Hindawi
Scientific Programming
Volume 2018, Article ID 2764016, 7 pages
https://doi.org/10.1155/2018/2764016

Research Article

Clustering for Probability Density Functions by New k-Medoids Method

D. Ho-Kieu,[1,2] T. Vo-Van,[3] and T. Nguyen-Trang[1,2]

[1]*Division of Computational Mathematics and Engineering, Institute for Computational Science, Ton Duc Thang University, Ho Chi Minh City, Vietnam*
[2]*Faculty of Mathematics and Statistics, Ton Duc Thang University, Ho Chi Minh City, Vietnam*
[3]*Natural Science College, Can Tho University, Can Tho City, Vietnam*

Correspondence should be addressed to T. Nguyen-Trang; nguyentrangthao@tdt.edu.vn

Received 24 November 2017; Revised 21 March 2018; Accepted 3 April 2018; Published 9 May 2018

Academic Editor: Emiliano Tramontana

This paper proposes a novel and efficient clustering algorithm for probability density functions based on k-medoids. Further, a scheme used for selecting the powerful initial medoids is suggested, which speeds up the computational time significantly. Also, a general proof for convergence of the proposed algorithm is presented. The effectiveness and feasibility of the proposed algorithm are verified and compared with various existing algorithms through both artificial and real datasets in terms of adjusted Rand index, computational time, and iteration number. The numerical results reveal an outstanding performance of the proposed algorithm as well as its potential applications in real life.

1. Introduction

Clustering plays a pivotal role in exploring the intrinsic structure of data, especially in data mining. Its main idea is to separate subgroups from an initial group such that objects in each subgroup have the most similarity. Therefore, it aims to minimize intracluster variation and to maximize the intercluster variation [1]. Cluster analysis is divided into two kinds: hard (crisp) clustering and soft (fuzzy) clustering [2]. For crisp clustering, k-means and k-medoids algorithms are the typical ones [1].

The primary difference of these algorithms is the way to approach center of cluster. For each iteration, k-means updates its center by average of mass for each cluster called centroid. However, by this approach, k-means is well-known to be sensitive to outlier despite efficiency in computational time. To overcome this shortcoming, k-medoids clustering (KMC) is a good solution because this technique employs object in the initial input being the reference point instead of center of mass [2]. That is the reason why its centers are named medoids. Among numerous KMC algorithms, the partition around medoids (PAM) firstly proposed by [3] is known to be the most powerful. However, computational time is still a drawback of PAM when it is applied to solve large problems [4]. Therefore, in this paper, one robust but straightforward scheme is employed to address the aforementioned difficulty. This scheme which is inspired from [5] intends to discover the most middle objects to be initial medoids.

Dating back to the history, the common object of clustering is usually discrete elements with a lot of works having been done like [6–11]. Nevertheless, with the fluctuation of data nowadays, it seems more proper to feature the data by series of numbers or functions rather than just a single point. This leads to considering the probability density functions (pdfs) as other object in clustering besides the discrete element [12]. So far, some of the state-of-the-art works related to clustering for pdfs can be mentioned as follows: Chen and Hung proposed a simple but effective automatic clustering algorithm for pdfs based on ad hoc technique [13]. Besides, Nguyentrang and Vovan considered many approaches to clustering problem both in the hierarchical and nonhierarchical ways [12]. Among them, a remarkable work related to k-means for pdfs called nonhierarchical method is

proposed. Furthermore, Tai et al. also applied an evolutionary technique to optimize the clustering solution [14].

Nevertheless, from an overview of the related works to clustering for pdfs, it is noticed that there is no research studying KMC for pdfs. Also, for a massive amount of data as pdfs, the computational time should be taken into consideration. Therefore, on the one hand, this paper proposes a KMC algorithm for pdfs (KMCF) for the first time. On the other hand, the convergence of KMCF algorithm is resolved. Many numerical examples are performed to evaluate the robustness as well as the effectiveness of proposed method. The numerical results of the KMCF algorithm are compared with that of existing ones in the literature. All results show the dominance of the proposed method from the perspectives of both accuracy and computational time.

The remaining part of the paper is organized as follows. Section 2 presents some related theories and proposes an algorithm for clustering of pdfs based on k-medoids method. Section 3 proves the convergence of the proposed algorithm. Section 4 discusses the numerical results of the proposed algorithm and existing ones. Section 5 gives conclusion of the whole work.

2. Related Theory and the Proposed Algorithm

2.1. Definitions. Let H be set including m probability density functions (pdfs) $H = \{f_1, f_2, \ldots, f_m\}, m > 2$ which is divided into k partitions ($2 \le k \le m$). One feasibility partition of all given pdfs in each cluster denoted as $\mathbf{W} = [w_{ij}]_{m\times k}$ should maintain the following properties:

(i) The minimum number of objects in one cluster is 1.

(ii) Each object definitely belongs to one cluster.

(iii) There is no common object between two clusters.

According to [15], the clustering problem is NP-hard when the number of clusters exceeds 3. In the case of the KMCF problem, the representing here-called medoids are objects in the initial input. Therefore, the set of the representing pdfs is defined as $F' = \{f_1', f_2', \ldots, f_k'\}$ and $F' \subset H$ as a result. For more details, one example will be given.

Suppose that we have 4 pdfs estimated from initial dataset. These pdfs are partitioned into 2 clusters, C_1 and C_2. By some techniques, the clustering result is $C_1 = \{f_1, f_2\}$ and $C_2 = \{f_3, f_4\}$, where f_1 and f_4 are, respectively, the medoids of C_1 and C_2. Then, the partition matrix is presented as follows:

$$\mathbf{W} = \begin{array}{c} \\ f_1 \\ f_2 \\ f_3 \\ f_4 \end{array} \begin{array}{cc} C_1 & C_2 \\ 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{array} = \begin{bmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \\ w_{31} & w_{32} \\ w_{41} & w_{42} \end{bmatrix}. \tag{1}$$

Therefore the set of the medoids is $F' = \{f_1', f_2'\} = \{f_1, f_4\}$.

2.2. L^1-Distance. Addressing one clustering problem requires determining the similarity between elements or pdfs before grouping. This mission can be handled by certain criteria such as distance, density, or shape [16]. In the field of clustering for pdfs, the L^1-distance firstly proposed by Pham-Gia et al. [17] is one of the most common criteria being used to evaluate the similarity between pdfs. The main technique is that this distance is primarily based on the maximum function to assess the level of proximity or separation between pdfs, which achieves many advantages as discussed in [18]. The definition of L^1-distance is stated as follows.

Definition 1. Let H be a set of m pdfs $H = \{f_1, f_2, \ldots, f_m\}$, $m > 2$, and $f_{\max} = \max\{f_1, f_2, \ldots, f_m\}$; then L^1-distance is defined by

$$\|f_1, f_2, \ldots, f_m\|_1 = \int_{R^n} f_{\max}(x)\, dx - 1. \tag{2}$$

For $m = 2$,

$$\|f_1, f_2\|_1 = \int_{R^n} |f_1(x) - f_2(x)|\, dx. \tag{3}$$

From (2), it is easy to show that $\|f_1, f_2, \ldots, f_m\|_1$ is a nondecreasing function in m with $0 \le \|f_1, f_2, \ldots, f_m\|_1 \le m - 1$. From (3), we obtain

$$\|f_1, f_2\|_1 = 2\left(\int_{R^n} f_{\max}(x)\, dx - 1\right). \tag{4}$$

2.3. The Proposed Algorithm

Problem. Given m pdfs $H = \{f_1, f_2, \ldots, f_m\}$ which are clustered into k partitions ($2 \le k \le m$), the mathematical program is considered as follows:

P: minimize $g(\mathbf{W}, F') = \sum_{j=1}^{k} \sum_{i=1}^{m} w_{ij} D(f_i, f_j')$

subject to $\sum_{j=1}^{k} w_{ij} = 1,\ j = \overline{1,m}$,

$$\sum_{j=1}^{k} w_{ij} = 1, \quad j = \overline{1,m}, \tag{5}$$

where

1. $\mathbf{W}$ and F' are defined in Section 2.1,
2. $D(f_i, f_j')$, $i = \overline{1,m}$, $j = \overline{1,k}$, is a measure for similarity between f_i and f_j'. In this paper, L^1-distance is chosen to calculate $D(f_i, f_j')$ [18].

We see that the problem P is a nonconvex program where a local minimum point does not need to be a global minimum. Based on the above denotations, the proposed k-medoids clustering algorithm for pdfs (KMCF) is presented as follows.

Step 1 (choose the initial medoids).

1. Calculate the distance between every pair of all objects based on L^1-distance, denoting $d_{ih} = \|f_i, f_h\|_1$, $i = \overline{1,m}$, $h = \overline{1,m}$.

2. Compute v_j for object j as follows:

$$v_j = \sum_{i=1}^{m} \frac{d_{ih}}{\sum_{l=1}^{m} d_{il}}, \quad h = \overline{1,m}. \tag{6}$$

3. Sort v_j in ascending order. Select first k objects having the smallest values v_j as the initial medoids. Then we have k initial medoids $f_j^{\prime(l)}$, $j = \overline{1,k}$ ($f_j^{\prime(l)}$ is the jth cluster center at the lth iteration).
4. Assign each object f_i, $i = \overline{1,m}$, to the nearest medoid which is equivalent to fixing the values of w_{ij}. Set $l = 1$.
5. Figure sum of distances from all objects to their medoids $g(\mathbf{W}, F')$.

Step 2 (update medoids).

1. In each initial established cluster, find a new medoid which is minimizing $g(*, F')$. Set $l = l + 1$.
2. Update the current medoids $f_j^{\prime(l)}$ in each cluster by replacing the new medoids $f_j^{\prime(l+1)}$.

Step 3 (assign object to their medoids).

Assign each object f_i, $i = \overline{1,m}$, to the nearest center which is equivalent to fixing the values of w_{ij}.

Compute the sum of distances from all objects to their new medoids $f_j^{\prime(l+1)}$.

If $f_j^{\prime(l)} = f_j^{\prime(l-1)}$, $j = \overline{1,k}$, or $g(*, F'^{(l)}) = g(*, F'^{(l-1)})$, then the algorithm stops. Otherwise it goes to Step 2.

By the above proposed scheme in Step 1, the distance matrix is just computed one time. Moreover, the method tends to select the k most middle objects as the initial medoids. As a result, this improves computational time significantly.

3. Convergence of the Proposed Algorithm

3.1. The Properties of Problem P. First, we defined the reduced objective function of the problem P as follows:

$F(\mathbf{W}) = \min\{g(\mathbf{W}, F')\}$ and $\mathbf{W}$ is any $m \times k$ matrix.

Lemma 2. *The reduced objective function F is a concave function.*

Proof. Consider two points $\mathbf{W^1}$ and $\mathbf{W}^2$ and let γ be any scalar so that $0 \le \gamma \le 1$; then

$$\begin{aligned} &F\left(\gamma \mathbf{W^1} + (1-\gamma)\mathbf{W}^2\right) \\ &\quad = \min\left\{g\left(\gamma \mathbf{W^1} + (1-\gamma)\mathbf{W}^2\right), F'\right\} \\ &\quad = \min\left\{\gamma g\left(\mathbf{W^1}, F'\right) + (1-\gamma)\, g\left(\mathbf{W}^2, F'\right)\right\} \\ &\quad = \min\left\{\gamma g\left(\mathbf{W^1}, F'\right) + (1-\gamma)\, g\left(\mathbf{W}^2, F'\right)\right\} \\ &\quad \ge \gamma \min\left\{g\left(\mathbf{W^1}, F'\right)\right\} + (1-\gamma)\min\left\{g\left(\mathbf{W}^2, F'\right)\right\} \\ &\quad = \gamma F\left(\mathbf{W^1}\right) + (1-\gamma)\, F\left(\mathbf{W}^2\right). \end{aligned} \tag{7}$$

Therefore, F is concave. Next, we show an important property of the constrain set (5). □

Lemma 3. *Consider a set S given by*

$$S = \left\{ \sum_{j=1}^{k} w_{ij} = 1,\ i = \overline{1,m},\ w_{ij} \ge 0,\ i = \overline{1,m},\ j = \overline{1,k} \right\}. \tag{8}$$

The extreme points of S satisfy constraint (5).

Proof. For visualization of Lemma 3 proof, we suppose that $k = 3$ and the probability of f_1 belonging to 3 clusters is 0.8, 0.1, and 0.1, respectively. Then, the pdf f_1 will be assigned to the first cluster due to the highest probability. Thus, 0.9 is one of the extremes of S corresponding to pdf f_1. Moreover, this extreme point will establish a basis as $\{1, 0, 0\}$. Also, it is an identity matrix. Each basic variable will receive value 1 and value 0 and vice versa. This completes the proof. Therefore, we have following definition. □

Definition 4. The reduced problem RP of the problem P is given as follows:

minimize $F(\mathbf{W})$ subject to $\mathbf{W} \in S$.

As the function g is concave, there exists an extreme solution of the problem **RP** which in turn satisfies the constrain set (2). Therefore, the following statement is given immediately.

Lemma 5. *Problems RP and P are equivalent.*

3.2. The Convergence of KMCF Algorithm. A point $(\mathbf{W}^*, F'^*)$ is called the partial optimal solution of problem P if it satisfies [19]

1. $g(\mathbf{W}^*, F'^*) \le g(\mathbf{W}, F'^*)$, $\forall \mathbf{W} \in S$.
2. $g(\mathbf{W}^*, F'^*) \le g(\mathbf{W}^*, F')$.

Thus, the following two problems are defined in order to receive the partial optimal solution.

Problem $\mathbf{P_1}$. Given $\widehat{F'}$, minimize $g(\mathbf{W}, \widehat{F'})$ subject to $\mathbf{W} \in S$.

Problem $\mathbf{P_2}$. Given $\widehat{\mathbf{W}} \in S$, minimize $g(\widehat{\mathbf{W}}, F')$ subject to F'.

Then, the below algorithm generates the partial optimal solutions. Then, it is essential to restate the KMCF algorithm. Since the step to find v_j for the object j is similar, so it will not be shown here.

The Restated KMCF

1. Choose initial medoids based on values of v_j; we get $F'^{(0)}$; solve $\mathbf{P_1}$ with $F' = F'^{(0)}$; then one gets that

$\mathbf{W}^{(0)}$ is an optimal basic solution of problem $\mathbf{P_1}$. Set $r = 0$. Denote $f_j^{\prime(r)}$ as the jth cluster center at the rth iteration.

2. Solve $\mathbf{P_2}$ with $\widehat{\mathbf{W}} = \mathbf{W}^{(r)}$. Let the solution be $F^{\prime(r+1)}$. If $g(\widehat{\mathbf{W}}, F^{\prime(r+1)}) = g(\widehat{\mathbf{W}}, F^{\prime(r)})$ stop, then the optimal solution is $(\mathbf{W}^*, F^{\prime *}) = (\widehat{\mathbf{W}}, F^{\prime(r+1)})$. Otherwise, go to step (3).

3. Solve $\mathbf{P_1}$ with $\widehat{F'} = F^{\prime(r+1)}$; then the basic solution will be $\mathbf{W}^{(r+1)}$ if $g(\mathbf{W}^{(r+1)}, \widehat{F'}) = g(\mathbf{W}^{(r)}, \widehat{F'})$ and stop. The optimal solution is $(\mathbf{W}^*, F^{\prime *}) = (\mathbf{W}^{(r+1)}, \widehat{F}')$; otherwise the algorithm comes back to step (2).

Theorem 6. *Algorithm restated KMCF converges to a partial optimal solution of problem* $\mathbf{P}$ *in a finite number of iterations.*

Proof. First we show that an extreme point of S is visited at most once by the algorithm before it stops. We will assume that this is not true; that is, $\mathbf{W}^{(r_1)} = \mathbf{W}^{(r_2)}$ for some r_1, r_2, where $r_1 \neq r_2$. When applying step (ii), we get two optimal solutions $F^{\prime(r_1+1)}$ and $F^{\prime(r_2+1)}$ for $\mathbf{W} = \mathbf{W}^{(r_1)}$ and $\mathbf{W} = \mathbf{W}^{(r_2)}$, respectively; that is,

$$g\left(\mathbf{W}^{(r_1)}, F^{\prime(r_1+1)}\right) = g\left(\mathbf{W}^{(r_2)}, F^{\prime(r_1+1)}\right) = g\left(\mathbf{W}^{(r_2)}, F^{\prime(r_2+1)}\right) \quad \text{since } \mathbf{W}^{(r_1)} = \mathbf{W}^{(r_2)}. \tag{9}$$

However, the sequence $g(*, *)$ generated by the algorithm is strictly decreasing. That means (9) is false. Therefore, an extreme point of S is visited at most once by the algorithm before it stops. Moreover, because there are a finite number of extreme points of S, the algorithm will reach the partial optimal solution after a limited number of iterations. Therefore, this guarantees the convergence of k-medoids type algorithms in general.

It is certain that the expected value of ARI for random partitions is zero. Anyway, it still has value 1 for perfect agreement between two partitions. Therefore, the ARI will be used in this paper for evaluating the results of the clustering algorithm. □

4. Numerical Results

In this section, four datasets are set up to evaluate performance of the proposed algorithm. The first two sets are the simulated data which are already published in [13, 18]. The third one is taken from the well-known dataset called CUReT which is available at http://www1.cs.columbia.edu/CAVE//software/curet. The final one is a real data extracted from a video of traffic situation at Ton Duc Thang University in Vietnam at the fixed moment. Besides, three other algorithms are also taken into account to make a comparison with the proposed algorithm. First is the proposed algorithm with medoids chosen randomly, namely, random k-medoids algorithm. Another one is the modification of k-means for pdfs called nonhierarchical approach [20]. The last one is

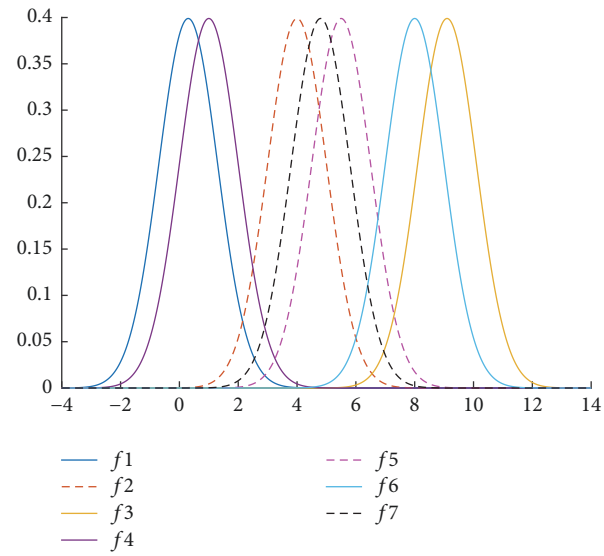


Figure 1: Pdfs of seven univariate normal distribution functions.

one of the state-of-the-art algorithms for pdfs, namely, self-update or briefly SU. All the compared algorithms will be given the suitable number of clusters in advance, except for SU. For the terminate condition, epsilon is 10^{-3} in case of SU; distance-based criteria will be employed for the remaining cases. Further, to test the stability, each algorithm is executed over independent 50 runs for every dataset and the average result is obtained as the final result. The performance of all algorithms is evaluated on three aspects: accuracy (ARI) [21], computational time (seconds), and iteration number. Further, we would like to point out that all the numerical results are developed in 2015-version Matlab software on an Intel (R) Core (TM) i3-4005U CPU @ 1.70 GHz with 4 GB main memory in Windows Server 2010 environment.

Example 1. In this example, the dataset is a kind of simple simulated data with "well-behaved" class structure and also well-studied in previous algorithms in field of clustering for pdfs. This data includes seven univariate normal distributed pdfs as presented in Figure 1. The details of the estimated parameters can be found in [18]. From Figure 1, one can receive the appropriate partition corresponding to three clusters as

$$C_1 = \{f_1, f_4\}, \quad C_2 = \{f_2, f_5, f_7\}, \quad C_3 = \{f_3, f_6\}. \tag{10}$$

The clustering result of all compared algorithms is listed in Table 1. It is obvious that, concerning the accuracy, the proposed algorithm and the SU achieve the absolute results with ARI 1, followed by the random k-medoids and the nonhierarchical approach, respectively. Regarding the computational time and iteration number, the proposed algorithm ranks first on a list of four algorithms. Although both proposed algorithm and SU obtain good results in accuracy,

Table 1: Comparison of algorithms in example 1 (7 normal pdfs).

Algorithm	Average adjusted Rand index	Average computational time (seconds)	Average iteration number
Proposed k-medoids	*1.00*	*0.007*	*1*
Random k-medoids	0.79	0.004	1
Nonhierarchical approach	0.49	0.004	2
SU	1.00	0.258	7

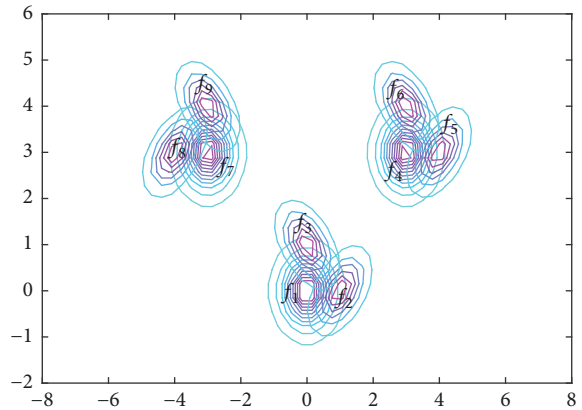


Figure 2: Contour of nine bivariate t distribution functions.

the proposed algorithm is still far superior to the SU method in the computational time. Therefore, it would be concluded that the proposed algorithm performs best in the first dataset.

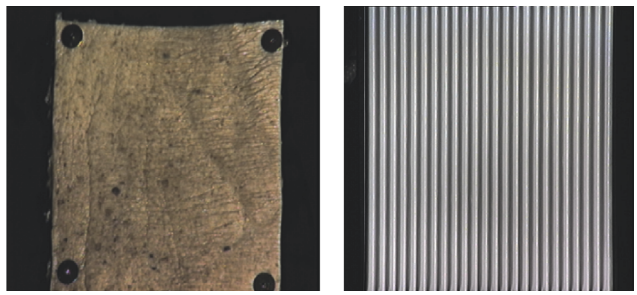
Example 2. In this example, the considered dataset is more complex due to a greater number of pdfs in two-dimensional space. Hence, a prediction in increasing the computational time is also considered. For more details, the data contains nine pdfs estimated by the bivariate t distribution with v degrees of freedom as described in [13]. All pdfs are shown in Figure 2. From the figure, one may find that the appropriate number of clusters is 3 and the corresponding partition is

$$\begin{aligned} C_1 &= \{f_1, f_2, f_3\}, \\ C_2 &= \{f_4, f_5, f_6\}, \\ C_3 &= \{f_7, f_8, f_9\}. \end{aligned} \tag{11}$$

The result of the performance of all algorithms is demonstrated in Table 2. It is clear that a similar trend to the first example is observed in this case in terms of ARI. Besides, concerning the computational time and iteration number, the proposed method is a bit slower than the random k-medoids algorithm and the nonhierarchical approach. More specifically, despite high precision, the comparable algorithm SU is still far inferior to the proposed algorithm regarding the computational time. So far, considering accuracy and computational pace, the proposed algorithm can be seen as the best candidate through the first two examples.

Table 2: Comparison of algorithms in example 2 (9 bivariate student pdfs).

Algorithm	Average adjusted Rand index	Average computational time (seconds)	Average iteration number
Proposed k-medoids	*1.00*	*0.026*	*1*
Random k-medoids	0.78	0.016	1
Nonhierarchical approach	0.32	0.02	2
SU	1.00	5.578	9



Human skin Ribbed paper

Figure 3: Two different samples in CUReT dataset.

Table 3: Comparison of all algorithms in example 3 (114 CUReT images).

Algorithm	Average adjusted Rand index	Average computational time (seconds)	Average iteration number
Proposed k-medoids	*1.00*	*0.254*	*1*
Random k-medoids	0.90	0.105	1
Nonhierarchical approach	0.10	0.368	33
SU	1.00	1.415	7

Example 3. In this example, we employ the image objects with large quantity to measure the robustness of the proposed algorithm. The dataset includes 114 texture images of size 640 × 480 pixels taken from the CUReT database. These objects are divided into two categories: 57 samples each of human skin and ribbed paper as demonstrated in Figure 3. Subsequently, Figure 4 illustrates the estimated pdfs of these images. It seems that this case is more complicated to cluster due to the significant overlapping area and numerous pdfs. The nominal partition is given as

$$\begin{aligned} C_1 &= \{f_1, f_2, \ldots, f_{57}\}, \\ C_2 &= \{f_{58}, f_{59}, \ldots, f_{114}\}. \end{aligned} \tag{12}$$

From what has been derived from Table 3, it is clear that there is no change in the order of algorithms with regard to the value of ARI. In aspects of computational time, the

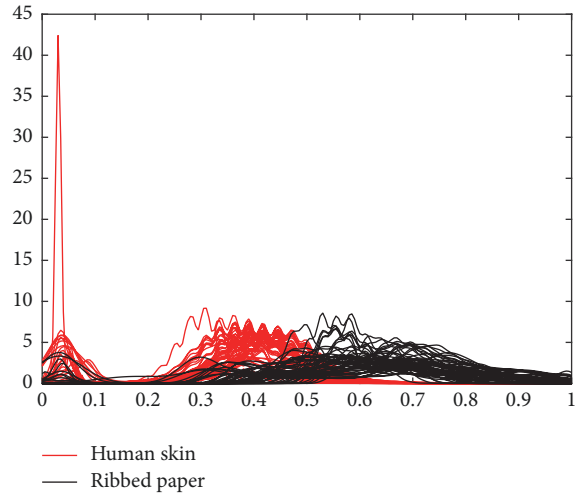


Figure 4: Pdfs of 114 images consisting of human skin and ribbed paper in CUReT dataset.

random k-medoids algorithm consumes the least time, in contrast to SU. About number of iterations, the nonhierarchical approach runs most iteration to give the satisfied partition instead of SU as in the previous cases. Meanwhile, two remaining algorithms just need an iteration to deduce the final result. Nevertheless, considering all surveyed perspectives, a balance among them is found in the proposed algorithm rather than the others. Therefore, it can be said that the proposed algorithm has achieved an outstanding performance in this case.

Example 4. In this example, one real data is considered to apply the proposed method. As known to the world, most countries in South East Asia usually deal with the traffic congestion, including Vietnam. This problem is regularly happening in the rush hour in famous public places. To study this situation, we extract images from a short daily video taken in front of the Ton Duc Thang University, Ho Chi Minh City, Vietnam. In general, 116 images of size 1920×1080 pixels are taken into account. The no-traffic jam group includes 46 photos and the traffic jam group 70 photos. The file will be provided when having requirement. From these photos, the pdfs are estimated as shown in Figure 5. The nominal partition is $C_1 = \{f_1, f_2, \ldots, f_{46}\}, C_2 = \{f_{47}, f_{59}, \ldots, f_{116}\}$.

The result in Table 4 reveals that this dataset is not quite easy to tackle for all algorithms. The first time we see a reduction of value of ARI for all compared algorithms in this case, with an exception of the proposed algorithm. Particularly, the SU just gets ARI 0.84 instead of 1 as the previous examples. A similar trend is witnessed in ARI of random k-medoids algorithm. Due to a greater number of pdfs, the computational time and the iteration number are both increased in performance of all algorithms. Although the SU was always the most competitive algorithm before, it is defeated convincingly in the two last cases. Therefore, it can be concluded that the proposed method is quite potential in real applications.

Table 4: Comparison of all algorithms in example 4 (116 traffic images).

Algorithm	Average adjusted Rand index	Average computational time (seconds)	Average iteration number
Proposed k-medoids	*1.00*	*0.394*	*1*
Random k-medoids	0.71	0.146	1
Nonhierarchical approach	0.15	0.576	35
SU	0.84	2.485	8

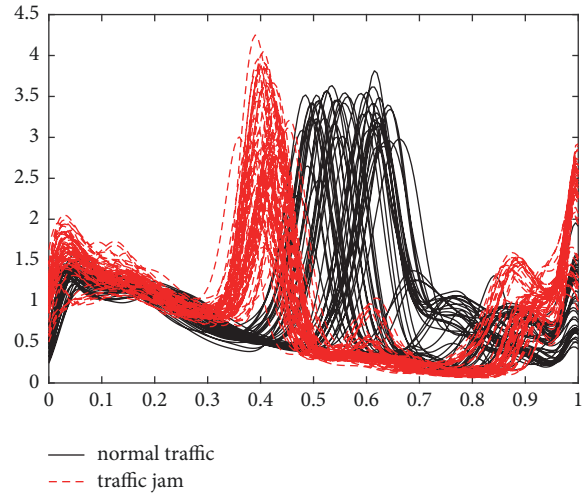


Figure 5: Estimated pdfs of 116 photos taken from the short video at Ton Duc Thang University in daily time.

Throughout 4 examples, all results of the proposed method are briefly presented in Table 5 plus its ranks regarding each criterion. Here, the rank ranges from the 1st to the 4th, which is corresponding to the total algorithms mentioned in the numerical part of the paper. From the table, it is obvious that the proposed method is almost in the first rank of accuracy of deduced partition (ARI). Meanwhile, the other algorithms do not produce the final partition as good as that of the proposed method. This not only confirms the enhancement of accuracy of the proposed method but also reveals its stability. A similar trend can be seen in the number of iterations of the proposed method. Despite some restrictions in computational time of the proposed method, it still deserves to be the best one through what was shown in all numerical examples compared with the three remaining algorithms.

5. Conclusion

In this paper, we have suggested a robust but straightforward algorithm for clustering pdfs based on k-medoids. By nature of k-medoid clustering, the proposed method expertly tackles the outlier compared with clustering algorithms using k-means technique. In addition to that, the recommended scheme speeds up the convergence of the proposed method

Table 5: Brief results of the proposed method in all examples and its ranks.

Criteria, rank	ARI		Computational time		Iteration number	
	Value	Rank	Value	Rank	Value	Rank
Example 1	1	1st	0.007	3rd	1	1st
Example 2	1	1st	0.026	3rd	1	1st
Example 3	1	1st	0.254	2nd	1	1st
Example 4	1	1st	0.394	2nd	1	1st

significantly since the distance matrix between pdfs is computed precisely one time. Besides, a general proof for convergence of the proposed method is given. Via all numerical examples, an outstanding performance of the proposed method is confirmed through the evaluation criteria. In particular, example 4 argues the potential applications of the proposed method in real life. However, the proposed algorithm only works well in the case of a number of clusters given in advance. Therefore, more future studies should be focused on automatic k-medoids clustering algorithm for pdfs.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

[1] W. Sheng and X. Liu, "A genetic k-medoids clustering algorithm," *Journal of Heuristics*, vol. 12, no. 6, pp. 447–466, 2006.

[2] J.-P. Mei and L. Chen, "Fuzzy clustering with weighted medoids for relational data," *Pattern Recognition*, vol. 43, no. 5, pp. 1964–1974, 2010.

[3] L. Kaufman and P. Rousseeuw, *Clustering by Means of Medoids*, North Holland Publishing Company, 1987.

[4] H. Miller and J. Han, "Spatial clustering methods in data mining: a survey," in *Geographic Data Mining and Knowledge Discovery*, Taylor & Francis, 2001.

[5] H.-S. Park and C.-H. Jun, "A simple and fast algorithm for K-medoids clustering," *Expert Systems with Applications*, vol. 36, no. 2, pp. 3336–3341, 2009.

[6] G. Celeux and G. Govaert, "Clustering criteria for discrete data and latent class models," *Journal of Classification: Classification Literature Automatic Search Service / plus CLASS*, vol. 8, no. 2, pp. 157–176, 1991.

[7] N. Bouguila and W. ElGuebaly, "Discrete data clustering using finite mixture models," *Pattern Recognition*, vol. 42, no. 1, pp. 33–42, 2009.

[8] Z. Izakian, M. Saadi Mesgari, and A. Abraham, "Automated clustering of trajectory data using a particle swarm optimization," *Computers, Environment and Urban Systems*, vol. 55, pp. 55–65, 2016.

[9] D. K. Panjwani and G. Healey, "Markov Random Field Models for Unsupervised Segmentation of Textured Color Images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 10, pp. 939–954, 1995.

[10] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: a review," *ACM Computing Surveys*, vol. 31, no. 3, pp. 264–323, 1999.

[11] R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan, "Automatic subspace clustering of high dimensional data for data mining applications," in *Proceedings of the 1998 ACM SIGMOD International Conference on Management of Data*, pp. 94–105, 1998.

[12] T. Nguyentrang and T. Vovan, "Fuzzy clustering of probability density functions," *Journal of Applied Statistics*, vol. 44, no. 4, pp. 583–601, 2017.

[13] J. H. Chen and W.-L. Hung, "An automatic clustering algorithm for probability density functions," *Journal of Statistical Computation and Simulation*, vol. 85, no. 15, pp. 3047–3063, 2015.

[14] V. V. Tai, N. T. Thao, and C. N. Ha, "Clustering for probability density functions based on Genetic Algorithm," in *Applied Mathematics in Engineering and Reliability, Proceedings of the 1st International Conference on Applied Mathematics in Engineering and Reliability (Ho Chi Minh City, Vietnam, May 2016)*, pp. 51–57, 2016.

[15] P. Brucker, "Optimization and Operations Research," in *Proceedings of a Workshop Held at the University of Bonn On the Complexity of Clustering Problems BT, October, 1977*, R. Henn, B. Korte, and W. Oettli, Eds., pp. 45–54, Springer, Heidelberg, Germany, 1978.

[16] O. Maimon and L. Rokach, "Clustering Methods BT," in *Data Mining and Knowledge Discovery Handbook*, O. Maimon and L. Rokach, Eds., pp. 321–352, Springer, Boston, MA, USA, 2005.

[17] T. Pham-Gia, N. Turkkan, and T. Vovan, "Statistical discrimination analysis using the maximum function," *Communications in Statistics—Simulation and Computation*, vol. 37, no. 1-2, pp. 320–336, 2008.

[18] T. Vovan, "L^1-distance and classification problem by Bayesian method," *Journal of Applied Statistics*, vol. 44, no. 3, pp. 385–401, 2017.

[19] R. E. Wendell and J. Hurter, "Minimization of a non-separable objective function subject to disjoint constraints," *Operations Research*, vol. 24, no. 4, pp. 643–657, 1976.

[20] T. V. Van and T. Pham-Gia, "Clustering probability distributions," *Journal of Applied Statistics*, vol. 37, no. 11, pp. 1891–1910, 2010.

[21] W. M. Rand, "Objective criteria for the evaluation of clustering methods," *Journal of the American Statistical Association*, vol. 66, no. 336, pp. 846–850, 1971.