EMITTER International Journal of Engineering Technology
Vol.2, No.1, June 2014
ISSN: 2443-1168

Centronit: Initial Centroid Designation Algorithm for K-Means Clustering

Ali Ridho Barakbah[1] andKohei Arai[2]

[1]Electronic Engineering Polytechnic Institute of Surabaya
Jalan Raya ITS Surabaya 60111, Indonesia
E-mail: ridho@pens.ac.id

[2]Department of Information Science
Saga University, 1 Honjo, Saga 840-8502, Japan
E-mail: arai@is.saga-u.ac.jp

Abstract

Clustering performance of the K-means highly depends on the correctness of initial centroids. Usually initial centroids for the K-means clustering are determined randomly so that the determined initial centers may cause to reach the nearest local minima, not the global optimum. In this paper, we propose an algorithm, called as Centronit, for designation of initial centroidoptimization of K-means clustering. The proposed algorithm is based on the calculation of the average distance of the nearest data inside region of the minimum distance. The initial centroids can be designated by the lowest average distance of each data. The minimum distance is set by calculating the average distance between the data. This method is also robust from outliers of data. The experimental results show effectiveness of the proposed method to improve the clustering results with the K-means clustering.

Keywords: K-means clustering, initial centroids, K-meansoptimization.

1. Introduction

Clustering is an effort to classify similar objects in the same groups. Cluster analysis constructs good cluster when the members of a cluster have a high degree of similarity each other (internal homogeneity) and are not like members of other clusters (external homogeneity) [1][2]. It means the process to define a mapping $f:D \rightarrow C$ from some data $D=\{d_1,d_2, ...,d_n\}$ to some clusters $C=\{c_1,c_2, ...,c_n\}$ on similarity between d_i. The applications of clustering is diversely in many fields such as data mining, pattern recognition, image classification, biological sciences, marketing, city-planning, document retrievals, etc.

The most well known, widely used and fast methods for clustering is K-means clustering developed by Mac Queen in 1967. The simplicity of K-means clustering made this algorithm used in various fields. K-means clustering is a partitioning clustering method that separates data into k mutually excessive groups. Through such the iterative partitioning, K-means clustering minimizes the sum of distance from each data to its clusters. K-means clustering is very popular because of its ability to cluster a kind of huge data, and also outliers, quickly and efficiently. It remains a basic framework for developing numerical or conceptual clustering systems because various possibilities of distance and prototype choice [3].

However, K-means clustering is very sensitive to the designated initial starting points as cluster centers. K-means clustering generates initial clusters randomly. If a randomly designated initial starting point close to a final cluster center, then K-means clustering can find the final cluster center. It, however, is not always. If a designated initial point is far from the final cluster center, it will lead to incorrect clustering results [4]. Because of initial starting points generated randomly, K-means clustering does not guarantee the unique clustering results [5]. K-means clustering is difficult to reach global optimum, but only to one of local minima [6].

2. RELATED WORKS

Several methods proposed to solve the cluster initialization for K-means clustering. A recursive method for initializing the means by running K clustering problems is discussed by Duda and Hart (1973). A variation of this method consists of taking the entire data into account and then randomly perturbing it *k* times [5]. Bradley and Fayyad [7] proposed an algorithm that refines initial points by analyzing distribution of the data and probability of data density. Penã et al. [8] presented empirical comparison for the initialization methods for K-means clustering and concluded that the random and Kaufman initialization method outperformed the other two methods with respect to the effectiveness and the robustness of K-means clustering. Shehroz and Ahmad [5] proposed Cluster Center Initialization Algorithm (CCIA) to solve cluster initialization problem. CCIA is based on two observations, which some patterns are very similar to each other. It initiates with calculating mean and standard deviation for data attributes, and then separates the data with normal curve into certain partition. CCIA uses K-means and density-based multi scale data condensation to observe the similarity of data patterns before finding out the final initial clusters. The experimental results of CCIA performed the effectiveness and robustness of this method to solve the several clustering problems.

3. ORIGINALITY

In this paper, we propose a new approach, called as Centronit, to designate initial centroids for K-Means clustering. Centronit is based on the calculation of data density inside the certain range of data distribution.

Centronit does not involve probabilistic variables at all so that it produces the constant clustering results. Moreover, because of reflecting the data distribution, Centronit is robust for outliers.

4. DESIGN SYSTEM

4.1 Basic Theory of K-means Clustering

Let A={a_i | i=1, ..., n} be attributes of n-dimensional vector and X={x_i | i=1, ..., r} be each data of A. The K-means clustering separates X into k partitions called clusters S={s_i | i=1, ..., k} where M∈X is M={m_i | i=1, ..., $n(s_i)$} as members of S. Each cluster has cluster center of C={c_i | i=1, ..., k}.

K-means clustering algorithm can be described as follows:

1. Initiate its algorithm by generating random starting points of initial centroids c_k.
2. Calculate the distance $d(x,c)$ between vector x_i to cluster center c_k. Euclidean distance used to be used to express the distance.
3. Separate x_i into s_k which has minimum $d(x,c)$.
4. Determine the new cluster centers defined as:

$$c_i = \frac{1}{p}\sum_{j=1}^{p} m(s_i, j), \qquad where\ p = n(s_i) \tag{1}$$

5. Go back to step 2 until c_i = c_i-1

To calculate the distortion of K-means clustering, let E:X→S be encode function to cluster X into S, and D:S→X be the decode function. The distortion of clustering can be defined as:

$$Distortion = \left|\sum_{i=1}^{r}(x_i - D[E(x_i)])\right| \tag{2}$$

The correct clustering has x_i=D[E(x_i)], so that $Distortion$ is 0. A good clustering performs minimum $Distortion$. Therefore, it try to make $Distortion$ as minimum as possible. Referring Eq.1 and M∈X, the effort to minimize $Distortion$ can be set by minimizing P as:

$$P = \left|(m(s_i, j) - c_j)\right| \tag{3}$$

wherec_k is the cluster center of $m(s_i,k)$. Therefore, the determining of initial centroids for K-means is very important because it can determine the distortion and/or the precision of clustering results.

4.2 Basic Concept of our proposed Centroit

It is common that the ideal initial cluster center resides near average gravity of the cluster members. It means that it absolutely depends on thedata distribution of them. For a simple case which has only 1 cluster, if the cluster center $\bar{x}$ of the clusters assumed as one of the members, it performs:

$$\frac{1}{n}\sum_{j=1}^{n}\left|\bar{x}-x_j\right| < \frac{1}{n}\sum_{j=1}^{n-1}\left|x_i-x_j\right| \quad (4)$$

where $x_i \neq x_j$. Eq. 4 expresses that $\bar{x}$is the lowest average distance to the other members. Because the cluster center is unknown in a priori, Equation 4 can be modified in order to look for the nearest members from $\bar{x}$that can be analyzed it as below:

$$T_i = \frac{1}{n-1}\sum_{j=1}^{n-1}\left|x_i-x_j\right| \quad (5)$$

Where$x_i \neq x_j$. From Equation 5, the nearest members from desired cluster center which show the most minimal T_i can be determined.

4.3 Capturing certain area

Starting from Equation 5, we then expand the case, which consists more than one cluster. The differential distance $|x_i - x_j|$ can be calculated because it has scalability factor in the certain area. Let we apply the circular region with diameter λ to appoint the certain range of area as shown in Figure 1.

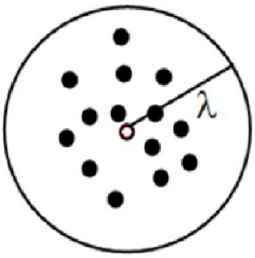

Figure 1. Certain area showed by circular region with diameterλ

Then, Equation 5 can be modified as follows:

$$T_i = \frac{1}{n_i-1}\sum_{j=1}^{n_i-1}\left|x_i-x_j\right| \quad (6)$$

where $|x_i - x_j| \leq \lambda$ and n_i is number of points inside the circular area which point *i* is its center.

Based on Equation 6, the most minimum *T* as the nearest members from desired cluster center is determined. This implies that the most minimum *T* can express the good initial centroids.

It can be explained that the nearest point to cluster center, assumed as *a*, can relatively cover calculation of all points which make n_a close to total points. On the contrary, the furthest point to cluster center, assumed as *b*, can just cover calculation of some points because the distance to furthest point is relatively ≤ λ. Figure 2(a) and 2(b) illustrates the different coverage calculation between two points. Therefore, it makes $n_a > n_b$, so that it causes $T_a < T_b$.

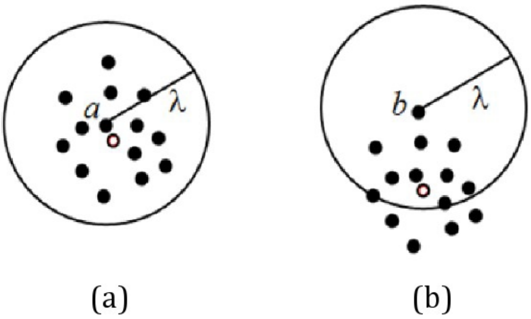

Figure 2. Illustration of different coverage calculation between point *a* in (a) and point *b* in (b), with n_a = 14, n_b= 10.

Now, we can apply Equation 6 for clustering cases more than one cluster by setting the appropriate value λ. Figure 3 shows use of λ for capturing the certain area of clustering.

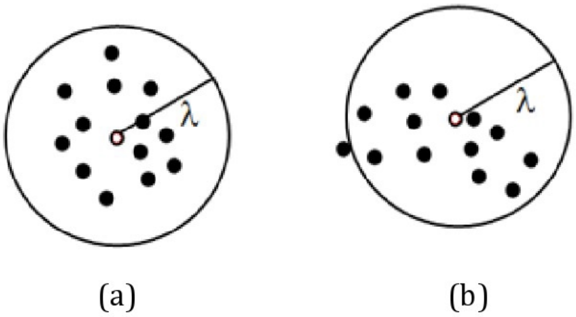

Figure 3. Capturing two areas of clustering cases

Because it just applied one value of λ, it could be n_m lower than total members of cluster, as is shown in Figure 3 that there is certain point outside the circular range. It will also make that the nearest point to the cluster center can be represented by the lowest minimal value of T_i. Meanwhile, it also can answer why this approach can be robust from outlier of data.

4.4 Setting appropriate λ

The most critical point is how to set the appropriate λ. It will be difficult if we set manually because the clustering cases are very various. If the dimension of data is more than three, it is difficult to imagine the visualization of the data distribution. In this paper, we try to overcome with the automatic setting of value λ.

Referring the case in Figure 3, the ideal λ can be determined by half of differential distance between the cluster centers, as is shown in Figure 4.

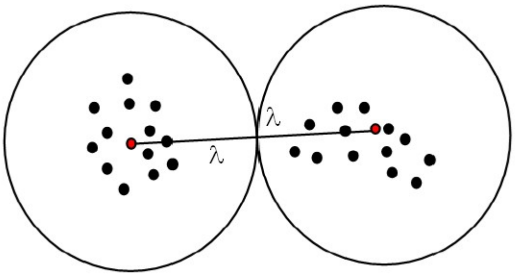

Figure 4. Determining λ with differential distance between the cluster centers.

It may cause the certain points those are near to the boundary involve some members of the other clusters, as is shown in Figure 5. That is what we aim because it will make T_i bigger, so that it has low chance to be initial cluster center.

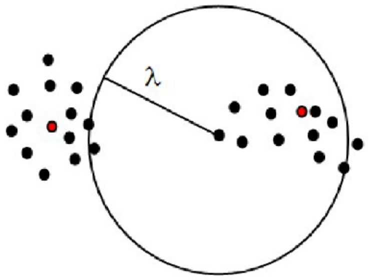

Figure 5. Illustration of a point which is involving some members of the other clusters.

The desired cluster centers of the clustering cases are not known in a priori so that the value of λ from differential distance between cluster centers has not been determined. The following approach to solve this problem is proposed. The distribution of the data actually can express the average distance among them. Based on Equation 5, it is possible to determine λ by calculating the average distance between all data points as follows:

$$\lambda = \frac{1}{2n(n-1)} \sum_{i=1}^{n} \sum_{j=1}^{n-1} \sqrt{\left|x_i - x_j\right|} \quad (7)$$

where$x_i \neq x_j$. From Equation 7, the value of λcan be set automatically. It expresses the average distance among separated clusters.

4.5 Execution steps

In this subsection the following execution steps of our proposed Centronit for initial centroids of *K*-means clustering is proposed,

1. Set $X=\{x_i \mid i=1, ..., r\}$ as each data of A where $A=\{a_i \mid i=1, ..., n\}$ is attribute of n-dimensional vector.
2. Set K as the predefined number of clusters.

3. Compute diameter of circular area λas Equation 7.
4. Compute n_i as number of points inside the circular area λ for x_i which point i is its center. Execute for i=1…n where n is total number of points.
5. Compute differential distance T_i for each point as Equation 6.
6. Init p=0 as a counter to compute number of initial centroids that will be designated.
7. Find T_m as the most minimum value of T.
8. Increment p=p+1
9. Assign c_p = T_m as designated initial cluster center
10. Set T_m with a defined big value
11. Repeat to step 7 until p=K

After processing, it will generate the designated initial centroids c_p where p=1, 2, ...,K. Then, we can apply it as initial centroids for K-means clustering. The experiment results will perform the accuracy of the proposed method.

5. EXPERIMENT AND ANALYSIS

Two experimental data, normal random data distribution datasets and real world datasets are used for evaluation of the proposed Centronit for K-means clustering.

5.1 Normal random distributed data

In order to evaluate an ability of the proposed method for well-separate case of clustering, two dimensional normal random data distribution datasets are used. We make 1000 experiments, then clustering performance of the proposed method is compared with Hierarchical methods (Single Linkage, Centroid Linkage, Complete Linkage and Average Linkage), Fuzzy C-means and K-means clustering using random initialization. For Fuzzy C-means clustering, 1.5 for degree of fuzziness is used as an example. For K-means clustering, we take the average results of 100 experiments.

The following variance factors V is defined as a performance measure in the experiments. Variance constraint [9] can express the density of the clusters with variance within cluster and variance between clusters [10][11]. The ideal cluster has minimum variance within clusters (V_w) to express internal homogeneity and maximum variance between clusters (V_b) to express external homogeneity [12].

$$V = \frac{V_w}{V_b} \tag{8}$$

Table 1 shows the variance factor comparison among the aforementioned methods. From Table 1, the variance factor of the proposed method shows the lowest V, same as V of the Hierarchical methods so that it is concluded that the proposed method does work so well for the well-separated clustering cases.

Table 1. Comparison of variance factor for normal distributed random dataset

Clustering Algorithm	***V***
Hierarchical Clustering	0.002997
Fuzzy c-means	0.005898
K-means (random init.)	0.0164869
K-means using Centronit	0.002997

5.2 Real world datasets

The real world datasets used are Iris data, Wine data, Fossil data, Ruspini data, Letter Recognition data and New Thyroid data which are widely used and well known datasets for evaluation of clustering algorithms.

The raw data of the real world datasets are used because comparison of clustering performance between the proposed method and the other existing methods is concerned. If we normalize the data, even though it is usual to get the better clustering results, the clustering results are not only dependent on clustering methods, but also are dependent on normalization methods.

Clustering performance of Single Linkage, Centroid Linkage, Complete Linkage, Average Linkage, Fuzzy c-means, *K*-means clustering with random designated initial cluster center is compared to the proposed method. The same datasets as CCIA [5] is used are utilized for the comparison, even though its clustering result computed after normalizing the data ranges from 0 to 1. For Fuzzy c-means clustering, 1.5 for degree of fuzziness is again used. For *K*-means clustering using random initialization, 100 iteration times is used and take the average results.

The following error percentage which is calculated from the number of misclassified patterns and the total number of patterns in the datasets is evaluated.

$$Error = \frac{Numberofmisclassified}{Numberofpatterns} x100\% \tag{9}$$

5.2.1 Iris dataset

This dataset is from the UCI Repository [13]. This dataset contains information about Iris flowers. There are three classes of Iris flowers, namely Iris Setosa, Iris Versicolor and Iris Virginica. The dataset consists of 150 examples with 4 attributes. One class is well separable against the other two. The others have a large overlap. Table 2 shows the comparison of error ratio between our proposed Centronit and other clustering algorithm for Iris dataset.

Table 2. Comparison of Error ratio for Iris dataset

Clustering Algorithm	Error (%)
Single Linkage	32
Centroid Linkage	9.3333
Complete Linkage	16
Average Linkage	9.3333
Fuzzy C-means	13.524
K-means (random init.)	17.7507
K-means using CCIA	11.33
K-means using Centronit	10.6667

5.2.2 Wine dataset

We also obtained this dataset from UCI Repository [13]. The data is the result of a chemical analysis of wines grown in a region in Italy but derived from three different cultivars. There are three classes. The dataset consists of 178 examples each with 13 continuous attributes. The dataset contains distribution 59 examples of class 1, 71 examples for class 2 and 48 examples for class 3. Table 3 shows the comparison of error ratio between our proposed Centronit and other clustering algorithm for Wine dataset.

Table 3. Comparison of Error ratio for Wine dataset

Clustering Algorithm	Error (%)
Single Linkage	57.3034
Centroid Linkage	38.764
Complete Linkage	32.5843
Average Linkage	38.764
Fuzzy c-means	30.3371
K-means (random init.)	32.6197
K-means using CCIA	5.05
K-means using Centronit	29.7753

The high error happened with *K*-means clustering using Centronit compared with CCIA because the raw data actually has far difference scale among attributes. There is an attribute that has high scale of value compared to the others. For this case, the data is usually better to standardize before clustering. Table 4 performs the error of *K*-means clustering using Centronit after normalizing the data using 4 different normalization methods.

Table 4. Error of *K*-means using Centronit after normalizing wine dataset

Normalization Method	Error (%)
Min-Max (0-1)	5.618
Z-Score	2.809
Sigmoid	2.809
Softmax	2.809

5.2.3 Fossil dataset

The Fossil data is obtained from Chernoff [14]. It consists of 87 nummulitidae specimens from Eocene yellow limestone formation of northwestern Jamaica. There are three 6 attributes with 3 classes which the distribution is 40 examples of class 1, 34 examples of class 2 and 13 examples of class 3. Table 5 shows the comparison of error ratio between our proposed Centronit and other clustering algorithm for Fossil dataset.

Table 5. Comparison of Error ratio for Fossil dataset

Clustering Algorithm	Error (%)
Single Linkage	13.7931
Centroid Linkage	11.4943
Complete Linkage	14.9425
Average Linkage	9.1954
Fuzzy c-means	11.5057
K-means (random init.)	8.5931
K-means using CCIA	0
K-means using Centronit	4.5977

K-means clustering using CCIA showed the smallest error compared to the others as is shown in Table 5. However, if any of the normalization is taken into account, then *K*-means clustering with Centronit shows the smallest errors as is shown in Table 6.

Table 6. Error of *K*-means using Centronit after normalizing Fossil dataset

Normalization Method	Error (%)
Min-Max (0-1)	0
Z-Score	12.6437
Sigmoid	4.5977
Softmax	12.6437

5.2.4 Ruspini dataset

The Ruspini dataset represents a simple, well-known example that is commonly used as a benchmark problem in evaluating clustering methods and is widely available, incorporated as a built-in data object in both R and S-plus statistics packages [15]. The dataset consists of 75 bi-variate attribute vectors. There are fthe classes. The dataset contains 23, 20, 17 and 15 in classes 1, 2, 3 and 4, respectively. Table 7 shows the comparison of error ratio between our proposed Centronit and other clustering algorithm for Ruspini dataset.

Table 7. Comparison of Error ratio for Ruspini dataset

Clustering Algorithm	Error (%)
Single Linkage	0
Centroid Linkage	0

Clustering Algorithm	Error (%)
Complete Linkage	4
Average Linkage	0
Fuzzy c-means	0
K-means (random init.)	13.7787
K-means using CCIA	4
K-means using Centronit	0

5.2.5 Letter recognition dataset

This dataset obtained from UCI Repository [13]. The objective is to identify each of a large number of black-and-white rectangular pixel displays as one of the 26 capital letters in the English alphabet. The character images were based on 20 different fonts and each letter within these 20 fonts was randomly distorted to produce a file of 20,000 unique stimuli. Each stimulus was converted into 16 primitive numerical attributes (statistical moments and edge counts), which were then scaled to fit into a range of integer values from 0 through 15. The training data consists of first 16000 items and then used the resulting model to predict the letter category for the remaining 4000. For experimental purpose we have taken 595 patterns of letter A and 597 patterns of letter D from the training dataset, as CCIA has done. Table 8 shows the comparison of error ratio between our proposed Centronit and other clustering algorithm for Letter Recogniton dataset.

Table 8. Comparison of Error ratio for Letter Recognition dataset

Clustering Algorithm	Error (%)
Single Linkage	49.8322
Centroid Linkage	48.1544
Complete Linkage	42.7852
Average Linkage	6.8792
Fuzzy c-means	13.1711
K-means (random init.)	8.2326
K-means using CCIA	8.55
K-means using Centronit	8.2215

5.2.6 New thyroid dataset

The new thyroid dataset is also obtained from UCI Repository [13]. The dataset contains information about classification whether a patient's thyroid to the class euthyroidism, hypothyroidism or hyperthyroidism. The diagnosis (the class label) was based on a complete medical record, including anamnesis, scan etc. The dataset consists 5 attributes, with 215 examples. The distribution is 150 of class euthyroidism, 35 of class hypothyroidism and 30 of class hyperthyroidism. Table 9 performs the comparison of error ratio between our proposed Centronit and other clustering algorithm for New Thyroid dataset.

Table 9. Comparison of Error ratio for New Thyroid dataset

Clustering Algorithm	Error (%)
Single Linkage	29.7674
Centroid Linkage	27.907
Complete Linkage	28.3721
Average Linkage	26.0465
Fuzzy c-means	14.4186
K-means (random init.)	20.9842
K-means using Centronit	13.9535

6. CONCLUSION

It is widely reported that the K-means clustering algorithm suffers from initial centroids. The main purpose is to optimize the designation of the initial centroids for K-means clustering. Therefore, in this paper we proposed Centronit as a new algorithm of initial centroid designation algorithm for K-Means Clustering. This algorithm is based on the calculating the average distance of the nearest data inside region of the minimum distance. The initial centroids can be designated by the lowest average distance of each data. The minimum distance is set by calculating the average distance between the data. This algorithm creates the unique clustering results because it does not involve the probabilistic calculation. Moreover, because of observing the data distribution, Centronit is robust for outliers. Experimental results with normal random data distribution and real world datasets perform the accuracy and improved clustering results as compared to some clustering methods.

REFERENCES

[1] G.A. Growe, **Comparing Algorithms and Clustering Data: Components of The Data Mining Process**, Thesis, Department of Computer Science and Information Systems, Grand Valley State University, 1999.

[2] V.E. Castro, **Why So Many Clustering Algorithms-A Position Paper**, *ACM SIGKDD Explorations Newsletter*, Volume 4, Issue 1, pp. 65-75, 2002.

[3] H. Ralambondrainy, **A Conceptual Version of The K-Means Algorithm**, *Pattern Recognition Letters 16*, pp. 1147-1157, 1995.

[4] YM. Cheung, ***k*-Means: A New Generalized K-Means Clustering Algorithm***, *Pattern Recognition Letters 24*, pp. 2883-2893, 2003.

[5] S.S. Khan, A. Ahmad, **Cluster Center Initialization Algorithm for K-Means Clustering**, *Pattern Recognition Letters*, 2004.

[6] B. Kövesi, JM. Boucher, S. Saoudi, **Stochastic K-Means Algorithm for Vector Quantization**, *Pattern Recognition Letters 22*, pp. 603-610, 2001.

[7] P.S. Bradley, U.M. Fayyad, **Refining Initial Points for K-Means Clustering**, *Proc. 15th International Conference on Machine Learning (ICML'98)*, 1998.

[8] J.M. Penã, J.A. Lozano, P. Larrañaga, **An Empirical Comparison of The Initilization Methods for The K-Means Algorithm**, *Pattern Recognition Letters 20*, pp. 1027-1040, 1999.

[9] C.J. Veenman, M.J.T. Reinders, E. Backer, **A Maximum Variance Cluster Algorithm**, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 24, No. 9, pp. 1273-1280, 2002.

[10] S. Ray, R.H. Turi, **Determination of Number of Clusters in K-Means Clustering and Application in Colthe Image Segmentation**, *Proc. 4th ICAPRDT*, pp.137-143, 1999.

[11] W.H. Ming, C.J. Hou, **Cluster Analysis and Visualization**, *Workshop on Statistics and Machine Learning, Institute of Statistical Science*, Academia Sinica, 2004.

[12] Ali Ridho Barakbah, Kohei Arai, **Identifying Moving Variance to Make Automatic Clustering for Normal Dataset**, *Proc. IECI Japan Workshop 2004 (IJW 2004)*, Musashi Institute of Technology, Tokyo, 2004.

[13] UCIRepository (http://www.sgi.com/tech/mlc/db/).

[14] C. Yi-tsuu, **Interactive Pattern Recognition**, *Marcel Dekker Inc*., New York and Basel, 1978.

[15] R.K. Pearson, T. Zylkin, J.S. Schwaber, G.E. Gonye, **Quantitative Evaluation of Clustering Results Using Computational Negative Controls**, *Proc. 2004 SIAM International Conference on Data Mining*, Lake Buena Vista, Florida, 2004.