bar{\gamma}$ that is based on them. The intra-cluster coefficient $\bar{\alpha}$ reflects intra-distortion of cluster through the maximum distance and the mean distance of cluster's extremely marginal objects (see Figure 1). The inter-cluster coefficient $\bar{\beta}$ reflects the distance among clusters. The $\bar{\beta}$ is the ratio between the closest distance from this cluster's centre to an extremely marginal object of other cluster and the mean distance from this cluster's central to all of extremely marginal objects of other cluster respectively. Authors use $\bar{\gamma}$ as index indicated the appropriate number of clusters. Then authors integrate $\bar{\alpha}$, $\bar{\beta}$ and $\bar{\gamma}$ into a weighted FCM algorithm [1, 5], and they are calculated adaptively while the weighted FCM is processing. The new algorithm integrated $\bar{\alpha}$, $\bar{\beta}$ and $\bar{\gamma}$ is called Fuzzy C-Means-Extended (FCM-E) so that, not only does it seek clusters but it also finds the appropriate number of clusters. For other partition-based clustering algorithms, the calculating and the integrating of coefficients $\bar{\alpha}$, $\bar{\beta}$ and $\bar{\gamma}$ can be studied for carrying out in a similar manner.

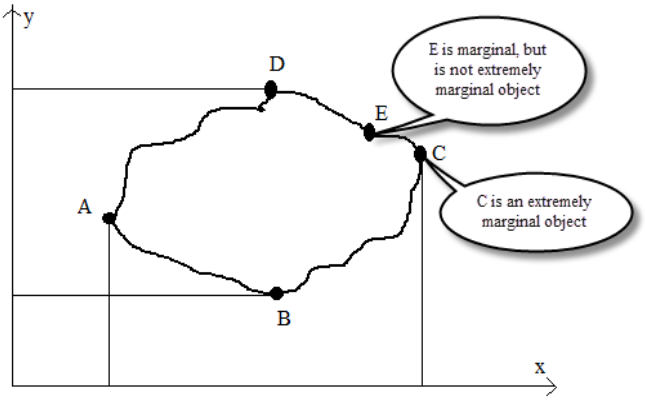


Figure 1. Describing of extremely marginal Objects A, B, C, D in two dimensional space.

Here, the authors chose weighted FCM to integrate $\bar{\gamma}$ because the authors thought that the addition of weighted vector is making the algorithm

more flexible. The attributes of object can play a different role instead of the same. Experimentally, while FCM-E runs from 1 to c_{max} where c_{max} being a given maximum number of clusters (c_{max}< n, n is the number of objects of data set), the authors find that $\bar{\gamma}$ normally indicates the best number of clusters c_{best} at neighborhood that it gets local minimum value after it decreases fast and normally start to have stable trend. In addition, the authors find that $\bar{\gamma}$ normally indicates the best number of clusters c_{best} when the weight of an attribute reflecting exactly the contribution of attributes, i.e. the bigger value of an attribute has, the bigger weight of the attribute has. To facilitate, here the authors use Euclid measure.

The authors experiment with FCM-E on some data sets of University of California, Irvine (UCI) [17] and compare the results of the proposed method with the results of other authors. The results of the proposed method obtained are encouraging.

The remainder of the paper is organized as follows. The next section reviews some existing methods for finding the number of clusters. Section 3 presents method including definition of some concepts, building coefficients $\bar{\alpha}$, $\bar{\beta}$ and index $\bar{\gamma}$ and analyzing of their fluctuation, proposed FCM-E algorithm, and evaluation of computational complexity. Section 4 presents experimental results on some data sets of UCI. Section 5 compares the results of the proposed method with the results of other methods. The last section is the conclusion of the paper.

2. Related Work

In this section, the authors mention some literatures concerned finding the number of clusters in clustering algorithms. Rosenberger and Chehdi [10] proposed an unsupervised clustering method called MLBG based upon the K-means algorithm. The originality of this method lies in the improvements realized in K-means algorithm and in its ability to determine the optimal number of clusters. Tibshirani *et al*. [16] proposed a statistic method Gap for estimating the number of clusters. The their method uses the output of clustering an algorithm, comparing the change in within the cluster dispersion to that expected under an appropriate reference null distribution. Sugar and James [14] developed a simple yet powerful non-parametric method for choosing the number of clusters based on distortion, a quantity that measures the average distance, per dimension, between each observation and its closest cluster center.

Salvador and Chan [11] proposed an algorithm that is method L finding the "knee" in a "# of clusters vs. clustering evaluation metric" graph. Sun *et al*. [15] proposed an improved FCM-based algorithm for selecting the number of clusters and an index for assessing clustering results. The index is a function of original data, cluster centers, and membership. Almost all mentioned methods only based on distortion information from within clusters. Shao and Wu [13] proposed an information-based criterion for determining the number of clusters in the problem of regression clustering. Sanguinetti *et al*. [12] present a novel spectral clustering algorithm that allows them automatically to determine the number of clusters in a data set. The algorithm is based on a theoretical analysis of the spectral properties of block diagonal affinity matrices; in contrast to established methods, they do not normalize the rows of the matrix of eigenvectors, and argue that the non-normalized data contains key information that allows the automatic determination of the number of clusters present. Pham *et al*. [9] suggested building a measure function for determining the number of clusters.

Kyrgyzov *et al*. [7] introduced a new criterion, based on Minimum Description Length (MDL), to estimate an optimal number of clusters. This criterion, called Kernel MDL (KMDL), is particularly adapted to the use of kernel K-means clustering algorithm. Its formulation is based on the definition of MDL derived for Gaussian Mixture Model (GMM). Motivated by the gap method of Tibshirani *et al*. [16], Yan and Ye [18] proposed the weighted gap and the Difference of Difference-weighted (DD-weighted) gap methods for estimating the number of clusters in data using the weighted within-clusters sum of errors: a measure of the within-clusters homogeneity. In addition, they proposed a "multilayer" clustering approach, which is shown to be more accurate than the original gap method, particularly in detecting the nested cluster structure of the data.

To estimate the number of clusters, Cheong and Lee [3] explore the problem through the EM algorithm, Maximum a Posteriori and Gibbs sampler. In addition, they investigate the Bayesian Information Criteria (BIC), the Laplace Metropolis criteria and the modified Fisher's criteria in order to determine the number of clusters. Capitaine and Frélicot [2] introduce an approach to find the optimal number of clusters of a fuzzy partition. They use measures of separation and degree of overlap of the clusters based on triangular norms and a discrete Sugeno integral. Zhao *et al*. [20] re-formulated the BIC in partitioning based on clustering algorithm. BIC is a method for detecting the number of clusters. To improve BIC getting results that are more reliable, they proposed an angle-based method for knee point finding of BIC.

Zalik [19] proposes a clustering validity index that addresses cluster validation especially clusters widely differ in density or size. This index is based on compactness and overlap measures. The author proposes ratio and summation type of index using the same compactness and overlap measures. The maximal value of index denotes the optimal fuzzy partition that is expected to have a high compactness and a low degree of overlap among clusters. Nguyen

and Doan [8] proposed an approach determining number of clusters based on coefficientsα, β obtained in the clustering process. The coefficients are built on all objects in clusters.

The above-mentioned methods have the different advantages and disadvantages but no method is perfect in any case. In the following section, the authors propose a method with the desire to do better than existing works.

3. Method

3.1. Definition of Concepts

Here, the authors define some concepts concerned cluster to base for the building of coefficients next.

- A marginal object: object locates on margin of cluster.
- An extremely marginal object: object locates on margin of cluster and has to be at least an attribute being minimum value or maximum value (in Euclid measure).
- The maximum distance among all extremely marginal objects is the distance between two extremely marginal objects being most distant in a cluster. It is denoted $\bar{d}_{\max}$.
- The mean distance among all extremely marginal objects is the ratio of the sum of all distances among extremely marginal objects in a cluster and number of distances. It is denoted $\bar{d}_{avr}$.
- The minimum distance from this cluster's centre to other cluster's all extremely marginal objects is the distance that from this cluster's centre to another cluster's closest extremely marginal object. It is denoted $\bar{\phi}_{\min}$.
- The mean distance from this cluster's centre to other cluster's all extremely marginal objects is the ratio of the sum of all distances from this cluster's centre to other cluster's all extremely objects and number of distances. It is denoted $\bar{\phi}_{avr}$.

3.2. Building of Coefficients and Analysis

Clearly, extremely marginal objects reflect quite exactly a form of cluster (see Figure 2). Thus, the authors build coefficients based on extremely marginal objects in clusters instead of building coefficients based on all objects in clusters as in [8].

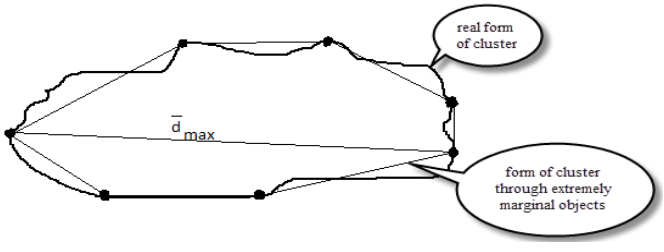


Figure 2. Form of cluster.

Set intra-cluster coefficient $\bar{\alpha}$ reflects the distortion within the cluster:

$$\bar{\alpha} = \frac{\bar{d}_{\max}}{\bar{d}_{avr}} = \frac{\max_{q}\left(\sqrt{\sum_{l=1}^{k} w_l (x_{il} - x_{jl})^2}\right)}{\frac{\sum_{q}\left(\sqrt{\sum_{l=1}^{k} w_l (x_{il} - x_{jl})^2}\right)}{q}} \quad (1)$$

Where j=1,…,p, i=1,…,p, $i \neq j$, p is number of extremely marginal objects in the each cluster, k is the number of dimensions, q is number of distances between extremely marginal objects in the each cluster. When a cluster has at least two objects, it has $\bar{\alpha} \geq 1$. When the cluster has only one object, it accepts $\bar{\alpha} = 0$.

Call $\bar{\alpha}_{\max}$ is the largest coefficient of the coefficients $\bar{\alpha}$ of all the clusters after a clustering process:

$$\bar{\alpha}_{\max} = \max_{c}(\bar{\alpha}) \quad (2)$$

Where, c is the number of clusters.

While the more $\bar{\alpha}_{\max}$ fluctuates, the more clusters distort, this is an indicator showing a trend of splitting cluster. On the contrary, while $\bar{\alpha}_{\max}$ go to stable trend, this is an indicator showing an appropriate number of clusters.

Set inter-cluster coefficient $\bar{\beta}$ reflects the ratio of distance between two clusters:

$$\bar{\beta} = \frac{\bar{\phi}_{\min}}{\bar{\phi}_{avr}} = \frac{\min_{p}\left(\sqrt{\sum_{l=1}^{k} w_l (x_{il} - c_{jl})^2}\right)}{\frac{\sum_{p}\left(\sqrt{\sum_{l=1}^{k} w_l (x_{il} - c_{jl})^2}\right)}{p}} \quad (3)$$

Where C_j is a centre of cluster j, j = 1,..,c, i = 1,..,p, where c is the number of clusters, p is the number of extremely marginal objects in clusters except cluster j, k is the number of dimensions. $\bar{\beta} \leq 1$. When it has only one cluster, it accepts $\bar{\beta} = 0$.

Call $\bar{\beta}_{\max}$ is the largest coefficient of the coefficients $\bar{\beta}$ after a clustering process.

$$\bar{\beta}_{\max} = \max_{c(c-1)}(\bar{\beta}) \quad (4)$$

Where, c is the number of clusters.

While the more $\bar{\beta}_{\max}$ tends to 1, the more two clusters tend small and closely, this is an indicator showing a trend of merging two clusters.

From the above analysis and through experiment, the authors find that $\bar{\alpha}_{\max}$ indicating the best number of clusters at neighborhood that it gets a local minimum value after it decreases fast. In addition, the $\bar{\beta}_{\max}$ indicating the best number of clusters at

neighborhood that it closely tends 1 after it increases fast.

$$\text{Set} \quad \bar{\gamma} = \bar{\alpha}_{\max} + (1 - \bar{\beta}_{\max}) \tag{5}$$

The authors assume $\bar{\gamma}$ will indicate the best number of clusters at neighborhood that it gets a local minimum value after it fluctuate fast and normally start to have stable trend.

3.3. Algorithm FCM-E

The authors integrate calculation of the index $\bar{\gamma}$ into the weighted FCM algorithm and use the index $\bar{\gamma}$ indicated an appropriate number of clusters in a clustering process. The Extended weighted Algorithm 1 FCM integrated $\bar{\gamma}$ is called FCM-E presented as follows:

Algorithm1. FCM-E

1) *Input n objects x_i, fuzzy parameter m > 1, epsilon small enough.*
2) *Input weighted vector w.*
3) *Input c_{min} and c_{max} (c_{min} >= 1, c_{min} < c_{max} < n).*
4) *For c = c_{min} to c_{max}*
   a) *Initialization matrix of members U_{cxn}*
   b) *Calculation of centre of cluster j C_j (j = 1,..,c)*
   c) *Update the distance matrix D (c x n)*
   d) *Update matrix of members U*
   e) *If the change of the matrix U is small enough compared to the previous step, then go to step f) otherwise go to step b)*
   f) *Based on the matrix U, x_i is arranged into clusters according to rules as follows: x_i will belong to any cluster that it has the greatest degree*
   g) *Save the c of data clusters at step c to disk*
   h) *Get all extremely marginal objects of all clusters*
   i) *Compute $\bar{\gamma}$ based on (1), (2), (3), (4), (5) as in section 3.2*
5) *Select c_{best} at neighborhood when $\bar{\gamma}$ get a local minimum value after $\bar{\gamma}$ fluctuates fast and may start to have stable trend.*
6) *Take the c_{best} of data clusters at step c_{best} from disk.*

3.4. Evaluation of Computational Complexity of FCM-E

The computational complexity of the algorithm FCM is O(tcn). Where n is the number of objects in a data set, c is the number of clusters, t is the number of iterations.

The computational complexity of the algorithm FCM-E is calculated as follows:

Set q ($q = c_{max}$) is the number of iterations for selecting of number of clusters.

O(FCM-E) = O(FCM) + [O(FCM) + O(Step g) + O($\bar{\gamma}$)]*q

O(FCM-E) = O(FCM) + [O(FCM) + O(Step g) + O($\bar{\alpha}$) + O($\bar{\beta}$)]*q

Where O(FCM) is the computational complexity of FCM, O(Step g) is the computational complexity of step g.

O($\bar{\alpha}$) is the computational complexity of $\bar{\alpha}$, O($\bar{\beta}$) is the computational complexity of $\bar{\beta}$.

On an average, the number of objects in each cluster is $\frac{n}{c}$.

Set p is the number of extremely marginal objects in each cluster. On an average, it can assume p ≈ k, with k is the number of dimensions.

So O(Step g) = c*(n/c)*k = n*k = O(nk)

O($\bar{\alpha}$) = c*p*(p-1) = c*k*(k - 1) = $ck^2 - ck \leq ck^2$.

Thus O($\bar{\alpha}$) ≈ O(ck^2)

O($\bar{\beta}$) = c*c*p = c*c*k = c^2k = O(c^2k)

In fact, the number of clusters c normally is smaller n, the number of iterations t, q normally is smaller n, and the number of attributes k normally is smaller n. It can set: t = max (t, q, k, c).

$$O(\text{FCM-E}) = O(tcn) + [O(tcn) + O(nk) + O(ck^2) + O(c^2k)]^* q$$
$$= O(tcn) + [O(tqcn) + O(knq) + O(k^2cq) + O(c^2kq)]$$
$$\leq O(nt^2) + [O(nt^3) + O(nt^2) + O(t^4) + O(t^4)]$$
$$\approx O(nt^3) < O(tqn^2)$$

The computational complexity of the algorithm FCM-E is smaller algorithm FCM+ in [8].

4. Experiment

In this section, the authors implement algorithm FCM-E and experiment on some data sets: Iris, Wine, Breast Cancer Wisconsin, and Glass (From: UCI Machine Learning Repository). Run algorithm FCM-E with parameters m = 2, epsilon = 0.0001 on these data sets.

4.1. Iris Data Set

The Iris data set has four attributes of 3 species namely Iris Setosa, Iris Versicolor, and Iris Virginica. Attributes are sepal length, sepal width, petal length, and petal width. Each class contains 50 instances.

Run algorithm FCM-E with a number of cluster c from 1 to 12 with the Iris data set and weighted sepal length = 0.5, weighted sepal width = 0.2, weighted petal length = 0.2, and weighted petal width = 0.1, it has $\bar{\gamma}$ obtained in running each once presented in Table 1. Here, the weight of attributes is conjectured on its contribution value.

Table 1. Statistic table of $\bar{\gamma}$ with Iris data set.

Algorithm FCM-E with number of clusters c	1	2	3	4	5	6	7	8	9	10	11	12
$\bar{\gamma}$ (c)	2.95	2.36	1.96	2.29	2.29	2.29	2.28	2.21	2.2	1.94	2.03	2.03

It looks at the Table 1 and the graph of $\bar{\gamma}$(c) in Figure 3., it predicts that the location of the number of appropriate clusters is in the neighborhood of the point at which the graph decreases fast and gets the local minimum value and starts to have stable trend. With the Iris data set, $\bar{\gamma}$(c) indicating the appropriate number of clusters is 3.

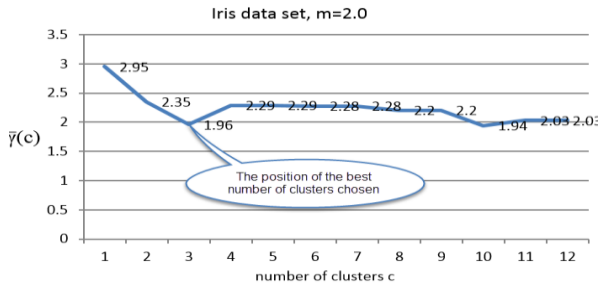


Figure 3. Graph $\bar{\gamma}$ (c) of Iris data set.

4.2. Wine Data Set

The Wine data set, which consists of 13 chemical attributes for 178 Italian wines belonging to 3 separable classes.

Run algorithm FCM-E with a number of cluster c from 1 to 10 with the Wine data set and weights of attributes in Table 2, it has $\bar{\gamma}$ obtained in running each once presented in Table 3. The weight of each attribute is the rate of the sum value of this attribute and the sum value of every attribute.

Table 2. Table of weights of attributes –wine data set.

Attributes	**1**	**2**	**3**	**4**	**5**	**6**
Weights of attributes	0.014465421	0.002599589	0.002633156	0.021691474	0.110979632	0.002553707
7	8	9	10	11	12	13
0.002257911	0.000402625	0.001770148	0.001770148	0.001065352	0.002905949	0.834904888

Table 3. Statistic table of $\bar{\gamma}$ with wine data set.

Algorithm FCM-E with number of clusters c	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
$\bar{\gamma}$ **(c)**	4.75	3.94	3.52	3.92	3.58	3.53	3.53	3.52	3.51	3.91

It looks at the Table 3 and the graph of $\bar{\gamma}$(c) in Figure 4., it predicts that the location of the number of appropriate clusters is in the neighborhood of the point at which the graph decreases fast and gets local minimum value. With the Wine data set, $\bar{\gamma}$(c) indicating the appropriate number of clusters is 3.

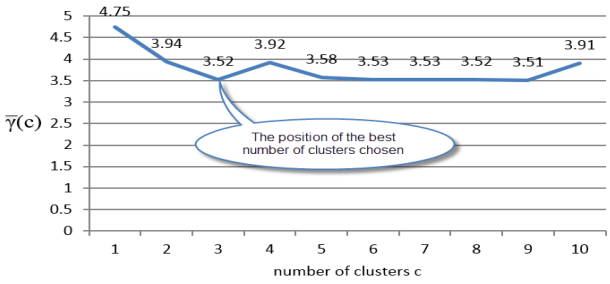


Figure 4. Graph $\bar{\gamma}$ (c) of wine data set.

4.3. Breast Cancer Wisconsin Data Set

The Breast Cancer Wisconsin data set has eleven attributes among attributes 2 through 10 have been used to represent objects. It has 16 missing attribute values and they are replaced by the mean value of attribute. Attribute 11 is class attribute. Number of objects is 699 (as of 15 July 1992). Each object has one of two possible classes: benign or malignant.

Run algorithm FCM-E with a number of cluster c from 1 to 12 with the Breast Cancer Wisconsin data set and weights of attributes in Table 4, it has $\bar{\gamma}$ obtained in running each once presented in Table 5. Here, the weight of each attribute is the rate of the sum of this attribute and the sum of every attribute.

Looking at the Table 5 and the graph of $\bar{\gamma}$(c) in Figure 5., it predicts that the location of the number of appropriate clusters is in the neighborhood of the point at which the graph decreases fast and gets local minimum value. With the Breast Cancer Wisconsin Data Set, $\bar{\gamma}$(c) indicating the appropriate number of clusters is 3. The next candidates of the appropriate number of clusters are 2 or 4.

Table 4. Table of weights of attributes – breast cancer wisconsin data set.

Attributes	**2**	**3**	**4**	**5**
Weights of attributes	0.1564812	0.111026655	0.113611027	0.099422316
6	7	8	9	10
0.11391507	0.1259248	0.121769535	0.101550623	0.056298774

Table 5. Statistic table of $\bar{\gamma}$ with breast cancer wisconsin data set.

Algorithm FCM-E with number of clusters c	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
$\bar{\gamma}$ **(c)**	3.01	2.69	2.35	2.36	2.51	2.85	3.41	3.19	2.59	3.06	2.92	3.38

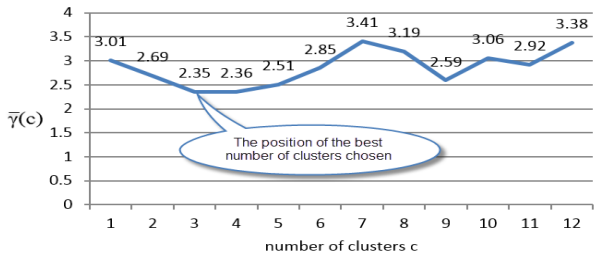


Figure 5. Graph $\bar{\gamma}$ (c) of breast cancer wisconsin data set.

4.4. Glass Data Set

The Glass data set has eleven attributes including an Id# attribute and a class attribute. The attributes from 2 to 10 have been used to represent objects and they are continuously valued. Attribute 11 is class attribute. The Glass data set has 214 objects belonging to 6 separable classes.

The authors normalized the Glass data set and then run algorithm FCM-E with a number of cluster c from 2 to 12 with the normalized Glass data set and weights of attributes in Table 6; it has $\bar{\gamma}$ obtained in running

each once presented in Table 7. The weight of each attribute is the rate of the sum of this attribute and the sum of every attribute.

Looking at the Table 7 and the graph of $\bar{\gamma}$(c) in Figure 6., it predicts that the location of the number of appropriate clusters is in the neighborhood of the point at which the graph decreases fast and gets the local minimum value. With the normalized Glass Data Set, $\bar{\gamma}$(c) indicating candidates of the appropriate number of clusters are 4 or 5 or 6.

Table 6. Table of weights of attributes –normalized glass data set.

Attributes	2	3	4	5
Weights of attributes	0.134516282	0.176829894	0.087784619	0.127207035
6	7	8	9	10
0.20218374	0.066779697	0.151921823	0.023677931	0.029098979

Table 7. Statistic table of $\bar{\gamma}$ with normalized Glass data set.

Algorithm FCM-E with number of clusters c	2	3	4	5	6	7	8	9	10	11	12
$\bar{\gamma}$ (c)	6.86	6.44	6.01	4.24	4.85	3.36	3.4	3.15	2.96	2.95	2.44

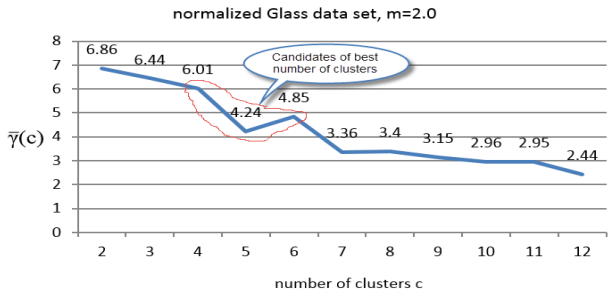


Figure 6. Graph $\bar{\gamma}$ (c) of normalized glass data set.

5. Comparision

5.1. Comparing the Results of the Proposed Method with the Results of the Fuzzy Modeling Approach of Capitaine H. and Frélicot C.

In this section, the authors compare the results of the proposed method with the results of the Fuzzy Modeling Approach of Capitaine and Frélicot [2] and the number of clusters in fact on some data sets. The results are presented in Table 8 that shows the efficiency of the proposed method.

In the fact, the number of clusters of the Iris data set is 3, the result of the proposed method method is also 3, but the result of the method of Capitaine and Frélicot [2] is 2.

In the fact, the number of clusters of the Wine data set is 3, the result of the proposed method and the result of the method of Capitaine and Frélicot [2] is the same.

Through this comparison, it can find that the proposed method of determining the number of clusters is quite appropriate with the number of clusters of some data sets in fact.

Table 8. Comparing the results of the proposed method with the results of fuzzy modeling approach.

Data sets	SOI T (method of Capitaine H., & Frélicot C.)				$\bar{\gamma}$ (the proposed method)		Number of clusters in fact
	Cmax	S	A	Ho	Cmax	Number of clusters	
Iris	10	2	2	2	12	3	3
Wine	10	3	3	3	10	3	3

5.2. Comparing the Results of the Proposed Method with the Results of the Gap Methods of Yan M. and Ye K.

Here, the authors compare the results of the proposed method with the results of the Gap Methods of Yan and Ye [18]. The results are presented in Table 9 that shows an incentive of the proposed method.

Table 9. Comparing the results of the proposed method with the results of the gap methods.

Method	Iris (G = 3)	Breast Cancer (G = 2)
Gap/uni	6/8	9
Gap/pc	4	9
Gap/uni	6	2
Gap/pc	4	2
DDGap/uni	2	2
DDGap/pc	2	2
Multilayer/pc	3	3
$\bar{\gamma}$ (the proposed method)	3	3

In the Table 9, the authors realize that it has not any method estimated the number of clusters of both data sets exactly. The methods having the best results are DDGap/uni, DDGap/pc, Multilayer/pc, and $\bar{\gamma}$ (the proposed method). DDGap/uni and DDGap/pc estimate exactly the number of clusters of Breast Cancer data set and estimate approximately the number of clusters of Iris data set. On the contrary, Multilayer/pc and the proposed method estimate exactly the number of clusters of Iris data set and estimate approximately the number of clusters of Breast Cancer data set.

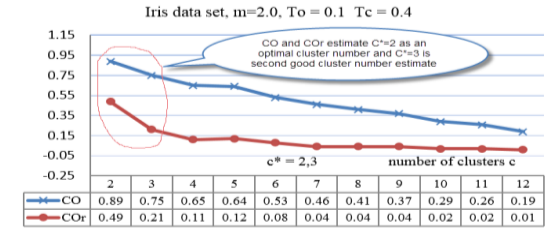
5.3. Comparing the Results of the Proposed Method with the Results of Cluster validity index of Zalik K.

In this section, the authors compare the results of the proposed method with the results of Cluster validity index of Zalik [19] on real data sets: Iris, normalized Glass. The authors do not find fuzziness coefficient in experiments on their Ionosphere data set and therefore the authors do not compare to this data set. Results are presented in following that shows an incentive of the proposed method.

5.3.1. Iris Data Set

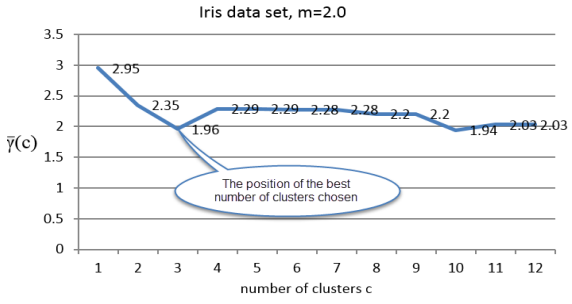
On the Iris data set, the proposed method indicating the best number of clusters is 3 that accords with the number of clusters in fact but the CO and COr of Zalik [19] estimated c = 2 as an optimal number cluster. In

addition, they showed c = 3 is the second best number cluster estimate (see Figure 7).



	2	3	4	5	6	7	8	9	10	11	12
CO	0.89	0.75	0.65	0.64	0.53	0.46	0.41	0.37	0.29	0.26	0.19
COr	0.49	0.21	0.11	0.12	0.08	0.04	0.04	0.04	0.02	0.02	0.01

a) Cluster validity index CO and COr of Zalik.

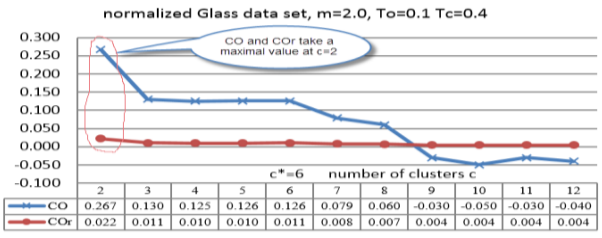


b) Index $\bar{\gamma}$ of the proposed method.

Figure 7. Comparing the results of the proposed method with the results of cCluster validity index of zalik, R. K. (2010) on the Iris data set.

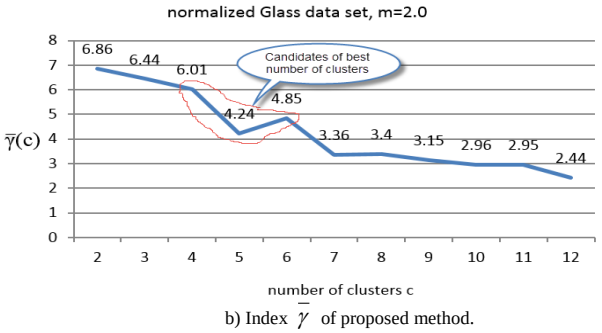
5.3.2. Normalized Glass data set

On the normalized Glass data set, the proposed method indicating the best number of clusters is in the neighborhood of the number of cluster c = 5 therefore the candidates are 4 or 5 or 6 near by the true number of clusters in fact (6 is the true number of clusters in fact). Whereas that CO and COr of Zalik [19] take a maximal value at c = 2 (the optimal number cluster can be found when finding the maximal value of validity index CO and COr) (see Figure 8).



	2	3	4	5	6	7	8	9	10	11	12
CO	0.267	0.130	0.125	0.126	0.126	0.079	0.060	-0.030	-0.050	-0.030	-0.040
COr	0.022	0.011	0.010	0.010	0.011	0.008	0.007	0.004	0.004	0.004	0.004

a) Cluster validity index CO and COr of zalik.



b) Index $\bar{\gamma}$ of proposed method.

Figure 8. Comparing the results of the proposed method with the results of Cluster validity index of Zalik (2010) on the normalized Glass data set.

6. Conclusions and Future Work

The determining of the true number of clusters has important sense in actual applications of clustering algorithms. In this paper, the authors have presented a method for finding the appropriate number of clusters in the clustering process. The proposed method is based on building an index $\bar{\gamma}$ indicated the appropriate number of clusters. This index is a combination of intra-cluster coefficient $\bar{\alpha}$ and inter-cluster coefficient $\bar{\beta}$. The coefficient $\bar{\alpha}$ reflects intra-distortion of clusters and the coefficient $\bar{\beta}$ reflects the distance among clusters and therefore the index $\bar{\gamma}$ reflects relation both intra-cluster and inter-cluster. By experiment, the authors assume $\bar{\gamma}$ will indicate the best number of clusters at neighborhood that it gets local minimum value after it fluctuate fast and normally start to have stable trend. The index $\bar{\gamma}$ has been integrated into the weighted FCM algorithm and the new algorithm is called FCM-E. Not only does FCM-E seek clusters, but it also finds the appropriate number of clusters. The experimental results of the FCM-E on some data sets of UCI are quite matched with the number of clusters of data sets in fact. The authors compare the results of the proposed method with the results of other methods and the results of the proposed method obtained are encouraging.

Recommendations for further research are the building the index $\bar{\gamma}$ based on the other measures and the integrating it into the other clustering algorithms required the input parameter being the number of clusters.

References

[1] Bezdek J., Ehrlich R., and Full W., "FCM: The Fuzzy C-Means Clustering Algorithm," *Computers and Geosciences*, vol. 10, no. 2-3, pp. 191-203, 1984.

[2] Capitaine H. and Frélicot C., "A Fuzzy Modeling Approach to Cluster Validity," *in Proceedings of IEEE International Conference on Fuzzy Systems*, Jeju Island, pp. 462-467, 2009.

[3] Cheong Y. and Lee H., "Determining the Number of Clusters in Cluster Analysis," *Journal of the Korean Statistical Society*, vol. 37, no. 2, pp. 135-143, 2008.

[4] Doan H. and Nguyen T., "An Adaptive Method to Determine the Number of Clusters in Clustering Process," *in Proceedings of The International Conference on Computer and Information Sciences*, Kuala Lumpur, pp. 1-6, 2014.

[5] Hathaway R. and Bezdek J., "Recent Convergence Results for the Fuzzy c-Means

Clustering Algorithms," *Joumal of Classificanon*, vol. 5, no. 2, pp. 237-247, 1988.
[6] Kalti K. and Mahjoub M., "Image Segmentation by Gaussian Mixture Models and Modified FCM Algorithm," *The International Arab Journal of Information Technology*, vol. 11, no. 1, pp. 11-18, 2014.
[7] Kyrgyzov I., Kyrgyzov O., Maıtre H., and Campedel M., "Kernel MDL to Determine the Number of Clusters," *in Proceedings of 5th International Conference Machine Learning and Data Mining in Pattern Recognition. Lecture Notes in Computer Science*, Leipzig, pp. 203-217, 2007.
[8] Nguyen T. and Doan H., "An Approach to determine the Number of Clusters for Clustering Algorithms," *in Proceedings of 4th International Conference Computational Collective Intelligence. Technologies and Applications*, Vietnam, pp. 485-494, 2012.
[9] Pham T., Dimov S., and Nguyen D., "Selection of K in K-means Clustering," *Journal of Mechanical Engineering Science*, vol. 219, no.1, pp.103-119, 2005.
[10] Rosenberger C. and Chehdi K., "Unsupervised Clustering Method with Optimal Estimation of the Number of Clusters: Application to Image Segmentation," *in Proceedings of 15th International Conference on Pattern Recognition*, Barcelona, 2000.
[11] Salvador S. and Chan P., "Determining the Number of Clusters/Segments in Hierarchical Clustering/Segmentation Algorithms," *in Proceedings of 16th IEEE International Conference on Tools with Artificial Intelligence*, Boca Raton, 2004.
[12] Sanguinetti G., Laidler J., and Lawrence N., "Automatic Determination of the Number of Clusters using Spectral Algorithms," *IEEE Workshop on Machine Learning for Signal Processing*, Mystic, 2005.
[13] Shao Q. and Wu Y., "A consistent Procedure for Determining the Number of Clusters in Regression Clustering," *Journal of Statistical Planning and Inference*, vol. 135, no. 2, pp. 461-476, 2005.
[14] Sugar C. and James G., "Finding the Number of Clusters in a Data set: An Information Theoretic Approach," *Journal of the American Statistical Association*, vol. 98, no. 463, pp. 750-763, 2003.
[15] Sun H., Wang S., and Jiang Q., "FCM-Based Model Selection Algorithms for Determining the Number of Clusters," *Pattern Recognition*, vol. 37, no. 10, pp. 2027-2037, 2004.
[16] Tibshirani R., Walther G., and Hastie T., "Estimating the Number of Clusters in a Data Set Via the Gap Statistic," *Journal of the Royal Statistical Society*, vol. 63, no. 2, pp. 411-423 2001.
[17] UCI Machine Learning Repository, available at: http://archive.ics.uci.edu/ml/datasets.html, Last Visited, 2013.
[18] Yan M. and Ye K., "Determining the Number of Clusters Using the Weighted Gap Statistic," *Biometrics*, vol. 63, no. 4, pp. 1031-1037, 2007.
[19] Zalik K., "Cluster Validity Index for Estimation of Fuzzy Clusters of Different Sizes and Densities," *Pattern Recognition*, vol. 43, no. 10, pp. 3374-3390, 2010.
[20] Zhao Q., Hautamaki V., and Fränti P., "Knee Point Detection in BIC for Detecting the Number of Clusters," *in Proceedings of International Conference on Advanced Concepts for Intelligent Vision Systems*, France, pp. 664-673, 2008.

Huan Doan received his BSc degree in Mathematics from Hue University of Science, Vietnam in 1988, and MSc degree in Computer Science from University of Information Technology (UIT), Vietnam National University Ho Chi Minh city (VNU-HCM) in 2012. He is currently pursuing PhD degree in Computer Science from University of Information Technology (UIT), VNU-HCM. He is also the director of EnterSoft Software Solution Joint Stock Company, Ho Chi Minh City, Vietnam. He has published about 8 research papers in the area of data mining and artificial intelligence, data analysis and risk analysis at international/national level conferences and journals.

Dinh Nguyen Nguyen has been the Associate Professor at Department of Information Systems, University of Information Technology (UIT), Vietnam National University Ho Chi Minh city (VNU-HCM). He received his BSc degree in Mathematics from Dalat University in 1984, MSc degree in Information Technology from University of Science (VNU-HCM) in 1997 and PhD degree in Information Technology from Institute of Information Technology (IOIT), Vietnamese Academy of Science and Technology (VAST) in 2004. He has published more than 35 research papers in the area of database, data mining and data analysis at international/national level conferences and journals. He is currently guiding 3 PhD students in the area of data mining and data analysis