Journal of Mathematical Research with Applications
Nov., 2019, Vol. 39, No. 6, pp. 745–754
DOI:10.3770/j.issn:2095-2651.2019.06.017
Http://jmre.dlut.edu.cn

Mongolian Similar Elements Clustering via Immune Clone Algorithm

Chunhua[1,2], **Chao ZHANG**[1,*], **Yan LIU**[3], **Wei WU**[1]

1. *School of Mathematical Sciences, Dalian University of Technology, Liaoning* 116024*, P. R. China;*
2. *College of Computer Sciences and Technology, Inner Mongolia University for Nationalities, Inner Mongolia* 028043*, P. R. China;*
3. *Department of General Education, Dalian Polytechnic University, Liaoning* 116034*, P. R. China*

Dedicated to the Memory of Professor L. C. HSU on the Occasion of His 100th Birthday

Abstract Text clustering is an important research issue of clustering technique. It aims to use the similar characteristics or similar expression to group the text so that the texts in the same clusters have the greatest similarity, and those in different clusters have the greatest dissimilarity. There are many characteristics in Mongolian structure and writing-mode compared with other kinds of characters. By combining K-means and clone immune algorithm, we propose a novel clustering technique called ICKM. Numerical experiments on four elements sets illustrate the validity of our method in the clustering task for Mongolian.

Keywords immune clone algorithm; Mongolian similar elements clustering; K-means

MR(2010) Subject Classification 91C20; 62H30; 68W40

1. Introduction

The Mongolian is the official language of Mongolia, and is one of the most widely-spoken and best-known members in the Mongolic language family. Mongolian nationality spreads over many provinces and autonomous regions of China, for example, Inner Mongolia Autonomous Region, Xinjiang, Gansu and Heilongjiang. Among them, the Inner Mongolia Autonomous Region is the largest and most concentrated area of Mongolian nationalities and the main language is Mongolian. Along with the information explosion, the processing of massive information becomes one key and challenging problem. The language recognition and clustering is a new technology that combines artificial intelligence with image processing and there have been many mature methods designed for English and Chinese. Nevertheless, there is still few work on the Mongolian recognition and clustering.

Artificial immune algorithm is a new artificial intelligent technique that imitates the function of the natural immune system, and provides many evolutionary learning mechanisms including

Received August 24, 2019; Accepted October 12, 2019
Supported by the Project of Inner Mongolia University for Nationalities (Grant No. NMDYB18009), the National Natural Science Foundation of China (Grant No. 61473328; 11401076) and the Fundamental Research Funds for the Central Universities (Grant No. DUT18JC02).

* Corresponding author
E-mail address: chao.zhang@dlut.edu.cn (Chao ZHANG)

noise tolerance, unsupervised learning, self-organization and memory. It has attracted many research attention from researchers of various fields relevant to artificial intelligent.

Clustering analysis belongs to an unsupervised learning task of machine learning and it groups the similar data in a cluster [1, 2]. There are many widely used clustering methods including K-means [3–5], fuzzy C-Means [6–9], hierarchical clustering [10], genetic algorithm based clustering technique [11–13] and XK-means [12].

K-means is a well-known clustering algorithm, where the cluster number K is provided in advance as an input. Based on the cluster number, K-means randomly selects K number of initial points from a data set. However, the clustering results are sensitive to the choice of initial cluster center. Additionally, K-means generally is trapped in the local minima that leads to a poor clustering result. To overcome the drawback in optimization, some genetic algorithms, which can provide the global optimization, have been widely used to optimize the K-means clustering problems. In this paper, we propose a hybrid clustering technique that incorporates K-means with immune clone algorithm.

The rest of this paper is organized as follows. In Section 2, we brief the basic mechanism of the immune clone algorithm as well as the clustering technique. In Section 3, we present the ICKM algorithm. In Section 4, we show the experimental results on K-means, GKA (Genetic K-means algorithm) [11], IGKM (Immune genetic k-means algorithm) [14] and ICKM algorithms and the last section concludes the paper.

2. Immune clone algorithm and clustering technique

In this section, we present some basic knowledge on immune clone algorithm and clustering technique.

2.1. Artificial immune system

Artificial immune system (AIS) is a model that imitates biological immune system including diversity, distributed computation, error tolerance, dynamic learning and self-monitoring [15, 16] and provides a new search algorithm for many artificial intelligent problems, for example, pattern recognition, learning and associative memory [17, 18]. In recent years, there have been many research interests lying in AIS and its application. AIS inherits the characteristics of general immune system and adopts a group search strategy including following several steps: (1) generating initial population; (2) computation of fitness value for each antibody; (3) selection operation, crossover operation and mutation operation; and (4) generating new population. Finally, through such iterative operations, the optimal solution of the problem is obtained with a high probability. Compared to other algorithms, the immune algorithm ensures the diversity of the population because of its own diversity and maintenance mechanisms so as to overcome the premature problem.

2.2. The basic concept of clone technique

Clone is an important concept in biological immune system. It can successively produce next generation and form population without sexual reproduction [19]. The basic idea of clone algorithm is to randomly generate antibody population consisting of n antibodies. After the operations on these antibodies including clonal proliferation, clonal mutation and clonal selection, it then chooses the excellent antibodies. In the process of generations, the clonal proliferation is performed to form sub-antibodies; the clonal mutation is performed for the ergodicity of the evolution process, which guarantees the appearance of a global optimal antibody in the evolution process, and updating of the super antibody for catching forever the global optimal antibody once it appears; and the clonal selection operation is performed for the population to have a good evolution direction.

In the clone algorithm, the antigen corresponds to the objective function and various constraints of the optimization problem; the antibody is related with the solution of the problem, and the affinity of antibodies and antigens reflects the matching degree between the optimal solution and the object function. In the implementation of evaluation process, the transition state of antibody population can be expressed as follows:

$$P(t) \xrightarrow{\text{Clonal Proliferation}} P'(t) \xrightarrow{\text{Clonal Mutation}} P''(t) \xrightarrow{\text{Clonal Selection}} P(t+1). \tag{2.1}$$

The following is briefly introduction of above three operators.

(1) Clonal proliferation operation T_c^C.

In the artificial immune system, the clonal proliferation operators are defined as follows:

$$T_c^C(P(t)) = [T_c^C p_1(t), T_c^C p_2(t), \ldots, T_c^C p_i(t), \ldots, T_c^C p_N(t)], \tag{2.2}$$

$$T_c^C p_i(t) = I_i.p_i(t), \quad i = 1, 2, \ldots, N, \tag{2.3}$$

where N is the population size and $p_i(t)$ is the each antibody of population. After cloning, the population becomes

$$P'(t) = \{p'_1(t), p'_2(t), \ldots, p'_N(t)\}. \tag{2.4}$$

(2) Clonal mutation operation T_m^C

For each antibody after being cloned, the clonal mutation is carried out according to mutation probability P_m:

$$P''(t) = T_m^C(P'_1(t)). \tag{2.5}$$

In this process, the population of mutated solutions is bred. Clonal mutation can expand the searching range (increasing the diversity of antibodies) and can help prevent premature evolution.

(3) Clonal selection operation

The usual roulette strategy is used in the clone algorithm. The probability that an antibody $p_i(t)$ is selected from the existing population to breed the next generation is given by

$$P(p_i(t)) = \frac{F(p_i(t))}{\sum_{l=1}^{N} p_l(t)}, \quad i = 1, 2, \ldots, N; \tag{2.6}$$

$$F(p_i(t)) = \frac{1}{\sum_{k=1}^{K} \sum_{i=1}^{n} u_{ik} \|x_i - \mathbf{c}_k\|}, \tag{2.7}$$

where $F(p_i(t))$ is the fitness value of antibody $p_i(t)$ and $\mathbf{c}_k$ presents the center of k-th cluster. Finally, the population of next generation is generated according to following formula:

$$P(t+1) = T_s^C(P''(t)). \tag{2.8}$$

2.3. Clustering technique

Clustering analysis groups the data according to some similar criterion such as similar structure or similar expression. In this manner, it is expected that the data in the same clusters have the greatest similarity, and the data in different clusters have the greatest dissimilarity. Clustering analysis has been effectively applied in various engineering and scientific disciplines, for example, psychology, biology, medicine, computer vision, communications and remote sensing.

Because of its simplicity and fast convergence speed, K-means and its variants have always been the most frequently used clustering algorithm. The main idea of this algorithm is to minimize (or maximize) the value of a certain criterion function, where the number of cluster K is required in advance and then K initial points are randomly selected from a data set. However, one shortcoming of K-means is easily to fall into local optimum. To improve the quality of the clusters, the genetic algorithm has been incorporated with K-means. The introduction of genetic mechanism has successively kept K-means avoiding local optimum [6,11]. The main contributions of this paper is to impose the immune clone algorithm into K-means and provide a new clustering technique, called ICKM algorithm.

3. ICKM algorithm

In this section, we first give some necessary notations and then describe the proposed ICKM algorithm.

3.1. Necessary notations

The aim is to group a set of n elements into K clusters. Each element is expressed as a $D\times D$ matrix:

$$x_i = \begin{bmatrix} x_{11}^i & x_{12}^i & \cdots & x_{1D}^i \\ \vdots & \vdots & \vdots & \vdots \\ x_{D1}^i & x_{D2}^i & \cdots & x_{DD}^i \end{bmatrix}, \quad i=1,2,\ldots,n, \quad k=1,2,\ldots,K.$$

Define

$$u_{ik} = \begin{cases} 1, & \text{if the } i\text{-th element belongs to the } k\text{-th cluster,} \\ 0, & \text{otherwise,} \end{cases} \tag{3.1}$$

and then form a label matrix $\mathbf{U} = [u_{ik}]$. It is required that each element should belong to precisely one cluster, and each cluster contain at least one element. Therefore, we have

$$\sum_{k=1}^{K} u_{ik} = 1, \quad i=1,2,\ldots,n, \tag{3.2}$$

$$1 \leq \sum_{i=1}^{n} u_{ik} < n, \quad k = 1, 2, \ldots, K. \tag{3.3}$$

Let the center of the k-th cluster be

$$\mathbf{c}_k = \begin{bmatrix} c_{11}^k & c_{12}^k & \cdots & c_{1D}^k \\ \vdots & \vdots & \vdots & \vdots \\ c_{D1}^k & c_{D2}^k & \cdots & c_{DD}^k \end{bmatrix},$$

where each element of $\mathbf{c}_k$ is defined by

$$c_{dj}^k = \frac{\sum_{i=1}^n u_{ik} \sum_{d=1}^D \sum_{j=1}^D x_{dj}^i}{\sum_{i=1}^n u_{ik}}. \tag{3.4}$$

Let $\| \cdot \|_F$ stand for the Frobenius norm. Then, for any two $D \times D$ matrices

$$y = \begin{bmatrix} y_{11} & y_{12} & \cdots & y_{1D} \\ \vdots & \vdots & \vdots & \vdots \\ y_{D1} & y_{D2} & \cdots & y_{DD} \end{bmatrix}$$

and

$$z = \begin{bmatrix} z_{11} & z_{12} & \cdots & z_{1D} \\ \vdots & \vdots & \vdots & \vdots \\ z_{D1} & z_{D2} & \cdots & z_{DD} \end{bmatrix},$$

the distance between them is

$$\|y - z\|_F = \Big(\sum_{i=1}^{D} \sum_{j=1}^{D} |y_{ij} - z_{ij}|^2 \Big)^{\frac{1}{2}}. \tag{3.5}$$

3.2. Evaluation strategies

To evaluate the clustering results, we adopt the following evaluation strategies: the sum squared error (SSE) [12], Xie-Beni index (XB) [12] and Davies-Bouldin (DB) evaluation criteria [20]. The SSE is defined as

$$SSE = \sum_{k=1}^{K} \sum_{i=1}^{n} u_{ik} \|x_i - \mathbf{c}_k\|^2. \tag{3.6}$$

SSE will be used as the evaluation function in the genetic operation of the numerical simulation later on. Generally speaking, lower SSE means better clustering result. Then XB is defined as

$$XB = \frac{SSE}{n * d_{\min}}, \tag{3.7}$$

where d_{min} is the shortest distance between clusters. To define the DB index [2, 13], we first defined the within-cluster scatter S_k:

$$S_k = (\frac{1}{|\mathbf{c}_k|} \sum_{x \in \mathbf{c}_k} \| x - \mathbf{c}_k \|^2)^{\frac{1}{2}}, \tag{3.8}$$

where $\mathbf{c}_k$ (resp., $|\mathbf{c}_k|$) denotes the set (resp., the number) of the samples belongs to the cluster k. Then, define a term for cluster as

$$R_k = \max_{j,j\neq k} \frac{S_k + S_j}{\|\mathbf{c}_k - \mathbf{c}_j\|}. \tag{3.9}$$

Finally, the DB is defined as:

$$DB = \frac{1}{K}\sum_{k=1}^{K} R_k. \tag{3.10}$$

In the numerical experiments, an iteration process stops if the iteration step reaches a given maximum number (T).

3.3. Main steps of ICKM

The process of ICKM listed is as follows:

Algorithm 1 Main steps of ICKM

```
1: Initialization: Set the population size $N$, the maximum number of iterations $T$, the mutation probability $P$ and the number of clusters $K$. Let $t = 0$.
2: Perform a clone operation based on the fitness value of the antibodies and get a new population denoted by $P^{'}(t)$ .
3: Clone mutation: each bit of selected antibody will be changed to a random number between 1 and $K$, each antibody of population will be chosen with the probability $P$, then, breed a new population denoted by $P^{''}(t)$.
4: Clone selection: The same number of antibodies are selected according to (6,7) and breed the next generation denoted by $P^{'''}(t)$.
5: Perform K-means on the $P^{'''}(t)$ to get the next generation population denoted by $P(t+1)$.
6: Elitist operation: Choose the best antibody from $P(t+1)$ and compare it with $p^*(t)$ to get $p^*(t+1)$. If affinity of the new antibody is better than that of original value, then the new antibody is stored in the place of the original one, otherwise the old antibody is kept in population.
7: Stop if the termination criterion (see the end of Section 3.2) is reached, otherwise go to 2 with $t \leftarrow t+1$.
```

4. Numerical experiments

4.1. SSE, XB and DB performances

The experiments are conducted in two aspects:

1) the performances of the algorithms in terms of Mongolian printed prototype similar elements and non-prototype similar elements based on the SSE, DB and XB.

2) the performances of the algorithms in terms of the handwritten Mongolian prototype similar elements and non-prototype similar elements based on the SSE, DB and XB.

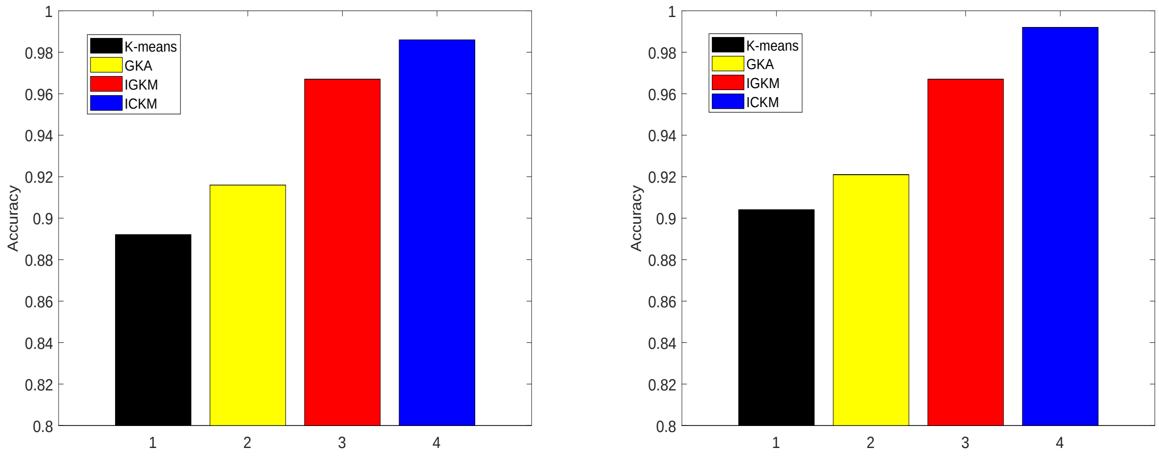
It is noteworthy that the sizes and shapes of the hand-writing elements are different for each person, even the hand-writing elements of the same person may have different sizes and shapes. However, on the premise of a fixed font and size, the same printed character has nothing to do with the typist.

Each algorithm is repeated fifty times on the four different elements sets. The averages over the fifty repeated results for the three evaluation criteria (SSE, DB and XB) are listed in Table 1. As shown in Table 1, ICKM achieves the lowest SSE, DB and XB over all the four element sets. Figure 1 also illustrate that ICKM achieves the highest accuracy for all the four element sets.

As an example to describe the clustering process, a typical iteration process on printed prototype set is shown in Figure 3 (a)–(c).

Syllables sets	Evaluation technique	K-means	GKA	IGKM	ICKM
handwritten-prototype	SSE	3.5320	3.4140	3.3720	3.320
	XB	9.631×10^{-2}	9.505×10^{-2}	9.418×10^{-2}	9.269×10^{-2}
	DB	1.6843	1.5265	1.4022	1.3901
handwritten-non-prototype	SSE	3.4960	3.3880	3.3720	3.200
	XB	9.567×10^{-2}	9.501×10^{-2}	9.411×10^{-2}	9.260×10^{-2}
	DB	1.6433	1.4653	1.3854	1.3509
printed-prototype	SSE	3.3560	3.2540	3.1980	3.1540
	XB	9.560×10^{-2}	9.499×10^{-2}	9.401×10^{-2}	9.230×10^{-2}
	DB	1.6329	1.4521	1.3708	1.3651
printed-non-prototype	SSE	3.3360	3.2100	3.1720	3.1000
	XB	9.458×10^{-2}	9.446×10^{-2}	9.342×10^{-2}	9.182×10^{-2}
	DB	1.6099	1.4387	1.3466	1.3256

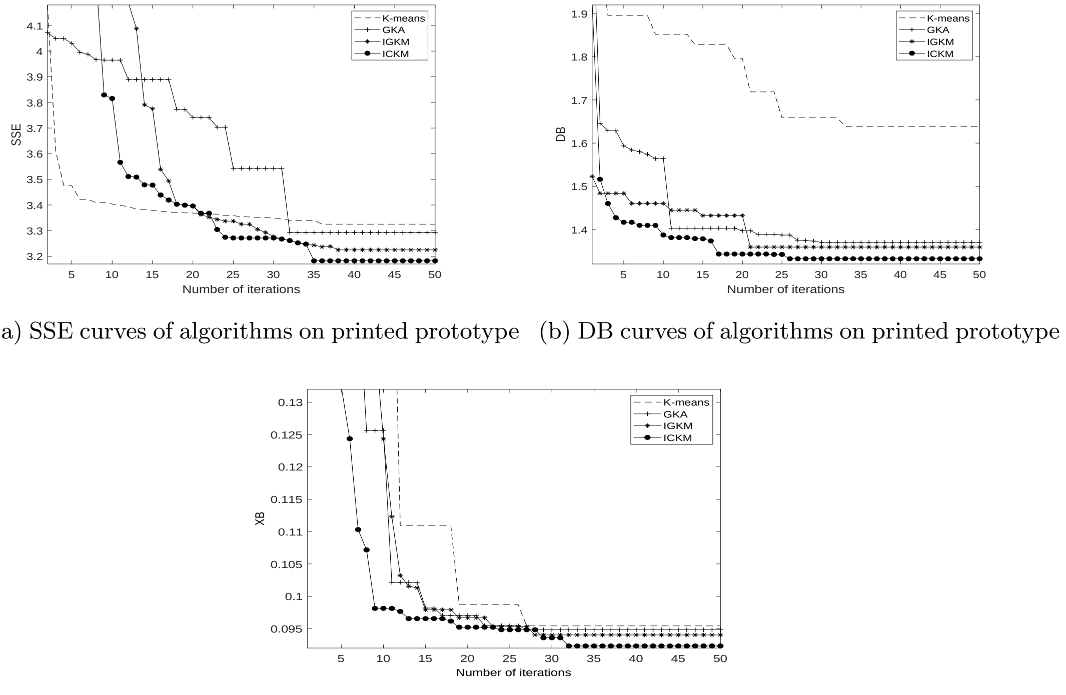
Table 1 Average SSE, XB and DB



(a) Accuracy of handwritten elements

(b) Accuracy of printed elements

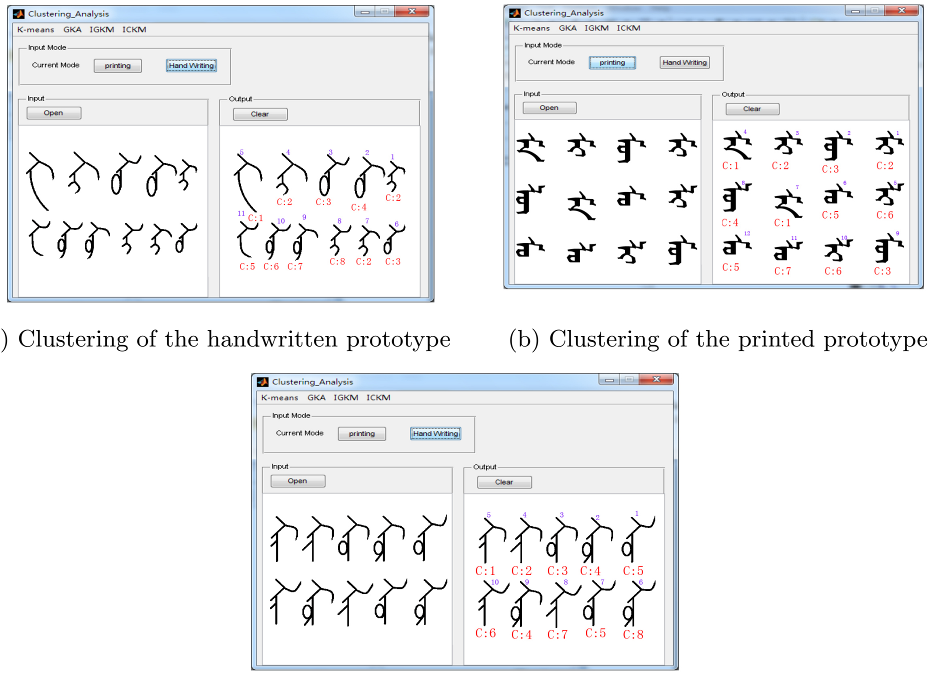
Figure 1 Results of clustering accuracy



(a) SSE curves of algorithms on printed prototype (b) DB curves of algorithms on printed prototype

(c) XB curves of algorithms on printed prototype

Figure 2 Curves of three evaluation criteria on printed prototype



(a) Clustering of the handwritten prototype (b) Clustering of the printed prototype

(c) Clustering of handwritten non-prototype

Figure 3 Clustering results of different sets

4.2. Computational time

Table 2 gives the average computational times over the fifty runs for each data set. It shows that the computational time of four algorithms does not show significant difference. However, our ICKM is better if we do not mind the computational time and we care very much about the accuracy.

Data Sets	**Evaluation Technique**	**K-means**	**GKA**	**IGKM**	**ICKM**
handwritten-prototype	SSE	8.325	8.237	8.785	8.226
	DB	9.328	9.542	9.875	9.937
	XB	8.547	8.763	8.432	8.229
handwritten-non-prototype	SSE	8.230	8.458	8.560	8.334
	DB	9.543	9.354	9.642	9.662
	XB	8.349	8.432	8.326	8.356
printed-prototype	SSE	8.242	8.286	8.432	8.333
	DB	9.135	9.230	9.426	9.424
	XB	8.365	8.231	8.256	8.234
printed-non-prototype	SSE	8.184	8.297	8.398	8.352
	DB	9.208	9.337	9.454	9.095
	XB	8.189	8.235	8.279	8.232

Table 2 Average running times (in seconds)

5. Conclusion

Mongolian information processing is a new research field and one of the most popular research concerned in this filed is the clustering of Mongolian elements with the help of artificial intelligence and image processing technique. In this paper, we proposed the ICKM clustering technique that incorporates the immune clone algorithm. The comparative experiments with K-means, G-KA and IGKM are conducted to solve two problems: the printed Mongolian elements and the hand-writing Mongolian elements. The experimental results illustrate the superiority of our algorithm over three kinds of evaluation criterions: sum of squared error (SSE), Davies-Bouldin index (DB) and Xie-Beni index (XB).

References

[1] Liang BAI, Jiye LIANG, Chuangyin DANG. *An initialization method to simultaneously find initial cluster centers and the number of clusters for clustering categorical data*. Knowl. Based Syst., 2011, **24**: 785–795.

[2] Zhihua DU, Yiwei WANG. *PK-means: a new algorithm for gene clustering*. Comput. Biol. Chem., 2008, **32**: 243–247.

[3] J. A. HARTIGAN, M. A. WONG. *A K-means clustering algorithm*. J. Royal Statis. Soc., 1979, **28**: 100–108.

[4] A. K. JAIN. *Data clustering: 50 years beyond K-means*. Pattern Recogn Lett., 2010, **31**: 651–666.

[5] A. AHMAD, L. DEY. *A K-means clustering algorithm for mixed numeric and categorical data*. Data Knowl. Eng., 2007, **63**: 503–527.

[6] J. C. BEZDEK, R. EHRLICH, W. FULL. *FCM: the fuzzy C-means clustering algorithm*. Computers and Geosciences, 1984, **28**: 191–203.

[7] M. LEE, W. PEDRYCZ. *The fuzzy C-means algorithm with fuzzy p-mode prototypes for clustering objects having mixed features*. Fuzzy Sets and Systems, 2009, **160**(24): 3590–3600.

[8] Xiaowei YANG, Guangquan ZHANG, Jie LU, et al. *A kernel fuzzy C-means clustering based fuzzy support vector machine algorithm for classification problems with outliers or noises*. IEEE Trans. Fuzzy Syst., 2011, **19**(1): 105–115.

[9] X. L. XIE, G. BENI. *A validity measure for fuzzy clustering*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1991, **13**: 841–847.

[10] Zhihua DU, Feng LIN. *A novel parallelization approach for Hierarchical clustering*. Parallel Comput., 2005, **31**: 523–527.

[11] U. MAULIK. *Genetic algorithm based clustering technique*. Pattern Recogn., 2000, **33**: 1455–1465.

[12] M. D. ANISUR, M. D. RAHMAN. *A hybrid clustering technique combining a novel genetic algorithm with K-means*. Knowl. Based Syst., 2014, **71**: 345–365.

[13] Y. K. LAM, P. W. TSANG. *explotatory K-means: A new simple and efficient algorithm for gene clustering*. Appl. Soft Comput., 2012, **12**: 1149–1157.

[14] Chun HUA, Chunying CHENG. *An Immune Genetic K-means Algorithm for Mongolian Elements Clustering*. ISNN 2018: International Symposium on Neural Networks, Minsk, Belarus, 25-28, Jun, 2018, **24**: 273-278, doi: LNCS 10878.

[15] L. N. D. CASTRO, J. I. TIMMIS. *Artificial immune systems as a novel soft computing paradigm*. Soft Comput., 2003, **7**: 526–544.

[16] L. P. KHOO, D. ALISANTOSO. *Line balancing of PCB assembly line using immune algorithms*. Eng. Comput., 2003, **19**: 92–100.

[17] G. C. LUH, C. H. CHUEH. *Multi-objective optimal design of truss structure with immune algorithm*. Comput. Struct., 2004, **82**: 829–844.

[18] Haifeng DU, Moaguo GONG, Licheng JIAO. *A novel algorithm of artificial immune system for high-dimensional function numerical optimization*. Prog. Nat. Sci., 2005, **15**: 463–471.

[19] Ruochen LIU, Licheng JIAO. *Gene transposon based clone selection algorithm for automatic clustering*. Inform. Sciences, 2012, **204**: 1–22.

[20] D. L. DAVIES, D. W. BOULDIN. *A cluster separation measure*. IEEE Trans. Pattern Anal. Mach. Intell., 1979, **PAMI-1**: 224–227.