Hindawi
Mathematical Problems in Engineering
Volume 2020, Article ID 5143797, 13 pages
https://doi.org/10.1155/2020/5143797

Research Article

Weighted k-Prototypes Clustering Algorithm Based on the Hybrid Dissimilarity Coefficient

Ziqi Jia[1] **and Ling Song**[2,3]

[1]*Nanfang College of Sun Yat-sen University, Guangzhou 510000, China*
[2]*School of Computer & Electronic Information, Guangxi University, Nanning 530004, China*
[3]*Guangxi Key Laboratory of Multimedia Communications and Network Technology, Nanning 530004, China*

Correspondence should be addressed to Ling Song; jqian@gxu.edu.cn

Received 25 September 2019; Accepted 17 April 2020; Published 25 July 2020

Academic Editor: Kishin Sadarangani

The k-prototypes algorithm is a hybrid clustering algorithm that can process Categorical Data and Numerical Data. In this study, the method of initial Cluster Center selection was improved and a new Hybrid Dissimilarity Coefficient was proposed. Based on the proposed Hybrid Dissimilarity Coefficient, a weighted k-prototype clustering algorithm based on the hybrid dissimilarity coefficient was proposed (WKPCA). The proposed WKPCA algorithm not only improves the selection of initial Cluster Centers, but also puts a new method to calculate the dissimilarity between data objects and Cluster Centers. The real dataset of UCI was used to test the WKPCA algorithm. Experimental results show that WKPCA algorithm is more efficient and robust than other k-prototypes algorithms.

1. Introduction

Cluster analysis belongs to unsupervised learning and is an important research direction in the field of machine learning [1]. Clustering analysis, as an important data analysis tool, can divide data objects into different subclusters by calculating the dissimilarity of data objects without marked samples. This study aimed to achieve the purpose that the data objects in the same cluster have less dissimilarity and the data objects in different clusters have more dissimilarity.

Clustering aims to find out the correlation between subclusters in datasets and to evaluate the dissimilarity among data objects in these subclusters [2]. In the field of Categorical Data clustering, the classical k-modes algorithm [3] uses the modes vector to represent the Cluster Centers. The modes vector is a combination of the eigenvalue that occurs most frequently of each feature in the subcluster. The dissimilarity between data objects to be clustered and the cluster is calculated by simple Hamming distance, and only the Categorical Data can be processed. In the field of Numerical Data clustering, the classical k-means algorithm [4] uses the means vector to represent the Cluster Centers, and the means vector is the average value of each eigenvalue in the subcluster. Euclidean distance is used to calculate the dissimilarity between the cluster and the data objects to be clustered, and only Numerical Data can be processed. Both the k-modes algorithm and the k-means algorithm can only handle a single type of data.

In the actual clustering division, people not only need to deal with the Categorical Data and Numerical Data, but also need to deal with a large number of mixed-type datasets composed of the Categorical Data and Numerical Data. Because there is a big difference between Categorical Data and Numerical Data, and mixed-type datasets are usually high dimensional, it is very complicated to deal with mixed-type data in cluster analysis. A simple method to deal with mixed-type data is data preprocessing, which directly converts the Categorical Feature into Numerical Feature. In other words, the mixed-type data are directly converted into the Numerical Data, and then the Numerical Data clustering algorithm is applied to the clustering. For example, the Categorical Feature is converted to binary string, and then

the algorithm based on Numerical Data is used to do clustering division. However, there are four disadvantages to using binary encoding for data preprocessing: (1) the original structure of the Categorical Data is destroyed, resulting in the meaningless binary features after conversion; (2) the implicit information of dissimilarity is ignored, which cannot truly reflect the structure of the dataset; (3) if the range of eigenvalues is large, the converted binary eigenvalues will have a larger dimension; and (4) maintenance is difficult, if new eigenvalues are added for the Categorical Feature, then all data objects will change [5].

To solve these problems, researchers have carried out a series of exploratory studies. k-prototypes algorithm [6] and its variant algorithm are mixed-type data clustering algorithms that take into account the Dissimilarity Coefficient of Categorical Feature and Numerical Feature at the same time. Such algorithms can process both Categorical Data and Numerical Data at the same time, but clustering parameters need to be set artificially. OCIL algorithm [7] is a hybrid clustering algorithm based on no parameters, and a uniform Dissimilarity Coefficient is given based on entropy. Like k-prototypes and its variants, the OCIL algorithm uses the k-means example to process mixed-type data. It is an iterative algorithm, sensitive to initialization, and more suitable for spherical distributed data. Ji et al. [8] improved the k-prototypes algorithm by considering the influence of feature weight on the clustering process and proposed the Dissimilarity Coefficient based on feature importance. Renato et al. [9] further improved the algorithm of Ji, endowed different features with different weights, and used Lp Distance Function as the new Dissimilarity Coefficient. Yao et al. [10] proposed an anonymous algorithm for hierarchical clustering based on k-prototypes (short name KLS). KLS algorithm improves the formula of the Dissimilarity Coefficient and unifies the weight setting of the Categorical Feature and Numerical Feature, but the weight needs to be specified in advance by experts. DPC-KNN-PCA algorithm [11] introduced the density peak algorithm into the k-prototypes algorithm to determine the initial Cluster Centers and improved the local neighborhood density ρ_i through the nearest neighbor algorithm. However, the selection of the nearest neighbor value k is easily affected by the dataset distribution. Dongwei et al. [12] proposed a k-prototypes algorithm based on the adaptive determination of the initial centroids (short name KP-ADIC). The KP-ADIC algorithm can determine the initial Cluster Centers adaptively, but its Dissimilarity Coefficient cannot fully calculate the dissimilarity between the data. Sangam et al. [13] proposed an equi-biased k-prototypes algorithm for clustering mixed-type data (short name EKACMD) in 2018. EKACMD algorithm is a variant algorithm of k-prototypes, which improves the Dissimilarity Coefficient by considering the relative frequency and distribution of each Categorical Feature. EKACMD algorithm can fully consider the structural characteristics of data in some cases and improve the clustering accuracy, but it is still not applicable in the case that the occurrence frequency of each eigenvalue of the Categorical Data is equal. Cui et al. [14] applied rough sets to k-prototypes algorithm and proposed RS-KP algorithm, which used rough sets to calculate the dissimilarity between eigenvalues. Although the RS-KP algorithm can deal with the outliers in the clustering of mixed-type data, it is difficult to cluster the discretized data when the eigenvalue ranges overlap, that is, the clustering results of the RS-AP algorithm are easily affected by the discretization.

In this paper, k-prototypes algorithm and its variants were analyzed and compared, and the automatic determination method of initial Cluster Centers was improved, and then a new Hybrid Dissimilarity Coefficient is proposed. The value of these improvements lies in the following: (1) avoiding the randomness of the selection of the initial Cluster Centers; (2) making the clustering method more suitable for the characteristics of mixed-type data; (3) there is no need to manually set various parameters in the clustering process, such as the number of clustering k and the weight parameter γ; (4) there is no limitation on the types of clustering data. We can process Categorical Data, Numerical Data, and mixed-type data at the same time, which not only makes the clustering results more ideal, but also provides a new idea for the analysis and mining of real-world data.

The organizational structure of this paper is as follows: Section 2 introduces the symbols related to this paper. Section 3 introduces the k-prototypes algorithm. In addition, Section 4 details the design of WKPCA algorithm, and Section 5 gives the experimental results and analysis. Finally, Section 6 is a summary of this paper.

2. Symbols

Table 1 shows the symbols associated with this article.

3. The k-Prototypes Algorithm

Huang [6] proposed a k-prototypes algorithm for clustering mixed-type data, which combines the ideas of k-means algorithm [2] and k-modes algorithm [3]. The k-prototypes algorithm divides the dataset into $k(k \in N^+)$ different subclusters to minimize the value of the Cost Function. The Cost Function is shown in the following formula:

$$F(U, Q) = \sum_{l=1}^{k} \sum_{i=1}^{n} u_{il} d(x_i, q_l). \tag{1}$$

The k-prototypes algorithm combines the "means" of the numerical part and the "modes" of the categorical part to build a new hybrid Cluster Center "prototype". On the basis of "prototype," it builds a Dissimilarity Coefficient formula and the Cost Function applicable to the mixed-type data. The parameter γ is introduced to control the influence of the Categorical Feature and the Numerical Feature on the clustering process. It is assumed that the mixed-type dataset has p Numerical Feature and $m - p$ Categorical Feature. For any $x_i, q_l \in D$, the definition of the Dissimilarity Coefficient of k-prototypes is shown in the following formula:

Table 1: Symbolic description.

Symbol	Description
D	$D_{n\times m} = [x_1, x_2, \ldots, x_n]^T$ is a nonempty limited datasets, containing n data objects, and each data object is descripted by m features
k	# of clusters
x_i	i^{th} data object. For any x_i, x_j, if and only if $x_{i,s} = x_{j,s}$, then $x_i = x_j$
$x_{i,s}$	s^{th} Categorical eigenvalues of i^{th} data object; $x_{i,s} \in DOM(A_s), s = \{1, 2, \ldots, p, p+1, \ldots, m\}$ is the feature dimension of the dataset, the former p−dimension is the Categorical Feature, and the latter $m-p$−dimension is the Numerical Feature
x_i^C	$x_i^C = [x_{i,1}^C, x_{i,2}^C, \ldots, x_{i,p}^C]^T$ is Categorical Data, $x_{i,s}^C \in DOM(A_S), s \in 1, 2, \ldots, p$
x_i^N	$x_i^N = [x_{i,1}^N, x_{i,2}^N, \ldots, x_{i,m-p}^N]^T$ is Numerical Data, $x_{i,s}^N \in R, s \in p+1, p+2, \ldots, m$
A_s	Describes the feature of data object x_i
A_s^C	Describes the feature of the Categorical Data part of the mixed-type data object x_i
A_s^N	Describes the feature of the Numerical Data part of the mixed-type data object x_i
$A_{ql,s}$	s^{th} eigenvalues of Cluster Centers
$DOM(A_s)$	The eigenvalue domain of feature A_s is described by $\mathrm{DOM}(A_s)$ and represents all possible values of each feature A_s; $\mathrm{DOM}(A_1), \mathrm{DOM}(A_2), \ldots, \mathrm{DOM}(A_m)$ is the eigenvalue domain corresponding to each feature; for Numerical Feature, $\mathrm{DOM}(A_s^N) \in R$; for Categorical Feature, its value domain is limited and disordered. Suppose that the s-dimensional feature of a dataset has n_s categories, then $\mathrm{DOM}(A_s^C) = \{A_{s,1}^C, ..., A_{s,t}^C \ldots, A_{s,n_s}^C\}$
$d(\cdot,\cdot)$	Calculates the dissimilarity between data objects
C_l	The cluster of l^{th}, $C = \{C_1, \ldots, C_l, \ldots, C_k\}$ is all the clusters contained in the dataset D
q_l	The Cluster Center corresponding to the l^{th} cluster: $Q = \{q_l, l = 1, 2, .., k\}$ is the Cluster Centers set
$\lvert C_l\rvert$	# of data objects in cluster C_l, given by formula $\lvert C_l\rvert = \lvert\{i \mid u_{i,l} = 1\}\rvert$
$\lvert A_{s,t}\rvert_{C_l}$	The s-dimension feature of cluster C_l, the data object number of eigenvalue of $A_{s,t}$, given by the formula $\lvert A_{s,t}\rvert_{C_l} = \lvert x_{i,s} = q_{l,s} = A_{s,t}, u_{i,l} = 1\rvert$
$\sum_{l=1}^{k} \lvert A_{s,t}\rvert_{C_l}$	Total number of data objects in all clusters contain eigenvalues $A_{s,t}$
$P(A_{s,t}^C)$	The relative frequency in the cluster of eigenvalues
$PR(A_{s,t}^C)$	The frequency distribution between clusters of eigenvalues
EnA_s^C	Entropy of Categorical Feature
$E_n'A_s^C$	The quantized entropy of Categorical Feature
w_s	The entropy weight of Categorical Feature
$\max(A_{i,s}^N)$	The maximum values of Numerical Feature
$\min(A_{i,s}^N)$	The minimum values of Numerical Feature $A_{i,s}^N$
$A_{i,s}^N$	Quantized Numerical Feature
$d_w(x_i, q_l)$	The weighted Hybrid Dissimilarity Coefficient
$F_w(U, Q)$	Consider the Cost Function of the weight
$\overline{d}$	The average distance between two data objects
ρ_i	Local neighborhood density
d_c	Cutoff distance
L_i	Relative distance

$$d(x_i, q_l) = \gamma \sum_{s=1}^{p} \delta\left(x_{i,s}^C - q_{l,s}^C\right) + \sum_{s=p+1}^{m} \sqrt{\left(x_{i,s}^N - q_{l,s}^N\right)^2},$$

$$\text{where} \quad \delta\left(x_{i,s}, q_{l,s}\right) = \begin{cases} 0, x_{i,s} = q_{l,s}, \\ 1, x_{i,s} \neq q_{l,s}. \end{cases} \tag{2}$$

The k-prototypes algorithm divided the Dissimilarity Coefficient of the mixed-type data into two parts for separate calculation. The categorical part adopts the simple Hamming distance, and the numerical part adopts the square of the Euclidean distance [15]. The proportion of the two types of data in the Dissimilarity Coefficient was adjusted by parameter γ [12]. It is an important adjustable parameter for k-prototypes algorithm. The purpose of introducing parameter γ is to avoid the clustering result value deviation from the Categorical Feature or the Numerical Feature and control the relative weight of dissimilarity between Categorical Data and Numerical Data. When $x_{i,s}^C = q_{l,s}^C$, $\delta(x_{i,s}^C, q_{l,s}^C)$ is equal to 0; when $x_{i,s}^C \neq q_{l,s}^C$, $\delta(x_{i,s}^C, q_{l,s}^C)$ is equal to 1; the basic steps of the k-prototypes algorithm are described as follows:

Step 1: k data objects were randomly selected from dataset D as the initial Cluster Centers.

Step 2: formula (2) is used to calculate the dissimilarity between x_i and q_l. According to the calculation result, x_i is allocated to the nearest cluster.

Step 3: according to the current Cluster Centers, the dissimilarity of the data object is recalculated. Reassign the data objects to the nearest subcluster, the values with the highest frequency are used in the categorical part, and the numerical part uses the method of average value to determine. Update the Cluster Centers.

Step 4: repeat Steps 2 and 3 until the Cost Function is no longer changing. If the Cost Function is no longer changing, the algorithm ends. Otherwise, skip to Step 2 to continue.

3.1. Description of Problem. k-prototypes algorithm can cluster mixed-type data, and the principle is simple and easy to operate, but there are still some shortcomings in the clustering process: (1) The random selection of the initial Cluster Centers results in the uncertainty and randomness of the clustering results, and the number of clusters (k) should be manually determined; (2) the simple Hamming distance is used to calculate the dissimilarity between the Categorical Data and the Cluster Centers, resulting in the loss of information and the inability to objectively reflect the real situation between the data objects and the clusters, resulting in inaccurate clustering results; (3) parameter γ used to adjust the proportion between Categorical Data and Numerical Data needs to be manually determined; and (4) the structural characteristics of Categorical Data and Numerical Data and the overall distribution of datasets have not been fully considered.

3.1.1. Problem with the Dissimilarity Coefficient. With the help of the artificial dataset D_1 shown in Table 2, the disadvantages of user directive parameter γ in the clustering process are discussed. D_1 contains 27 data objects, and each data object is described by two Numerical Features and one Categorical Feature. Categorical Feature A_3 has three eigenvalues $\mathrm{DOM}(A_3) = \{\mathrm{A}, \mathrm{B}, \mathrm{C}\}$; the values of Numerical Feature A_1 and A_2 are in the range of 0–80.

About the feature of A_3, a solid triangle, a solid circle, and a solid square are prescribed to represent eigenvalue data objects A, B, C. When parameter $\gamma = 0$, clustering results of D_1 only depends on two features A_1 and A_2. The clustering results are shown in Figure 1. D_1 has three clusters. To facilitate observation, the three clusters were separated by dotted lines. When $\gamma > 0$, the data object x_7 can be moved to the C_2, because most of the feature (A_3) of the data objects is the same between the object x_7 and cluster C_2. Similarly, based on the above reasons, a data object x_{10} can be moved to cluster C_1. When the value of the parameter γ changes, the clustering results of the data objects $x7$ and $x10$ will change accordingly. The data objects $x7$ and $x10$ may be divided into cluster $C1$ or cluster $C2$. Data objects x_1 and x_{20} may remain in the original cluster because they are too far away from the other clusters nearby, even if they have the same eigenvalues as the data objects in the other clusters nearby. In summary, it is important to define the parameter γ on the same scale. For related discussion, see literature [13].

3.1.2. Problems in Initial Cluster Center Selection. The classical k-prototypes algorithm is very sensitive to the initial Cluster Centers, which are selected by random initialization method or manual setting method, both of which lead to the instability of clustering results to a certain extent. The initial Cluster Centers with different locations and k values will produce different clustering results. As shown in Figure 2, the actual cluster number of this dataset is $k = 3$. Figure 2 shows the clustering results generated by different initial Cluster Centers when the initial cluster number is set to $k = 2$, $k = 3$, and $k = 4$ (the contents described in Figure 2 from left to right are random selection of initial Cluster Centers, clustering iteration process, and final clustering result). Therefore, it is very important for the clustering algorithm to find a suitable initial Cluster Center.

4. Weighted k-Prototype Clustering Algorithm Based on Hybrid Dissimilarity Coefficient (WKPCA)

The motivation for the proposed algorithm is (1) to provide an effective method for the expression of Dissimilarity Coefficient of mixed-type data clustering and (2) to avoid the uncertainty caused by random selection of initial Cluster Centers.

In order to solve the problem of quantitative measurement of information, in 1948 Shannon cited the concept of thermal entropy in thermodynamics and proposed the concept of "Information Entropy". The occurrence probability of discrete random events is defined as Information Entropy. The size of the Information Entropy is related to the probability of random events. The smaller the probability of an event, the more information is generated and the smaller the entropy of information. For example, the Information Entropy of the event "Heavy rain in a place where it does not rain frequently" is large; the larger the probability of an event, the less information is generated, and the greater the entropy of information. For example, the event "The sun rises in the east" will definitely happen, so it has very little information. Shannon's Information Entropy formula [16] is defined as follows:

$$Enx = -\sum_{i=1}^{n} p(x_i)\log(p(x_i)),$$
$$\text{where} \quad \begin{bmatrix} D \\ p_{(x)} \end{bmatrix} = \begin{bmatrix} x_1, & x_2, & \cdots, & x_n \\ p(x_1), & p(x_2), & \cdots, & p(x_n) \end{bmatrix}. \tag{3}$$

$p(x_i) = (|x_i|/|D|), 1 < i < n$, represents the probability of a random event x_i. x_i is a divided subset of dataset D. When D satisfies the condition $p(x_1) = p(x_2) = \cdots = p(x_n)$, $En(A)$ takes the maximum value $\log|D|$; when D satisfies the condition $p(x_1) = 1, n = 1$,$En(A)$ takes the minimum value of 0. Information Entropy has the following basic properties:

Nonnegative: there is a negative sign in the Information Entropy formula, which represents the reduction or elimination of the disordered state after the system is obtained, that is, the magnitude of the uncertainty is eliminated

Symmetry: all variables of a function are interchangeable without affecting the value of the function $En(X) = En(p(x_1), p(x_2), p(x_3), \ldots, p(x_n)) = En(p(x_2), p(x_1), p(x_3), \ldots, p(x_n))$

4.1. Dissimilarity Coefficient of Categorical Based on Entropy Weight. Information Entropy can be used to calculate the discreteness of data and assign appropriate weight to each feature to improve the clustering effect. In the clustering

Table 2: Artificial dataset D_1.

Data	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}	x_{12}	x_{13}	
A_1	12	20	28	18	20	25	29	33	24	45	45	48	52	
A_2	39	36	30	52	44	41	54	46	55	59	63	70	70	
A_3	B	A	A	A	A	A	B	A	A	B	B	B	B	
Data	x_{14}	x_{15}	x_{16}	x_{17}	x_{18}	x_{19}	x_{20}	x_{21}	x_{22}	x_{23}	x_{24}	x_{25}	x_{26}	x_{27}
A_1	51	52	53	54	55	61	62	64	67	69	71	73	76	79
A_2	66	63	58	23	14	8	66	19	30	7	24	11	23	27
A_3	B	B	B	C	C	C	B	C	C	C	C	C	C	C

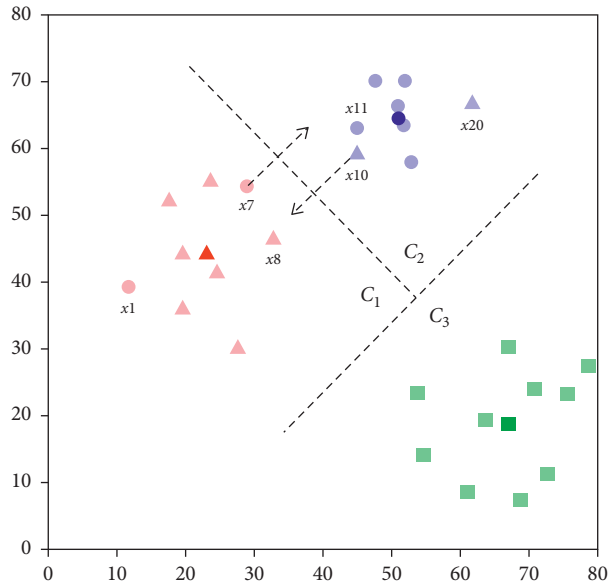
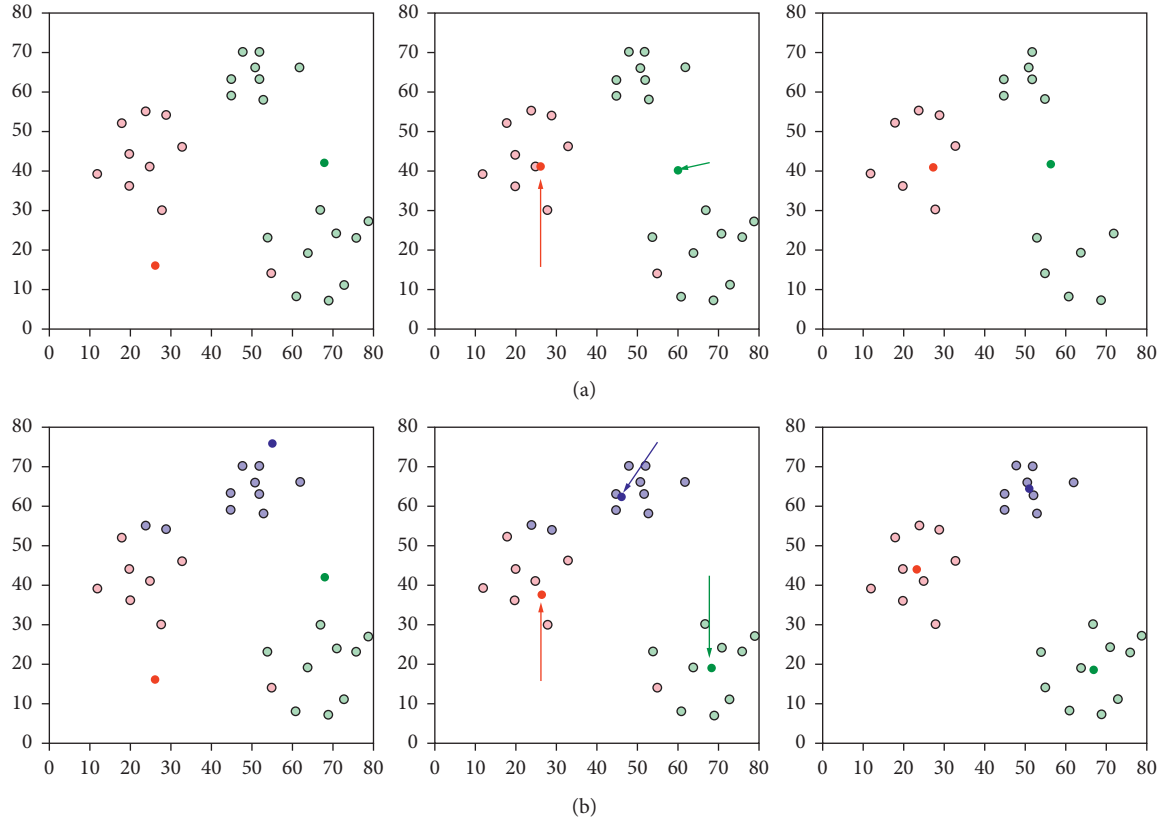


Figure 1: Clustering results of the artificial dataset D_1.



(a)

(b)

Figure 2: Continued.

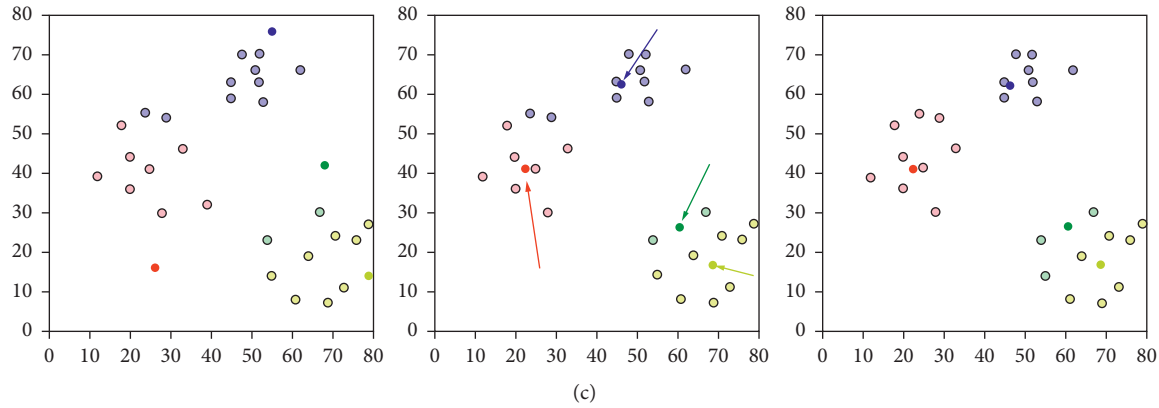


Figure 2: Schematic diagram of k-prototypes algorithm sensitive to initial Cluster Center selection. Clustering results of (a) $k = 2$, (b) $k = 3$, and (c) $k = 4$.

process, the importance of a Categorical Feature is inversely proportional to its dissimilarity [15]. To some extent, the Information Entropy of each Categorical Feature reflects the weight(w_s) of each Categorical Feature. Therefore, according to the uncertainty of the value of each Categorical Feature, this paper uses Information Entropy to calculate the importance of each Categorical Feature in the clustering process and assigns weight(w_s) to the Dissimilarity Coefficient.

Definition 1 (the intracluster relative frequency of eigenvalues). Suppose that most of the data objects in cluster C_l contain the same Categorical Eigenvalue $A_{s,t}^C$ which means that the eigenvalue $A_{s,t}^C$ appears frequently in cluster C_l, so the intracluster dissimilarity of the eigenvalue $A_{s,t}^C$ will be low. The intracluster relative frequency of the eigenvalues is defined as follows:

$$P\left(A_{s,t}^C\right) = \frac{\left|A_{s,t}^C\right|_{C_l}}{\left|C_l\right|}. \tag{4}$$

Definition 2 (the intercluster distribution frequency of eigenvalues). The intercluster distribution frequency of the eigenvalue refers to the occurrence frequency of the eigenvalue $A_{s,t}^C$ relative to the total frequency of the eigenvalue in all clusters. The intercluster distribution frequency of the eigenvalue is defined as follows:

$$PR\left(A_{s,t}^C\right) = \frac{\left|A_{s,t}^C\right|_{C_l}}{\sum_{i=1}^{k}\left|A_{s,t}^C\right|_{C_l}}. \tag{5}$$

Definition 3 (Dissimilarity Coefficient of categorical). Let $d^C(x_i, q_l)$ represent the dissimilarity of the Categorical Data portion of the mixed-type dataset, and the definition is shown in the following formula:

$$d^C(x_i, q_l) = \delta\left(x_{i,1}^C, q_{l,1}^C\right) + \delta\left(x_{i,2}^C, q_{l,2}^C\right) + \cdots + \delta\left(x_{i,p}^C, q_{l,p}^C\right) = \sum_{s=1}^{p}\delta\left(x_{i,s}^C, q_{l,s}^C\right),$$

$$\text{where} \quad \delta\left(x_{i,s}, q_{l,s}\right) = \begin{cases} 1 - \dfrac{\left|A_{s,t}^C\right|_{C_l}}{\left|C_l\right|}\dfrac{\left|A_{s,t}^C\right|_{C_l}}{\sum_{i=1}^{k}\left|A_{s,t}^C\right|_{C_l}}, & A_{i,s}^C = A_{ql,s}^C, \\ 1, & A_{i,s}^C \neq A_{ql,s}^C. \end{cases} \tag{6}$$

Definition 4 (the entropy of Categorical Feature). From the perspective of information theory, the importance of feature can be seen as the dissimilarity of the dataset relative to the feature. Basak [17] mentions that if the information content of a feature is high, the dissimilarity of the feature is also high. Let x_i be a discrete random variable belonging to the finite dataset D, and $P(x_i)$ is a probability function of the discrete random variable x_i. Because the eigenvalue domain of Categorical Data is certain, the eigenvalues in the eigenvalue domain can be regarded as discrete and

independent. Suppose there are n_s different eigenvalues of a certain Categorical Feature (A_s^C), and the probability of the occurrence of each eigenvalue is $p(A_{s,t}^C), 1 \le t \le n_s$, then the importance of Categorical Feature can be calculated by formula (7) [17]. $P(A_{s,t}^C)$ is the intracluster relative frequency of the eigenvalue $A_{s,t}^C$ mentioned in Definition 1:

$$EnA_s^C = -\sum_{A_{s,t}^C \in DOM(A_s^C)} P\left(A_{s,t}^C\right)\log\left(P\left(A_{s,t}^C\right)\right). \tag{7}$$

Theorem 1. *($0 \le P(A_{s,t}^C) \le 1, \sum_{t=1}^{n_s} P(A_{s,t}^C) = 1$). The larger the value of $P(A_{s,t}^C)$, the larger the proportion of the eigenvalues $A_{s,t}^C$ in the feature A_s^C. Then, the intracluster dissimilarity between data objects to be clustered with the eigenvalues $A_{s,t}^C$ and the cluster C_l is smaller.*

Observing formula (7), we can find that the more the possible values of eigenvalue $A_{s,t}^C$ are, the smaller the entropy of the Categorical Feature is. In practice, it is not the case that the larger the value domain of eigenvalues, the higher the importance. Considering the actual situation, the more different values a data object has on a feature, the less influence this feature has on clustering. In order to reduce the influence of Categorical Feature with too many different values on clustering [18], formula (7) is further modified as follows.

Definition 5 (quantified entropy). When defining the entropy of the feature A_s^C, we divide by the number (n_s) of possible values of feature A_s^C. The definition of entropy of Categorical Feature (A_s^C) after quantization is shown as follows:

$$E_n'A_s^C = -\frac{1}{n_s}\cdot\sum_{t=1}^{n_s} P\left(A_{s,t}^C\right)\log\left(P\left(A_{s,t}^C\right)\right). \tag{8}$$

Definition 6 (weight (w_s) on the s^{th}-dimensional feature of Categorical Feature). The eigenvalue distribution of each dimension feature is different, and the eigenvalue of different distribution will make the feature of different dimension Categorical Feature occupy different importance. In order to better discover all or part of the "prototypes" hidden in the dataset, the weight of each dimension should be taken into account when defining the Dissimilarity Coefficient. Let the weight of each dimension feature be $w_1, \ldots, w_s, \ldots, w_m$ and $w_s > 0$, $s = 1, 2, \ldots, m$, the weighted data object is $x_i^T = W \times x_i^T$, where $i = 1, 2, \ldots, n$. The weight W is defined as shown in the following formula:

$$W = \begin{bmatrix} w_1 & 0 & 0 & 0 & 0 \\ 0 & w_2 & 0 & 0 & 0 \\ 0 & 0 & w_s & 0 & 0 \\ 0 & 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & 0 & w_p \end{bmatrix}. \tag{9}$$

The weight of the Categorical Feature per dimension is defined as the ratio of the redundancy of the dimension feature to the sum of the overall redundancy. The calculation method of entropy redundancy is $1 - E_n'A_s^C$. All quantized entropy values in the dataset were normalized to obtain the entropy weight of each Categorical Feature. The definition of w_s is shown as follows:

$$w_s = \frac{1 - E_n'A_s^C}{\sum_{s=1}^{p}\left(1 - E_n'A_s^C\right)}. \tag{10}$$

Theorem 2 ($0 \le w_s \le 1, \sum_{s=1}^{p} w_s = 1$). $\sum_{s=1}^{p} E_n'A_s^C$ is the sum of the weights of all features. The larger the value w_s, the larger the weight of the feature A_s^C.

Definition 7 (Dissimilarity Coefficient of categorical based on entropy weight). For any $x_i, q_l \in D$, the Dissimilarity Coefficient of categorical based on entropy weight between x_i and q_l is defined as shown in the following formula:

$$d_w^C(x_i, q_l) = \sum_{s=1}^{p}\delta\left(w_s \cdot x_{i,s}, w_s \cdot q_{l,s}\right) = \sum_{s=1}^{p} w_s \cdot \delta\left(x_{i,s}, q_{l,s}\right), \tag{11}$$

The proposed Dissimilarity Coefficient is demonstrated by using the artificial dataset D_2 which is shown in Table 3. The dataset is described by three Categorical Features$A = \{A_1, A_2, A_3\}$, $\mathrm{DOM}(A_1) = \{\mathrm{A}, \mathrm{B}, \mathrm{C}\}$, $\mathrm{DOM}(A_2) = \{\mathrm{E}, \mathrm{F}\}$, and $\mathrm{DOM}(A_3) = \{\mathrm{H}, \mathrm{I}, \mathrm{J}\}$; there are three clusters C_1,C_2, and C_3 corresponding to the Cluster Centers $q_1(\mathrm{A}, \mathrm{E}, \mathrm{H})$, $q_2(\mathrm{A}, \mathrm{E}, \mathrm{H})$, and $q_3(\mathrm{B}, \mathrm{E}, \mathrm{J})$. Suppose clustering division is needed for the new data object $x_{16}(\mathrm{A}, \mathrm{E}, \mathrm{H})$.

The dissimilarity computing result of the k-prototypes algorithm is $d(x_{16}, q_1) = d(x_{16}, q_2) = 0 + 0 + 0 = 0$, and $d(x_{16}, q_3) = 1 + 0 + 1 = 2$. The dissimilarity computing result of the EKACMD algorithm is $d(x_{16}, q_1) = (1 - (3/5) \times (3/7)) + (1 - (4/5) \times (4/11)) + (1 - (5/5) \times (5/9)) = 1.896392$, $d(x_{16}, q_2) = 1.896392$, and $d(x_{16}, q_3) = 2.836363$. The dissimilarity computing result of the WKPCA algorithm is $w_{A_1,C_1} = 1 - E_n'A_{1,C_1}^C / [1 - E_n'A_{1,C_1}^C] + [1 - E_n'A_{2,C_1}^C] + [1 - E_n'A_{3,C_1}^C] = 0.311037$; $d(x_{16}, q_1) = (1 - (3/5) \times (3/7)) \times w_{A_1,C_1} + (1 - (4/5) \times (4/11)) \times w_{A_2,C_1} + (1 - (5/5) \times (5/9)) \times w_{A_3,C_1} = 0.589848$ $d(x_{16}, q_2) = 0.632130$ and $d(x_{16}, q_3) = 0.946240$. According to the above calculation, it can be seen that the correct clustering division of x_{16} can be carried out by using WKPCA algorithm.

4.2. Quantized Numerical Dissimilarity Coefficient

Definition 8 (quantitative numerical Dissimilarity Coefficient). The classical k-prototypes algorithm uses the Euclidean Distance to calculate the dissimilarity of the numerical part. Direct calculation of data of different orders of magnitude will not only increase the difficulty of

Table 3: Artificial dataset D_2.

Cluster	C_1			Cluster	C_2			Cluster	C_3		
Data objects/features	A_1	A_2	A_3	Data objects/features	A_1	A_2	A_3	Data objects/features	A_1	A_2	A_3
x_1	A	E	H	x_6	A	E	H	x_{11}	B	E	I
x_2	A	E	H	x_7	A	E	H	x_{12}	B	E	I
x_3	A	E	H	x_8	B	E	H	x_{13}	B	E	J
x_4	B	E	H	x_9	A	F	H	x_{14}	B	F	J
x_5	B	F	H	x_{10}	A	E	I	x_{15}	C	F	J
Cluster Centers	q_1 (A, E, H)			Cluster Centers	q_2 (A, E, H)			Cluster Centers	q_3 (B, E, J)		

calculation, but also cause a large error between the calculated results and the real situation. Therefore, Numerical Data should be dimensionless before calculation. The paper adopts Max-Min Standardization, and the processing method is shown in formula (12). The quantified numerical Dissimilarity Coefficient is defined as follows:

$$A_{i,s}^{N'} = \frac{A_{i,s}^{N} - \min\left(A_{i,s}^{N}\right)}{\max\left(A_{i,s}^{N}\right) - \min\left(A_{i,s}^{N}\right)}, \tag{12}$$

$$d^{N}(x_i, q_l) = \sum_{s=p+1}^{m} \sqrt{\left|A_{i,s}^{N'} - \text{means}_{l,s}^{N'}\right|^2}. \tag{13}$$

4.3. Weighted Hybrid Dissimilarity Coefficient

Definition 9 (weighted Hybrid Dissimilarity Coefficient). Suppose the mixed-type dataset D has m-dimension features (the front p-dimension features are Categorical Features, and the latter $m - p$-dimension features are Numerical Features). Numerical Features are treated as a whole (a vector), while the Categorical Features are treated as p-dimensional vectors. Take the example of a data object x_i which has a Categorical Feature of p-dimension and has a Numerical Feature of $m - p$-dimension. In the process of calculating dissimilarity, one numerical vector and p-dimensional categorical vectors are chosen for calculation. That is, there are $1 + p$-dimensional vectors involved in the calculation of Dissimilarity Coefficient. Therefore, for arbitrary data object in the dataset D, the Dissimilarity Coefficient between them is defined as follows:

$$d_w(x_i, q_l) = \frac{p}{1+p} \sum_{s=1}^{p} w_s \cdot \delta\left(x_{i,s}^{C}, q_{l,s}^{C}\right) + \frac{1}{1+p} d^{N}(x_i, q_l). \tag{14}$$

4.4. Determination of Initial Cluster Centers.

The classical k-prototypes algorithm is very sensitive to the selection of Cluster Centers. The appropriate initial Cluster Centers and cluster number k are particularly important for k-prototypes algorithm.

Definition 10 (the average distance). The average distance between two data objects x_i and x_j is defined as follows:

$$\overline{d} = \frac{2}{n(n-1)} \sum_{i=1}^{n} \sum_{j=i+1}^{n-1} d_{i,j}. \tag{15}$$

Definition 11 (local neighborhood density). The local neighborhood density is defined as shown in the following formula:

$$\begin{cases} \rho_i = \sum_{i \neq j} \chi\left(d_{i,j} - d_c\right), \\ \chi\left(d_{i,j} - d_c\right) = \begin{cases} 1, & d_{i,j} - d_c \leq 0, \\ 0, & d_{i,j} - d_c > 0, \end{cases} \end{cases} \tag{16}$$

where $\chi(x)$ is a piecewise function, when $d_{i,j} - d_c \leq 0$, $\chi(x) = 1$; otherwise, $\chi(x) = 0$ cutoff distance d_c is a critical value that limits the search scope.

Definition 12 (distance threshold). L is defined as the distance threshold between arbitrarily data objects x_i and x_j in dataset D, which is defined as follows:

$$L_i = \begin{cases} \min\limits_{x_i}\left(d_{i,j}\right), & \rho_i < \rho_j, \\ \max\limits_{x_i}\left(d_{i,j}\right), & \rho_i \geq \rho_j. \end{cases} \tag{17}$$

The Cluster Centers generally satisfies the following two assumptions. Firstly, the local neighborhood density of the central point of the cluster is higher than that of the surrounding noncentral point of the cluster. Second, the relative distance between the center points of each cluster is large. Based on the above assumptions, this section presents the specific process of self-determining the initial Cluster Centers:

Step 1: formula (2) is used to calculate the distance matrix of the data.

Step 2: formula (16) is used to calculate the local neighborhood density value (ρ_i).

Step 3: formula (17) is used to calculate the distance threshold (L_i).

Step 4: sort the data in the dataset in descending order according to the local neighborhood density and get the sort sequence $D' = \{x_1', x_2', x_3', \ldots x_n',\}$. x_1' is the initial Cluster Center q_1, and q_1 is stored into the Cluster Centers set Q.

Step 5: for $x_i' \in D'$, determine whether there is a x_1' that satisfies dist $(x_i', q_l) > L$. If it is satisfied, x_i' is taken as the

next Cluster Center and put into the Cluster Centers set Q. Otherwise, proceed to the next data object x'_{i+1}.

Step 6: determine whether all the data objects in D' have been accessed. If not, skip to Step 4 to continue execution. Otherwise, the elements in the collection of Q are the initial Cluster Centers, and $|Q|$ is clustering number k.

Step 7: end.

4.5. Cost Function considering Weights

Definition 13. (Cost Function considering weights). WKPCA algorithm is to find k subclusters where the following Cost Function as shown in formula (18) is minimized:

$$F_w(U,Q) = \sum_{i=1}^{n}\sum_{s=1}^{m}\sum_{l=1}^{k} u_{il} \cdot d_w(x_i, q_l), \tag{18}$$

where u_{il} represents the membership degree of the data object x_i to the cluster C_l. $U_{n\times k}$ represents the membership degree matrix. $u_{il} = 1$ indicates that the data object x_i belongs to the cluster C_l, and $u_{il} = 0$ indicates that the data object x_i does not belong to the cluster C_l. $F(U,Q)$ is the cost of dividing x_i, that is, the sum of the dissimilarity of all data objects in cluster C_l to the center of its cluster. When the value of the Cost Function reaches a minimum value when the constraint conditions: $u_{il} \in \{0,1\}, 1 \le i \le n, 1 \le l \le k$, $\sum_{l=1}^{k} u_{il} = 1, 1 \le i \le n$, $0 < \sum_{i=1}^{n} u_{il} < n, 1 \le l \le k$ are satisfied, the clustering process ends. The WKPCA algorithm steps are described as follows:

Input: dataset D containing n data objects

Output: k subclusters after clustering

Step 1: initialization procedure. Formula (2) is used to calculate the dissimilarity between the data objects.

Step 2: according to the automatic selection method of initial Cluster Centers in Section 4.4, k data objects are selected from dataset D as the initial Cluster Centers.

Step 3: iterative process. Formula (11) was used to calculate the dissimilarity between the data object and the Cluster Center, and x_i was assigned to the nearest cluster according to the calculation results.

Step 4: according to the current cluster center, the dissimilarity of the data object is recalculated. Update the Cluster Centers.

Step 5: repeat Steps 2 and 3 until the Cost Function is no longer changing. If the Cost Function is no longer changed, the algorithm ends. Otherwise, skip to Step 2 to continue.

The flowchart of WKPCA algorithm is shown in Figure 3.

The time complexity of the algorithm in this paper is higher than that of the classical k-prototypes algorithm, which is mainly consumed in the process of selecting the initial Cluster Centers. However, after the optimal initial Cluster Centers is determined, the number of iterations will be reduced and satisfactory clustering results will be obtained, so as to make up for the high time complexity to some extent.

5. Experimental Results and Analysis

5.1. Experimental Environment. Simulation experiments in this article are implemented in Python, and all experiments are run on the i7-8700K CPU@3.70GHz in the Intel(R) Core(TM), Windows 10 operating system. For experimental verification, five mixed-type datasets of Bank Marking (short name Bank), Zoo, Heart Disease (short name Heart), Lymphography (short name Lym), and Australian Credit Approval (short name ACA) were selected from UCI (UCI datasets: http://archive.ics.uci.edu/ml/datasets.html.2011) machine learning database. The details of the selected dataset are shown in Table 4.

The dataset used in this article has data missing phenomena, such as the ACA dataset. Deleting missing data directly from the dataset does not affect the clustering results. Therefore, before the experiment, all the data with missing values were deleted to ensure the integrity of the dataset and the accuracy of the clustering results. The complete version of the ACA dataset has 690 pieces of data, and the paper selects 623 pieces of data with complete eigenvalues to form a cleaned dataset. In addition, the numerical features are normalized by using the Maxi-Mini Normalization methods.

5.2. Performance Index. In order to evaluate the quality of clustering, the index AC (clustering accuracy) shown in formula (19) was used as the evaluation criterion. The indicator AC represents the ratio of the number of data objects correctly divided into the cluster C_l to the total number of data objects. The closer the clustering result is to the real clustering partitioning result of the dataset, the larger the index AC value is, the better the clustering result is, that is, the better the clustering algorithm is. NUM^+ indicates the number of data objects correctly assigned to C_l:

$$\mathrm{AC} = \frac{\sum_{l=1}^{k} NUM^+}{n}. \tag{19}$$

5.3. Analysis of Experimental Results. In order to verify the universality of WKPCA algorithm and eliminate the accidental results of a single experiment, we experimented with multiple real UCI datasets. The experiment compared the performance of WKPCA algorithm with k-prototypes algorithm proposed by Huang [6] and the EKACMD algorithm proposed by Sangam [13] on mixed-type datasets. Each algorithm was executed 30 times on each dataset to take the average value, and the statistics of clustering results are summarized in Tables 5–9. For k-prototypes algorithm and EKACMD algorithm, five different clustering parameters $k = 2$, $k = 4$, $k = 6$, $k = 8$, and $k = 10$ were set in the paper for

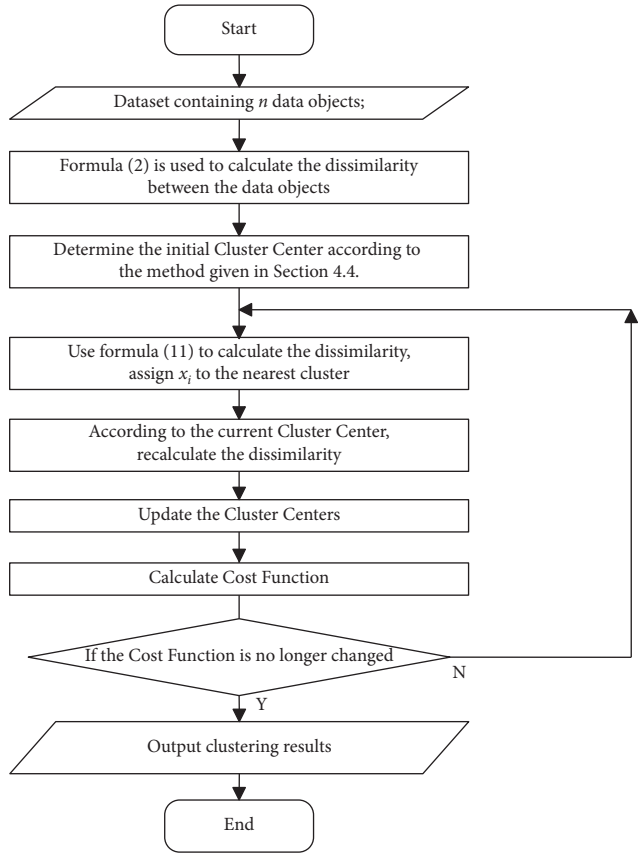


Figure 3: Flowchart of WKPCA algorithm.

Table 4: Description of mixed-type dataset.

Dataset	# of features		# of data	# of missing data	# of clusters
	Categorical (p)	Numerical ($m-p$)			
Bank	10	10	41188	0	2
Zoo	15	1	101	0	7
Heart	7	6	303	0	2
Lym	16	2	148	0	4
ACA	9	6	690	67	2

experiments. Since neither EKACMD nor WKPCA needed to set clustering parameter γ, the paper only sets cluster parameters of k-prototypes separately.

The Bank dataset has 41,188 data objects, 10 Categorical Features, 10 Numerical Features, and 2 clusters. The Bank dataset was sampled at a sampling rate of 4.8%. Table 5 shows that when $k = 2$, the AC values of WKPCA are 9.43% and 5% higher than those of k-prototypes and EKACMD, respectively.

The Zoo dataset has 101 data objects, 15 Categorical Features, 1 Numerical Feature, and 7 clusters. Table 6 shows that when $k = 7$, the AC values of WKPCA were 11.88% and 4.95% higher than those of k-prototypes and EKACMD, respectively.

The Heart Disease dataset has 303 data objects, 7 Categorical Features, 6 Numerical Features, and 2 clusters. Table 7 shows that when $k = 2$, the AC values of WKPCA were 10.35% and 2.7% higher than those of k-prototypes and EKACMD, respectively.

The Lym dataset has 148 data objects, 7 Categorical Features, 6 Numerical Features, and 4 clusters. Table 8 shows that when $k = 4$, the AC values of WKPCA are 9.62% and 6.45% higher than those of k-prototypes and EKACMD, respectively.

The ACA dataset has 690 data objects, 9 Categorical Features, 5 Numerical Features, and 2 clusters. Table 9 shows that when $k = 2$, the AC values of WKPCA are 5.02% and2.99% higher than those of k-prototypes and EKACMD, respectively.

The results in Tables 5–9 show that, in terms of clustering accuracy, the proposed algorithm achieves better clustering results than other algorithms. The Hybrid Dissimilarity

Table 5: Comparison of cluster accuracy on the Bank dataset.

Parameter settings and sampling rate = 4.8%	k-prototypes	EKACMD	WKPCA
$k = 2$	0.783195	0.8275	0.88535
$k = 4$	0.8272	0.838	—
$k = 6$	0.8205	0.8885	—
$k = 8$	0.891292	0.904	—
$k = 10$	0.900089	0.912	—

Table 6: Comparison of cluster accuracy on the Zoo dataset.

Parameter settings	k-prototypes	EKACMD	WKPCA
$k = 2$	0.514851	0.574257	—
$k = 4$	0.693069	0.70297	—
$k = 6$	0.70297	0.742574	—
$k = 7$	0.742574	0.811881	0.86732
$k = 8$	0.60396	0.70297	—
$k = 10$	0.712871	0.722772	—

Table 7: Comparison of cluster accuracy on the Heart dataset.

Parameter settings	k-prototypes	EKACMD	WKPCA
$k = 2$	0.426558	0.557158	0.584158
$k = 4$	0.520010	0.630300	—
$k = 6$	0.576567	0.660066	—
$k = 8$	0.569966	0.676468	—
$k = 10$	0.589570	0.692970	—

Table 8: Comparison of cluster accuracy on the Lym dataset.

Parameter settings	k-prototypes	EKACMD	WKPCA
$k = 2$	0.545247	0.542292	—
$k = 4$	0.572702	0.604324	0.668918
$k = 6$	0.529730	0.574324	—
$k = 8$	0.625945	0.661621	—
$k = 10$	0.600000	0.651351	—

Table 9: Comparison of cluster accuracy on the ACA dataset.

Parameter settings	k-prototypes	EKACMD	WKPCA
$k = 2$	0.532330	0.552706	0.582608
$k = 4$	0.682198	0.692898	—
$k = 6$	0.669710	0.678260	—
$k = 8$	0.700000	0.708695	—
$k = 10$	0.697101	0.700000	—

Coefficient considers the importance of each feature in the clustering process and can automatically calculate the weights of different features. The above reasons enable the algorithm in this paper can obtain better clustering results.

Figure 4 shows the clustering accuracy of WKPCA, k-prototypes, and EKACMD on the Bank dataset with different parameters k. It can be seen from Figure 4 that WKPCA has a good clustering result.

Figure 5 shows the clustering accuracy of WKPCA, k-prototypes, and EKACMD on the Zoo dataset with different parameters k. It can be seen that the curve of WKPCA is higher than that of k-prototypes and EKACMD.

Figure 6 shows the clustering accuracy of WKPCA, k-prototypes, and EKACMD on the Heart dataset with different parameters k. It can be seen that when $k = 4$ and $k = 6$, the clustering precision of k-prototypes and EKACMD is relatively close. The true cluster number of the Heart dataset was $k = 2$, and the clustering accuracy of WKPCA at $k = 2$ is much better than that of EKACMD.

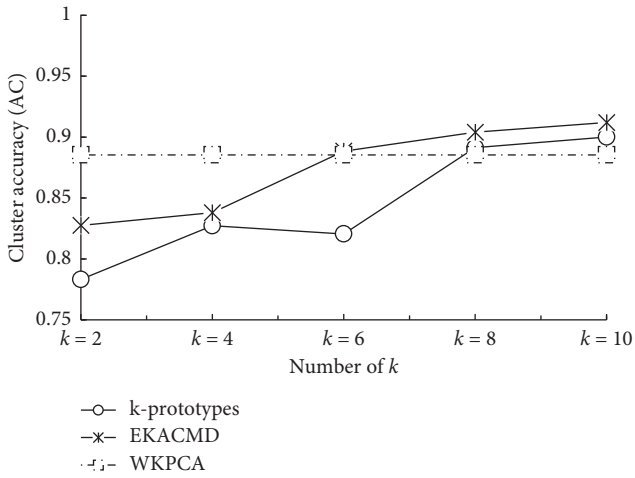


Figure 4: Comparison of cluster accuracy on the Bank dataset.

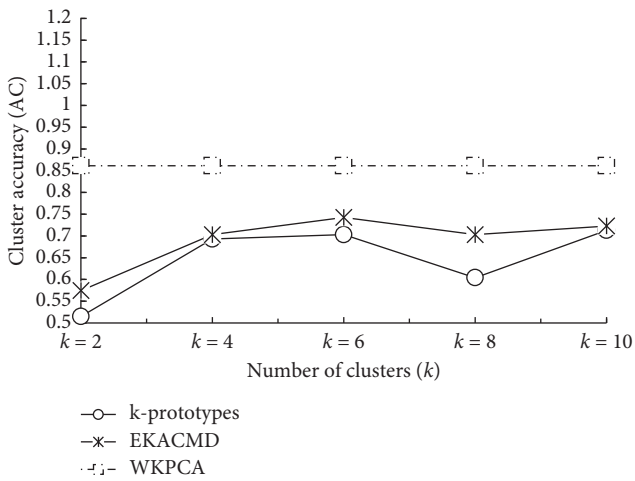


Figure 5: Comparison of cluster accuracy on the Zoo dataset.

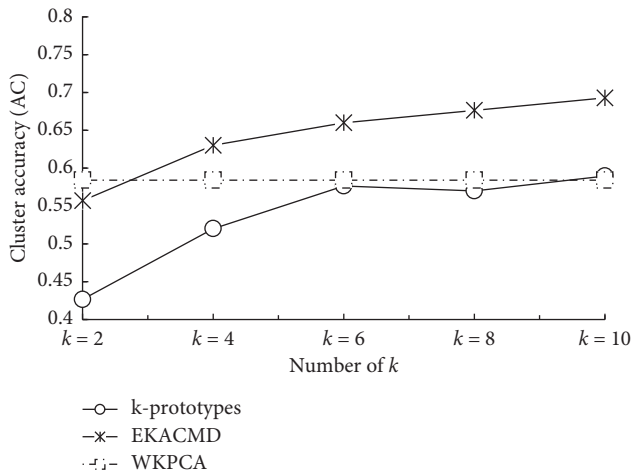


Figure 6: Comparison of cluster accuracy on the Heart dataset.

Figure 7 shows the clustering accuracy of WKPCA, k-prototypes, and EKACMD on the Lym dataset with different parameters k. It can be seen that the effect of WKPCA

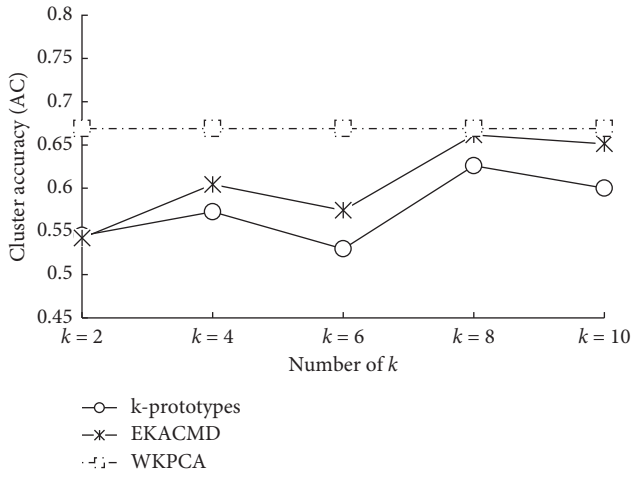


Figure 7: Comparison of cluster accuracy on the Lym dataset.

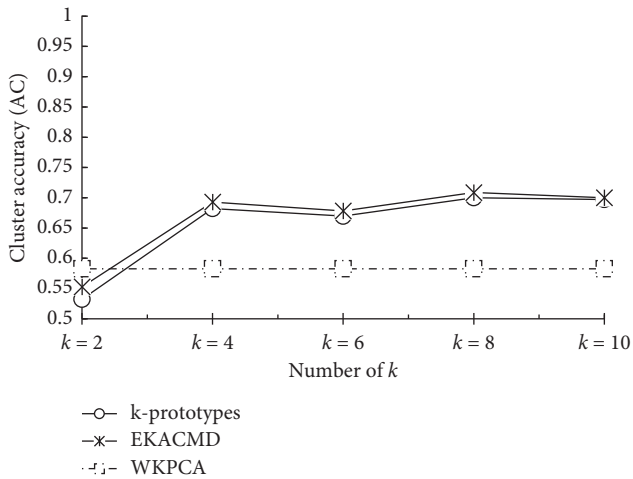


Figure 8: Comparison of cluster accuracy on the ACA dataset.

algorithm is better than that of k-prototypes and EKACMD overall.

Figure 8 shows the clustering accuracy of WKPCA, k-prototypes, and EKACMD on the ACA dataset with different parameters k. When $k = 2$, the clustering accuracy is WKPCA > EKACMD > k-prototypes.

From Figures 4–8, it can be seen that, in the case of random initialization, the proposed parameter-free WKPCA algorithm is superior to the k-prototypes algorithm and the EKACMD algorithm in clustering accuracy. As can be seen from the detailed information of the dataset shown in Table 4, the ratio of the Categorical Features and the Numerical Features in the selected datasets is mostly different. For example, the Zoo dataset has 1 Numerical Feature and 15 Categorical Features; the Lym dataset has 2 Numerical Features and 16 Categorical Features. Although there is a large gap between the two types of feature distribution of these datasets, the WKPCA algorithm still achieves satisfactory clustering results. This indicates that the proposed Hybrid Dissimilarity Coefficient is applicable for various

complex datasets, and it is not necessary to set any parameters manually to adjust the weights of the Categorical Features and the Numerical Features.

6. Conclusions

The weighted k-prototypes clustering algorithm based on the Hybrid Dissimilarity Coefficient is an extension of the classical k-prototypes clustering algorithm. The method of automatic selection of initial Cluster Centers is improved by means of average distance, local neighborhood density, and relative distance. Considering the spatial distribution information of the data, the Cluster Center is more in line with the actual situation. The uncertainty of clustering caused by different initial Cluster Center selection is avoided. For Categorical Data, the coefficient of type dissimilarity based on entropy weight is used. For Numerical Data, different numerical values are standardized by using quantized numerical Dissimilarity Coefficient. For mixed-type data, the paper used a weighted Hybrid Dissimilarity Coefficient. The proposed Hybrid Dissimilarity Coefficient not only retained the characteristics of different types of data, but also effectively improved the clustering accuracy and clustering effectiveness, and its robustness was better than other k-prototypes clustering algorithms. Finally, WKPCA algorithm is proposed to realize mixed-type data clustering. In Step 1, the WKPCA algorithm automatically determines the initial Cluster Centers by calculating the average distance and local neighborhood density. Compared with other k-prototypes algorithms, it takes more time, but a more accurate Cluster Centers can be selected in the initial stage of clustering. Make sure the Cluster Centers is located in the region with the highest sample density, and the distance between them is the longest, which reduces the number of algorithm iterations. The paper algorithm improves the clustering accuracy, but sacrifices the time performance. Therefore, the next step will focus on improving time complexity. To sum up, although the time performance of the proposed algorithm needs to be improved, its clustering accuracy and clustering effectiveness have been significantly improved.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (grant no. 61762030), the Innovation-Driven Development Special Fund of Guangxi (grant no. AA17204017), and the Scientific Research and Technology Development Program of Guangxi (grant nos. AB19110050 and AB18126094).

References

[1] A. L. Simone, "MapReduce-based fuzzy c-means clustering algorithm: implementation and scalability," *International Journal of Machine Learning and Cybernetics*, vol. 6, no. 6, pp. 923–934, 2015.

[2] N. I. On, T. Boongoen, and N. Kongkotchawan, "A new link-based method to ensemble clustering and cancer microarray data analysis," *International Journal of Collaborative Intelligence*, vol. 1, no. 1, pp. 45–67, 2014.

[3] J. Macqueen, "Some methods for classification and analysis of multivariate observation," in *Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability*, California Press, Berkeley, CA, USA, 1967.

[4] Z. Huang, "Extensions to the k-means algorithm for clustering large data sets with categorical values," *Data Mining and Knowledge Discovery*, vol. 2, no. 3, pp. 283–304, 1998.

[5] C.-C. Hsu, "Generalizing self-organizing map for categorical data," *IEEE Transactions on Neural Networks*, vol. 17, no. 2, pp. 294–304, 2006.

[6] Z. Huang, "Clustering large data sets with mixed numeric and categorical values," in *Proceedings of the 1st Pacific-Asia Conference on Knowledge Discovery and Data Mining*, Singapore, 1997.

[7] Y. M. Cheung and H. Jia, "A unified metric for categorical and numerical features in data clustering," in *Proceedings of the 17th Pacific-Asia Conference on Knowledge Discovery and Data Mining*, Springer-Verlag Press, Berlin, Germany, 2013.

[8] R. C. De Amorim and V. Makarenkov, "Applying subclustering and Lp distance in Weighted K-Means with distributed centroids," *Neurocomputing*, vol. 173, pp. 700–707, 2016.

[9] J. Ji, T. Bai, C. Zhou, C. Ma, and Z. Wang, "An improved k-prototypes clustering algorithm for mixed numeric and categorical data," *Neurocomputing*, vol. 120, pp. 590–596, 2013.

[10] Y. Yao and Y. Sun, "An anonymous algorithm for hierarchical clustering based on K-prototypes," in *Proceedings of the 2016 4th International Conference on Machinery, Materials and Information Technology Applications*, Berlin, Germany, 2016.

[11] M. Du, S. Ding, and H. Jia, "Study on density peaks clustering based on k-nearest neighbors and principal component analysis," *Knowledge-Based Systems*, vol. 99, no. 99, pp. 135–145, 2016.

[12] D. Guo, Y. Chen, and J. Chen, "A k-prototypes algorithm based on adaptive determination of the initial centroids," in *Proceedings of the 10th International Conference on Machine Learning and Computing*, ACM Press, New York, NY, USA, 2018.

[13] R. S. Sangam and O. Hari, "An equi-biased k-prototypes algorithm for clustering mixed-type data," *Speinger,Indian Academy of Sciences*, vol. 43, no. 3, p. 37, 2018.

[14] G. Cui and C. GaoHongwei, "Rough set processing out-li-ers in cluster analysis," in *Proceedings of the 2019 IEEE 4th International Conference on Cloud Computing and Big Data Analysis (ICCCBDA)*, New York, NY, USA, 2019.

[15] L. Kaufman and P. J. Rousseeuw, *Finding Groups in Data: An Introduction to Cluster Analysis*, 2009.

[16] C. E. Shannon, "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, no. 3, pp. 379–423, 1948.

[17] J. Basak and R. Krishnapuram, "Interpretable hierarchical clustering by constructing an unsupervised decision tree," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 1, pp. 121–132, 2005.

[18] M. Du, S. Ding, and Y. Xue, "A novel density peaks clustering algorithm for mixed data," *Pattern Recognition Letters*, vol. 97, pp. 46–53, 2017.