Hindawi
Mathematical Problems in Engineering
Volume 2023, Article ID 9762363, 1 page
https://doi.org/10.1155/2023/9762363

Retraction

Retracted: An Accurate Method of Determining Attribute Weights in Distance-Based Classification Algorithms

Mathematical Problems in Engineering

Received 13 September 2023; Accepted 13 September 2023; Published 14 September 2023

This article has been retracted by Hindawi following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of one or more of the following indicators of systematic manipulation of the publication process:

(1) Discrepancies in scope
(2) Discrepancies in the description of the research reported
(3) Discrepancies between the availability of data and the research described
(4) Inappropriate citations
(5) Incoherent, meaningless and/or irrelevant content included in the article
(6) Peer-review manipulation

The presence of these indicators undermines our confidence in the integrity of the article's content and we cannot, therefore, vouch for its reliability. Please note that this notice is intended solely to alert readers that the content of this article is unreliable. We have not investigated whether authors were aware of or involved in the systematic manipulation of the publication process.

Wiley and Hindawi regrets that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our own Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their agreement or disagreement to this retraction. We have kept a record of any response received.

References

[1] F. Liu and J. Wang, "An Accurate Method of Determining Attribute Weights in Distance-Based Classification Algorithms," *Mathematical Problems in Engineering*, vol. 2022, Article ID 6936335, 15 pages, 2022.

Hindawi
Mathematical Problems in Engineering
Volume 2022, Article ID 6936335, 15 pages
https://doi.org/10.1155/2022/6936335

Research Article

An Accurate Method of Determining Attribute Weights in Distance-Based Classification Algorithms

Fengtao Liu and Jialei Wang

Glorious Sun School of Business & Management, Donghua University, Shanghai 200051, China

Correspondence should be addressed to Fengtao Liu; lft@dhu.edu.cn

Received 6 April 2022; Revised 8 May 2022; Accepted 11 May 2022; Published 27 May 2022

Academic Editor: Xuefeng Shao

Weight determination aims to determine the importance of different attributes; determining accurate weights can significantly improve the accuracy of classification and clustering. This paper proposes an accurate method for attribute weight determination. The method uses the distance from the sample point of each class to the class center point. It can minimize the weights and determines the attribute weights of the constraints through the objective function. In this paper, the attribute weights obtained by the exact solution are applied to the K-means clustering algorithm; three classic machine learning data sets, the iris data set, the wine data set, and the wheat seed data set, are clustered. Using the normalized mutual information as the evaluation index, a confusion matrix was established. Finally, the clustering results are visualized and compared with other methods to verify the effectiveness of the proposed method. The results show that this method improves the normalized mutual information by 0.11 and 0.08, respectively, compared with the unweighted and entropy weighted methods for iris clustering results. Furthermore, the performance on the wine data set is improved by 0.1, and the performance on the wheat seed data set is improved by 0.15 and 0.05.

1. Introduction

Weights reflect the importance of different attributes, and the influence of different attribute weights on algorithm results is sometimes very different. It is necessary to determine accurate attribute weights. Let us take K-means as an example. K-means clustering is a typical distance-based clustering algorithm. K-means is widely used due to its fast-running speed, simplicity, and ease of understanding. However, traditional K-means does not consider the importance of features, resulting in poor clustering effects with traditional K-means in some problems. The distance class algorithm uses the distance between sample attributes to classify and cluster [1, 2]. Generally, the sample cluster is divided by clustering birds of a feather [3, 4] to achieve the effect of high similarity within the cluster and low similarity outside the cluster [5]. The distance between sample attributes is a "distance measure" [6, 7]. The similarity measure defined by us means that the larger the distance, the smaller the similarity [8, 9]. Differences between different attributes may not be obvious or even wrong in some distance performance, which can be achieved through "distance metric learning." In other words, assigning different weights to sample attributes improves learning effects [10].

At present, the problem of weight determination can be divided into two methods: subjective weight determination and objective weight determination. Domain experts compare the importance degree of each attribute with fuzzy language to determine the weight. The methods of subjective weight determination by experts include the analytic hierarchy process (AHP), sequence diagram method, simple weighting, etc. The analytic hierarchy process is a widely used method at present. Pourghasemi et al. used fuzzy logic and an analytic hierarchy process (AHP) model to make a landslide sensitivity map of Iran's landslide-prone area (Haraz) for land planning and disaster reduction [11]. Lin and Kou [12], based on the multiplication AHP model, proposed a heuristic method, and priority vectors were derived from the PCM in the whole hierarchy.

Although the subjective weight determination method has achieved good results in some conditions, it is limited by the shortcomings of artificial judgment, inability to find experts, and so on. Therefore, the objective weight determination method is used in many cases. The methods of objective weight mainly include the entropy weight method, principal component analysis method, and factor analysis. Meimei et al. proposed two methods to determine the optimal weight of attributes based on entropy and measure [13]. Chen combined the entropy weight method with Topsis to determine the weight of Topsis attributes and analyzed the influence of electronic warfare on Topsis [14]. Amaya et al. proposed a proposal on collaborative cross entropy to solve combinatorial optimization problems [15]. In addition to the above method, Lu et al. used a KNN combination of distance thresholds to determine the weight [16]. And other scholars used algorithm combinations to determine the weight [17–20]. In recent years, ensemble learning has become a research hotspot, and some scholars have determined the contribution degree of attributes to classification results through ensemble learning algorithms, for example, random forest [21], XGBoost, etc. Random forest determines the weight by calculating the attribute contribution, which is a way of calculating the weight value developed with the development of ensemble learning [22]. And Liu et al. constructed multiple mixed 0–1 linear programming models (MLPMs) to obtain the classification range of alternatives and weights of policy attributes applied in maldistributed decision-making problems [23].

In this paper, a distance-based classification algorithm is proposed to find the minimum distance between the midpoint of the category to which the data belong and the attribute vector. The distance between data points in the same category is closer and the distance between data points in different categories is farther to achieve the effect of improving the classification. In this paper, Lingo is used to solve the weights, and the solved weights are applied to the K-means clustering iris data set, wine data set, and wheat seed data set. Compared with the weights determined by the class and entropy weight method, the method proposed in this paper has different degrees of improvement in the clustering effect.

The key contributions of this work are as follows: (1) The algorithm accurately determines the attribute weights and identifies the solution from the data set itself. (2) This method overcomes the shortcomings of AHP and other methods. (3) It is less subjective and does not need to calculate entropy [24, 25]. (4) There is no need to use formulas such as variance to obtain attribute weights. There is no need for many trial and error steps, and there is no need for integrated learning to build models.

The rest of this paper is organized as follows: Section 2 explains the idea of solving the weights in this paper. Section 3 describes the K-means clustering process and evaluation indicators. Section 4 describes the experimental procedure. Section 5 is a summary of the full text.

2. Determining Weights

2.1. The Solution Idea. The purpose of clustering and classification is to obtain groups such that objects within a group are more similar than objects in different groups [26]. The weights are determined by minimizing the distances between attribute vectors within the same group and the center vector to maximize the distance between the different groups, thus effectively separating the different clusters. When the distance between the attribute vectors of each group and the center of the group reaches the minimum value, the distance between the different groups is maximized. The weight determined is the optimal attribute weight. The weight of the solution is applied to a known or unknown data set to improve the learning effect. The solution idea comes from the KNN algorithm [27].

2.1.1. KNN Algorithm. The KNN algorithm is a relatively mature and simple machine learning algorithm in theory. The idea of KNN is that if a sample has a high probability of belonging to a certain category among the k nearest samples in the feature space, and most of them belong to a certain category, then the sample is also classified in this category. KNN is classified by measuring the distance between different characteristic values, generally using the Euclidean distance. In classification decisions, this method only determines the category of the samples to be classified according to the category of the nearest sample or several samples. The KNN solution process is as follows:

Step 1: Calculate distances. The distance between characteristic values is calculated, the distance between the test data and each training data value. Generally, the Euclidean distance is used for calculation, and the Manhattan distance and Mahalanobis distance can also be used. Table 1 shows some distance formulas.

Step 2: Sort by increasing distance.

Step 3: Classify samples according to distance. Select k data points with the smallest distance from the sample point to determine the type of data with the highest frequency among the K sample points.

Step 4: Identify categories. The category with the highest frequency in the first K points is used as the predictive classification of the test data. Classification methods are divided into simple and weighted voting methods.

2.1.2. Weight Solution Idea. The idea of solving weights comes from the reverse solution method of KNN. KNN makes classification judgments according to the occurrence frequency of categories, and the purpose of determining the weight is to improve the learning effect. In the KNN algorithm, we aim to make all k surrounding sample points belong to a certain category. The distance between samples of the same category should be small, and the distance between samples of different categories should be large. The minimum distance between the sample vector of a category and the center point is reflected in the sample vector of the category. The steps of determining the weights are as follows:

Step 1: Identify categories. Classify sample data of different categories according to the known data.

Table 1: Several commonly used distance formulas.

Distance name	Brief explanation	Distance formula
Euclidean distance	The straight-line distance between two points	$d = \sqrt{\sum_{k=1}^{n} (x_{1k} - x_{2k})^2}$
Manhattan distance	The sum of the absolute wheelbases of two points in standard coordinates	$d = \sum_{k=1}^{n} \lvert x_{1k} - x_{2k} \rvert$
Chebyshev distance	The maximum value of the difference between coordinates	$d = \max_i (\lvert x_{1i} - x_{2i} \rvert)$
Markov distance	The covariance distance of data	$d = \sqrt{(X-\mu)^T S^{-1} (X-\mu)}$. The covariance matrix is denoted as S, and the mean is denoted as μ

Step 2: Choose K. The sample number of each category is calculated after classification, and the value K is the sample number of the category.

Step 3: Calculate the distance. Calculate the distance between the sample of the category and the center point vector, carry out the weighting calculation, and obtain the weight when the distance is the smallest.

2.2. Solution Process. The goal of this method is to minimize the distance between a classification sample of the data set and the center point of the category to which it belongs. In this experiment, the Euclidean distance is adopted. In addition to the Euclidean distance, other distance functions, such as the Mahalanobis distance, Manhattan distance, and Chebyshev distance, can be adopted. This paper presents an accurate analytical method for weighted attribute distance functions.

Sample classification $C = \begin{pmatrix} c_1 \\ \vdots \\ c_i \\ \vdots \\ c_n \end{pmatrix}$. The attribute vector of each sample is $x_i = \begin{pmatrix} r_1^i \\ \vdots \\ r_p^i \\ \vdots \\ r_k^i \end{pmatrix}$. The attribute vector values of the center point under the label are $s^i = \begin{pmatrix} s_1^i \\ \vdots \\ s_p^i \\ \vdots \\ s_k^i \end{pmatrix} (i = 1, 2, 3, \ldots, n)$, where $\lambda_p (p = 1, 2, \ldots, k)$ is the weight of each attribute. The constraint conditions are $0 \le \lambda_p \le 1$, $\sum_{p=1}^{k} \lambda_p = 1$, and $n = n_1 + n_2 + \cdots + n_i + \cdots + n_n$.

Let us define the objective function as

$$sd = \arg\min \sum_{m=1}^{n} \left[\sum_{i=1}^{n_i} \sqrt{\sum_{p=1}^{k} \lambda_p^2 \left(r_p^i - s_p^i \right)^2}_{\,m} \right], \tag{1}$$

where n_i is the number of samples under each category and n is the total number of samples. By solving the attribute vector of the center point of each label, the minimum value λ_p of the objective function is obtained by taking the partial derivative or using the gradient descent method. When sd is the minimum value, the weight λ_p of each attribute is obtained. Namely, the sum of the distance between the sample point of each category and the center point of each category is the smallest. Table 2 shows the meanings of the other parameters. In this experiment, the Euclidean distance is used to determine the weight; other distances can also be used for the calculation.

3. K-Means Algorithm

3.1. K-Means Algorithm Process. The K-means algorithm is an unsupervised learning algorithm that has become one of the most widely used clustering algorithms [28, 29]. It is a distance-based clustering algorithm that uses the distance between objects as an evaluation index of similarity.

The traditional K-means Algorithm 1 process is as follows:

3.2. Evaluation Indicators. In this experiment, the normalized mutual information [30, 31] (NMI) is used as the evaluation index of clustering quality. NMI is commonly used in clustering to measure the similarity of two clustering results. It can objectively evaluate the accuracy of an algorithm partition compared with the standard partition. The range of NMI is 0 to 1, and the higher it is, the greater the accuracy is. The concept of NMI comes from relative entropy, namely, *KL* divergence and mutual information.

Relative entropy is an asymmetrical measure of the difference between two probability distributions, and in the discrete case, it is defined as

$$KL(p||q) = \sum p(x) \log \frac{p(x)}{q(x)}, \tag{2}$$

where $p(x)$ and $q(x)$ are the two probability distributions of the random variable x.

Mutual information [32] is a useful information measure in information theory. It can be regarded as the amount of information contained in a random variable about another random variable. Mutual information is the relative entropy of the joint probability distribution and edge probability product distribution of two random variables X and Y, which is defined as

$$I(X;Y) = \sum_x \sum_y p(x,y) \log \frac{p(x,y)}{p(x)p(y)}. \tag{3}$$

Normalized mutual information is the result of the normalization of mutual information and is defined as

Table 2: Brief explanations of various parameters.

Parameter name	Parameter meaning
C	Sample classification
x_i	Sample attribute vector
s^j	Vector of the center point under each class
λ_p	Weight value
n_i	Number of samples under the category
n	Total number of samples
k	Number of attributes
sd	Objective function value

Input: number of clusters K, data set D
Output: K clusters.
Algorithm steps:
Step 1: Take K, which means we will divide the data set into K groups.
Step 2: Randomly select K points from the data set as the initial clustering centers.
Step 3: Calculate the distances between all points and the K cluster centers and put the samples into the class with the center with the shortest distance.
Step 4: Calculate the average coordinates of the data points in each class cluster to update the center of the cluster.
Step 5: Repeat steps (3) and (4) until the cluster center remains unchanged.

Algorithm 1: K-means clustering process.

$$NMI(X;Y) = 2\frac{I(X;Y)}{H(X) + H(Y)}, \tag{4}$$

where $H(X)$ and $H(Y)$ are the information entropy of the random variables X and Y and $I(X;Y)$ is the mutual information of X and Y.

3.3. K-Means with the Accurate Weight Determination Method. The traditional K-means algorithm does not consider the importance degree of attributes, so the distance weights from each attribute to the center point of the cluster are equal. However, in many cases, the importance of different attributes may not be equal. Application of traditional K-means to these scenarios will inevitably lead to inaccurate clustering results. In this paper, the exact solution process of feature weights is carried out before the K-means algorithm is applied. The obtained weights are weighted by the distance between each attribute and the center point to obtain the final distance between the sample point and the center of the cluster. Figure 1 shows the flowchart of the k-means algorithm using the exact weight solution method.

4. Experimental Process

4.1. Introduction to the Data Sets

4.1.1. Iris Data Set. The iris data set is a commonly used machine learning data set [33]. It includes four attributes, the length of the calyx (Speal Length), the width of the calyx (Speal Width), the length of the petal (Petal Length), and the width of the petal (Petal Width). The unit of the four attributes is CM, which is a numerical variable, and there are no missing values. Figure 2 shows a scatter plot of iris data attributes. Figure 3 shows the histogram of iris data attributes. The mountain iris, chameleon iris, and Virginia iris are the three categories. Each category collects 50 sample records, for a total of 150 irises.

4.1.2. Wine Data Set. The wine data set is a publicly available data set from the University of California Irvine (UCI). It is the result of a chemical analysis of wines grown in the same region of Italy from three different varieties. The analysis determined the values of 13 attributes of each of the three wines. The attributes are class identifiers, represented by categories 1, 2, and 3. Figure 4 shows the distribution of wine attributes. There are 59 samples in category 1, 71 samples in category 2, and 48 samples in category 3. There are no missing values in this data set.

4.1.3. Wheat Seed Data Set. The wheat seed data set is commonly used in classification and clustering tasks. There are 210 records, 7 features, and 1 label in the data set. Figure 5 shows the distribution of wheat seed attributes. The labels are divided into 3 categories with 70 samples in each category, and there are no missing values.

4.2. Determining Attribute Weights

4.2.1. Determining the Attribute Weights of the Iris Data Set. The category number of the iris data set is 3, so the objective function used to determine the weights of the four attributes according to formula (1) is

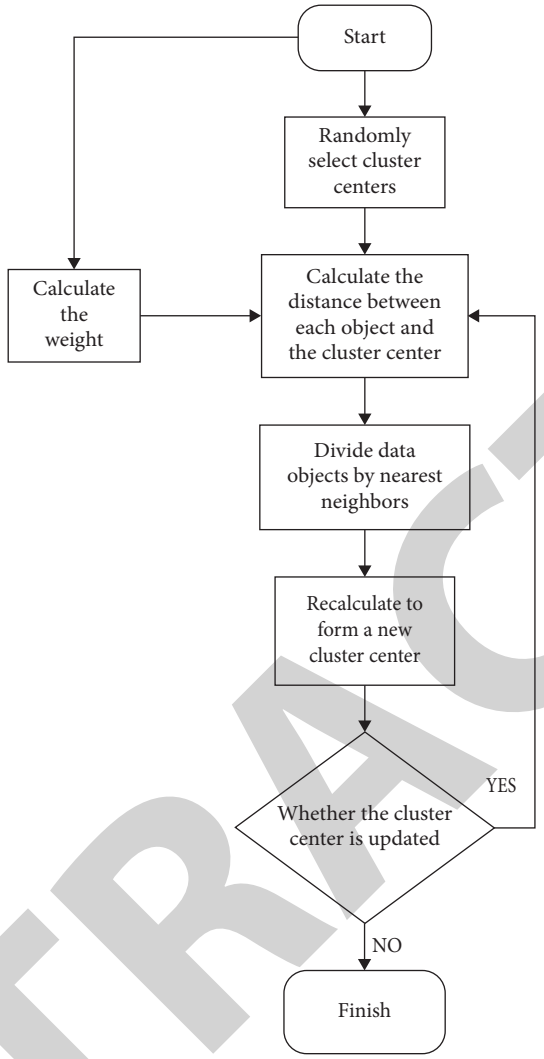

Figure 1: Flowchart of applying the exact weight determination method to the K-means algorithm.

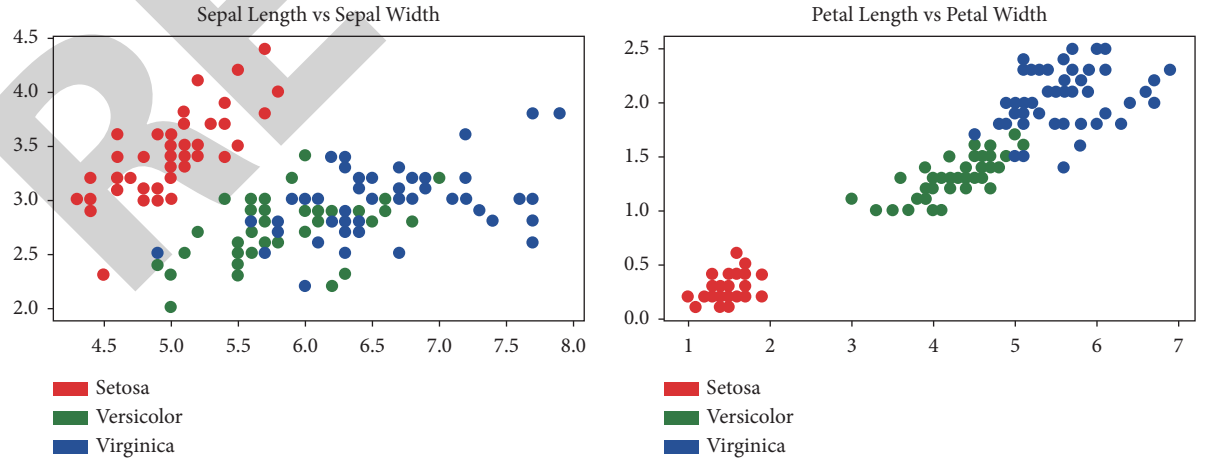

Figure 2: Scatter plot of calyx length and width and petal length and width in the iris data set.

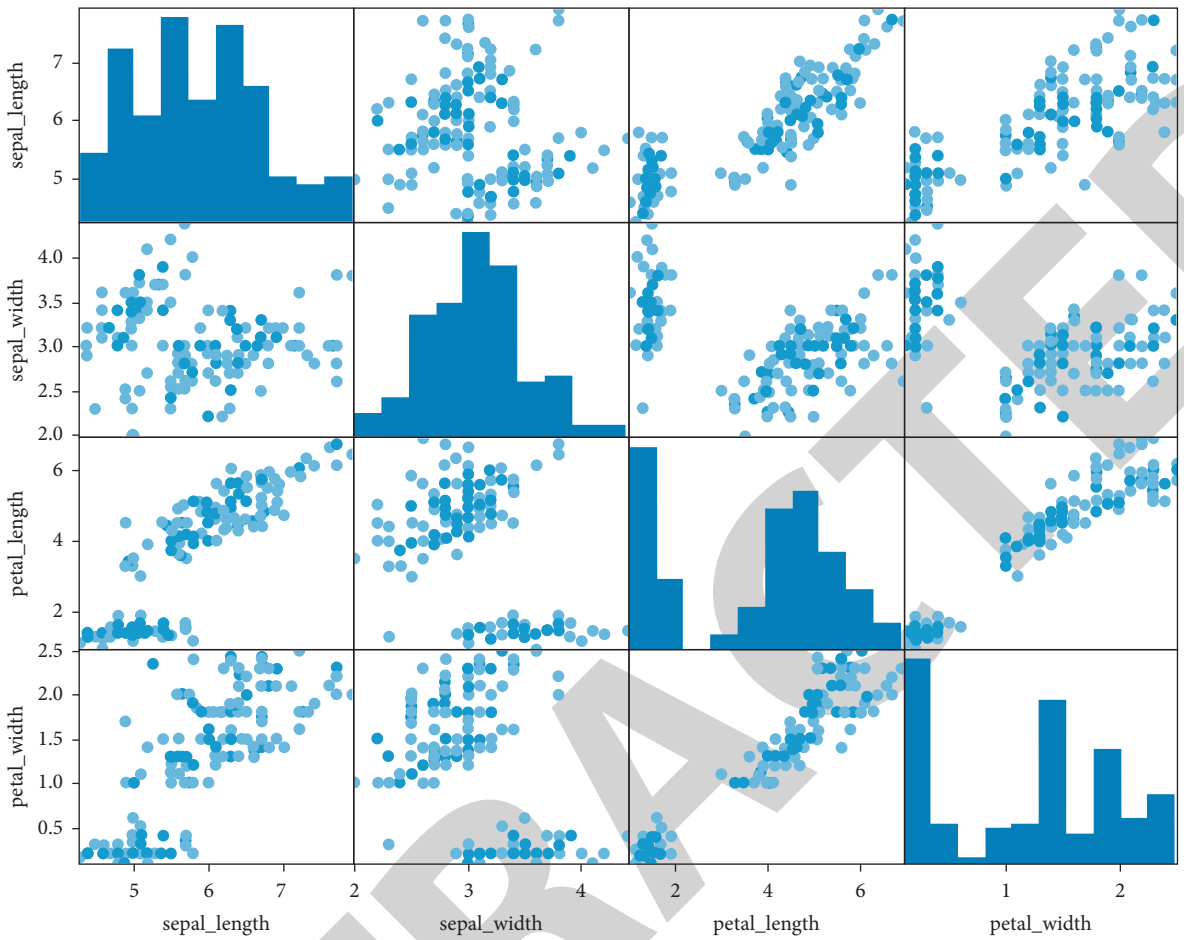

Figure 3: Histogram and scatter diagram of each attribute in the iris data set.

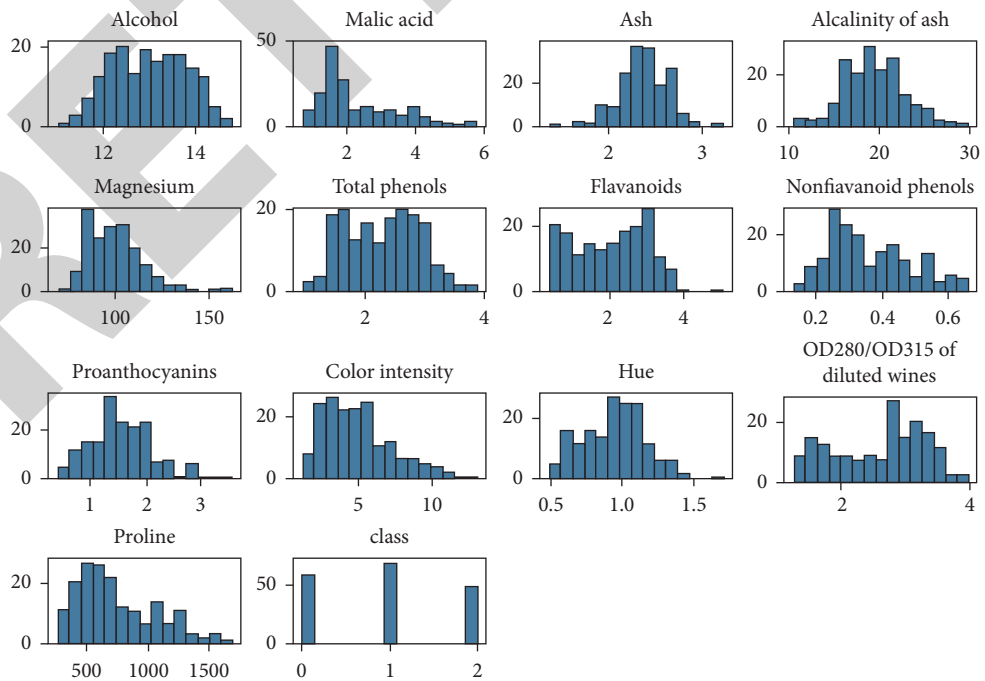

Figure 4: Wine data set.

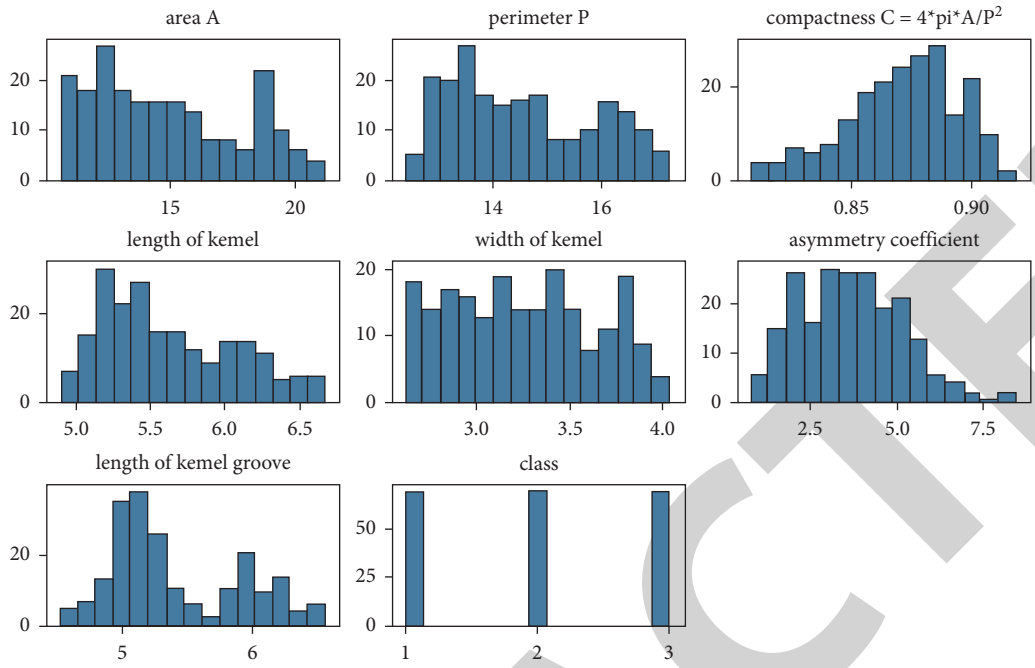

Figure 5: Wheat seed data set.

$$sd = \arg\min\left\{\left[\sum_{i=1}^{n_1}\sqrt{\sum_{p=1}^{k}\lambda_p^2\left(r_p^i - s_p^h\right)^2}\right] + \left[\sum_{j=1}^{n_2}\sqrt{\sum_{p=1}^{k}\lambda_p^2\left(r_p^j - s_p^l\right)^2}\right] + \left[\sum_{r=1}^{n_3}\sqrt{\sum_{p=1}^{k}\lambda_p^2\left(r_p^s - s_p^m\right)^2}\right]\right\}, \quad (5)$$

where n_1, n_2, n_3 are the numbers of samples of mountain iris, chameleon iris, and Virginia iris, respectively; n is the total number of samples; k is the number of attributes; the iris has four attributes of calyx length, calyx width, petal length, and petal width (so $k = 4$); and the meanings of the other parameters are given below. Table 3 illustrates the number of irises and the k value for each category, and Table 4 shows the center vectors and parameter meanings of various types of flowers.

We separate the three categories of the data set and calculate the attribute vector values of the center points under the three tags $s^h = \begin{pmatrix} 5.006 \\ 3.428 \\ 1.462 \\ 0.246 \end{pmatrix}$, $s^l = \begin{pmatrix} 5.936 \\ 2.77 \\ 4.26 \\ 1.326 \end{pmatrix}$, and $s^m = \begin{pmatrix} 6.588 \\ 2.974 \\ 5.552 \\ 2.026 \end{pmatrix}$. In the experiment, LINGO12.0 is used to solve, and the values are rounded to $\lambda_k\,(k = 1, 2, 3, 4)$ in Table 5.

4.2.2. Determining the Wine Data Set Attribute Weights. The number of sample categories in the wine data set is 3. The objective function is established according to formula (5). Data set is divided by the mean of the attributes for dimensionless processing, where n_1, n_2, n_3 are the numbers of samples under different sample categories, n is the total number of samples, and K is the number of attributes. The meaning of each parameter is given below. Table 6 lists the number and parameter significance of the three categories of the wine data set. Table 7 illustrates the three categories of wine center vector parameters.

The vector values of the attributes of the center points under the three labels are $s^h = (1.057, 0.860, 1.037, 0.873, 1.066, 1.237, 1.469, 0.801, 1.193, 1.092, 1.109, 1.209, 1.493)^T$, $s^l = (0.9444, 0.827, 0.948, 1.038, 0.947, 0.984, 1.025, 1.004, 1.024, 0.610, 1.103, 1.066, 0.695)^T$, and $s^m = (1.011, 1.426, 1.029, 1.098, 0.995, 0.731, 0.385, 1.236, 0.725, 1.462, 0.713, 0.644, 0.843)^T$.

The rounded results $\lambda_3\lambda_k\,(k = 1, 2, \ldots, 13)$ are given below. Table 8 shows the weight values.

4.2.3. Determining the Attribute Weights of the Wheat and Wheat Seed Data Set. The number of sample categories in the wheat seed data set is 3. The objective function is established according to formula (5). Data set is divided by the mean of the attributes for dimensionless processing, where n_1, n_2, n_3 are the numbers of samples under different sample categories, n is the total number of samples, k is the number of attributes, and the meanings of each parameter are as given below. Table 9 shows the parameter values needed to calculate the weight of wheat seeds.

The meanings of the other attributes are the same as in Table 7. The vector values of the attributes of the center point under the three labels are $s^h = (0.965, 0.981, 1.010, 0.978, 0.995, 0.720, 0.940)^T$, $s^l = (1.234, 1.108, 1.014, 1.092, 1.128, 0.985, 1.113)^T$, and $s^m = (0.799, 0.909, 0.975, 0.929,$

Table 3: Main parameter explanation and the determined values of the objective functions of the iris attribute weights.

Symbol	Brief explanation	Numerical value
n_1	Number of mountain iris samples	50
n_2	Number of chameleon iris samples	50
n_3	Number of Virginia iris samples	50
k	Number of data set attributes	4

Table 4: Explanation of other parameters used in solving the objective function of iris attribute weights.

Symbol	Brief explanation
s^h	Center vector of mountain iris samples
s^l	Center vector of chameleon iris samples
s^m	Center vector of Virginia iris samples
x_i	Sample attribute vector of the category

Table 5: The weight values of the iris characteristic attributes are accurately determined.

Weight	Value
λ_1	0.053
λ_2	0.117
λ_3	0.107
λ_4	0.722

Table 6: Main parameters and values of the objective functions of wine attribute weights.

Symbol	Brief explanation	Numerical value
n_1	Samples with a category of 1	59
n_2	Samples with a category of 2	71
n_3	Samples with a category of 3	48
k	Number of data set attributes	13

Table 7: Explanation of the other parameters of the objective functions of wine attribute weights.

Symbol	Brief explanation
s^h	The sample category has 1 center vector
s^l	The sample category has 2 center vectors
s^m	The sample category has 3 center vectors
x_i	The sample attribute vector of the category

$0.875, 1.294, 0.946)^T$. The rounded $\lambda_k (k = 1, 2 \ldots 7)$ results are given below. Table 10 shows the calculated weights of wheat seed attributes.

4.3. Analysis of the Experimental Results. The methods of K-means with accurately determined weights, traditional K-means, and K-means with entropy weights are used to cluster the iris, wine, and wheat seed data sets. The normalized mutual information and the confusion matrix [34] are used as evaluation criteria to evaluate the three methods.

4.3.1. Weight Entropy Method. The basic idea of the entropy weight method [35, 36] used to determine the objective weight is the index variability. Weight is determined according to the information entropy [37], which is the expectation of information content. The probability of the occurrence of a data value is negatively correlated with it. The higher the information entropy of an attribute is, the less information it can provide, the smaller the role it plays in evaluation, and the smaller its weight is. Table 11 shows the weight values of iris attributes obtained by the exact solution method. Table 12 shows the weight values of the attributes of the wine data set obtained by the exact solution method. Table 13 shows the weight values of the attributes of the wheat seed data set obtained by the exact solution method.

Table 8: The weight values of the wine characteristic attributes are accurately obtained.

Weight	Value
λ_1	0.745
λ_2	0.0043
λ_3	0.065
λ_4	0.036
λ_5	0.036
λ_6	0.017
λ_7	0.01
λ_8	0.0068
λ_9	0.0077
λ_{10}	0.0069
λ_{11}	0.023
λ_{12}	0.026
λ_{13}	0.011

Table 9: Main parameter explanation and value of the objective functions in determining wheat seed attribute weights.

Symbol	Brief explanation	Numerical value
n_1	Samples with a category of 1	70
n_2	Samples with a category of 2	70
n_3	Samples with a category of 3	70
k	Number of data set attributes	7

Table 10: Weight values of wheat seed characteristic attributes obtained by the accurate solution method.

Weight	Value
λ_1	0.0328
λ_2	0.151
λ_3	0.504
λ_4	0.133
λ_5	0.071
λ_6	0.0012
λ_7	0.104

4.3.2. Iris Data Clustering Results. The experiment is implemented in the Python 3.8.5 environment, and the maximum number of K-means iterations after inputting the attribute weight is 200. The normalized mutual information is selected as the evaluation criterion, and the confusion matrix is established. The normalized mutual information

Table 11: Weight values of iris attributes obtained by the entropy weight method.

Weight	Value
λ_1	0.193
λ_2	0.112
λ_3	0.318
λ_4	0.376

Table 12: Weight values of wine characteristic attributes obtained by the entropy weight method.

Weight	Value
λ_1	0.049
λ_2	0.123
λ_3	0.022
λ_4	0.041
λ_5	0.059
λ_6	0.066
λ_7	0.109
λ_8	0.080
λ_9	0.067
λ_{10}	0.099
λ_{11}	0.069
λ_{12}	0.091
λ_{13}	0.120

Table 13: Weight values of wheat seed characteristic attributes obtained by the entropy weight method.

Weight	Value
λ_1	0.205
λ_2	0.158
λ_3	0.07
λ_4	0.155
λ_5	0.168
λ_6	0.115
λ_7	0.126

can make the clustering results to 0-1 so that the clustering accuracy of the two methods can be seen intuitively [38]. The effect of clustering on a certain category can be obtained through a confusion matrix [39]. Clustering results can be visualized to make the results more intuitive [40]. The above methods are used to compare the results of the K-means algorithm with weights, K-means without weights, and K-means with weights determined by the entropy weight method. Table 14 shows the NMI of the iris data set after clustering by the three methods.

NMI is an external evaluation standard method for clustering [41]. By calculating the normalized mutual information of the real labels and the labels after clustering, the accuracy of clustering can be seen [42, 43]. The NMI of the three methods after clustering the iris data set is shown in Table 14. First, it can be concluded from the table that the NMI after clustering by K-means with weights is approximately 0.11 higher than that after clustering without weights. The clustering effect of K-means after determining the attribute weights is better, which confirms the feasibility of this method. Second, when the results obtained by the entropy

Table 14: NMI of the iris data set after clustering by the three methods.

Normalized mutual information (NMI)	
K-means with the exact weight	0.864
K-means without weight	0.758
K-means with the entropy weight	0.785

weight method are put into the K-means algorithm, the NMI after clustering is 0.785. It is 0.03 higher than that of traditional K-means without weights. However, the NMI after clustering of the algorithm proposed in this paper for accurately determining the weights is 0.08 higher than that of the entropy weight method. Finally, although the weight determined by the entropy weight method improves the accuracy of the iris data clustering class to a certain extent compared with clustering without weights, it is far from the improvement achieved by the weight determination method proposed in this paper. The confusion matrix after clustering is given below. Table 15 shows the confusion matrix of the effect of the three methods on iris clustering.

The confusion matrix is an effective tool for evaluating classifications and clustering criteria [44], as it can be used to clearly see in which categories the model does not perform well [45]. The confusion matrix [46, 47] after the three methods of clustering is shown in Table 15. First, it can be seen from the table that the clustering effect of the three methods is equally good for the mountain iris. These samples can be clustered accurately. All three methods are largely accurate in the category of the chameleon iris, but there is a large difference among the three in the category of the Virginia iris. K-means without weights incorrectly clustered 14 samples of Virginia iris into the category of chameleon iris. Compared with K-means clustering without weights, the improvement of K-means clustering after weight determination by the entropy weight method is not very large. Second, for the clustering of Virginia iris, the weight clustering results are almost the same as those of all attributes after weight determination by the entropy weight method. After the weights are determined by the entropy weight method, 13 Virginia irises are incorrectly clustered into the chameleon iris category, while only 14 samples are incorrectly classified even with uncertain weights. Neither method could accurately cluster Virginia irises, and it was more difficult to cluster Virginia irises than the other two iris categories. Figure 2 shows that the calyx and petal lengths and widths of the Virginia iris and chameleon iris are similar. The data are mixed and difficult to distinguish, which means that the two methods cannot distinguish the two flower categories well. The difference in the properties of the mountain iris and the other two flowers is relatively large. The weight obtained by the algorithm with the accurate solution is applied to K-means, which can distinguish the two categories well, proving the accuracy and efficiency of the method. Finally, the effect of K-means clustering determined by the entropy weight method is visualized.

Figure 6 shows the results of clustering the iris data set with the weights obtained by our method. Figure 7 shows the clustering results without attribute weights. Figure 8

Table 15: Confusion matrix of the three methods for clustering the iris data set.

	Confusion matrix		Mountain iris	Chameleon iris	Virginia iris
Accurate method	Real category y	Mountain iris	50	0	0
		Chameleon iris	0	48	2
		Virginia iris	0	4	46
Without weight		Mountain iris	50	0	0
		Chameleon iris	0	48	2
		Virginia iris	0	14	36
Entropy weights		Mountain iris	50	0	0
		Chameleon iris	0	49	1
		Virginia iris	0	13	37

shows the clustering results of the weights determined by the entropy weight method. The effect diagram after clustering shows more intuitively that some sample points are still mixed in the clustering results of chameleon iris and Virginia iris by K-means without weights. These points are not effectively divided into different clusters. However, K-means with accurately determined weights has a better effect on the clustering of the two types. Points of different categories are effectively clustered into different clusters.

4.3.3. Wine Data Clustering Results. The wine data set has more attributes than the iris data set. The results of the following three methods are compared: the K-means algorithm for calculating the weights by the exact solution method, K-means without weights, and K-means with weights determined by the entropy method. Table 16 shows the NMI values of the three methods for clustering the wine data set, and Table 17 shows the confusion matrix of the three methods for clustering the wine data set.

According to the NMI after clustering by the three methods, the method for solving the weight proposed in this paper improves the results by approximately 0.1 compared with those of the other two methods. The entropy weight method does not improve the results much in the wine data clustering class, so different weight solving methods apply to different situations. According to the confusion matrix after clustering by the three methods, the exact solution method performs better than the other two methods on the three sample categories. There is little difference between the entropy weight method and K-means without weights. Figure 9 shows the clustering results of the wine data set by the method in this paper, Figure 10 shows the clustering results without attribute weights, and Figure 11 shows the entropy weighting method clustering results.

4.3.4. Cluster Results on Wheat Seed Data. The number of attributes in the wheat seed data set is between those of the iris data set and the wine data set. The results of the weighted K-means algorithm, the K-means package in SKLearn, and weighted K-means with weights determined by the entropy weight method are compared below. Table 18 shows the NMI results of the three methods for clustering wheat seeds,

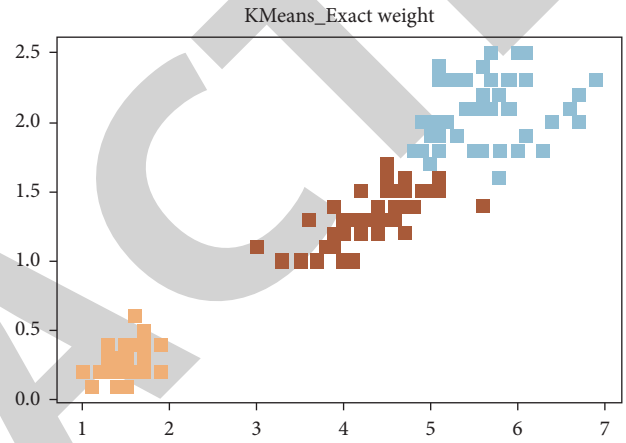

Figure 6: K-means clustering effect diagram of the iris data after the exact solution method is used to determine the weights.

and Table 19 shows the confusion matrix results after clustering.

The attribute importance of the wheat seed data set varies. Compared with the K-means clustering results without weights, the normalized mutual information after K-means clustering with weights is greatly improved. The normalized mutual information after applying the exact solution method and the entropy weight method of determining the weights is improved by 0.15 and 0.1, respectively. However, compared with the entropy weight method, the weight method proposed in this paper improves the normalized mutual information by 0.05, and the clustering effect is better.

It can be seen from the confusion matrix after clustering by the three methods that the improvement of weighted K-means compared with unweighted K-means is mainly in the data set of categories 3. The number of correct samples in the clustering of the precise solution method and entropy weight method is increased by 19 and 15, respectively, compared with that of traditional K-means. Compared with the entropy weight method, the exact solution method performs better in the clustering of category 3. Figure 12 represents the clustering result of wheat seed data by the method in this paper, while Figure 13 shows the clustering results of the unweighted data set. Figure 14 shows the clustering results of the wheat seed data set by the entropy weight method. As seen from the

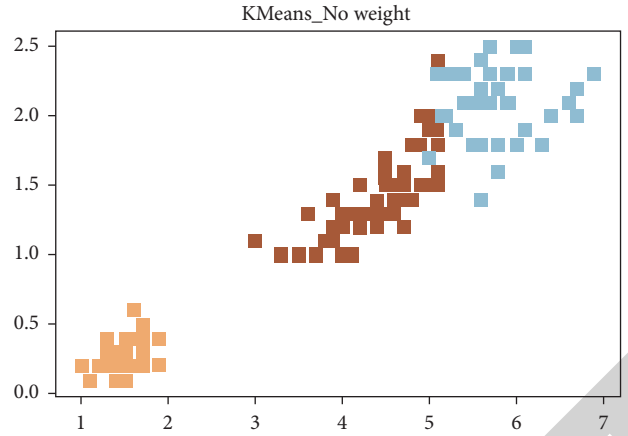

Figure 7: Effect diagram of K-means without weights on iris data clustering.

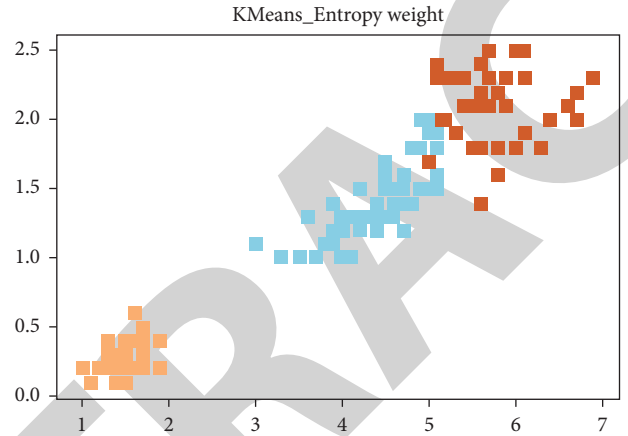

Figure 8: K-means clustering effect diagram of iris data after the entropy weight method is used to determine the weights.

Table 16: NMI aggregated by the three methods for wine data.

Normalized mutual information (NMI)	
K-means with the exact weight	0.865
K-means without weights	0.765
K-means with entropy weight	0.765

Table 17: Confusion matrix of the three methods for clustering the wine data set.

Confusion matrix			Category y_pred after K-means clustering with the exact solution method		
			Category 1	Category 2	Category 3
Accurate method	Real category y	Category 1	59	0	0
		Category 2	5	64	2
		Category 3	0	0	48
Without weight		Category 1	58	0	1
		Category 2	2	60	9
		Category 3	1	0	47
Entropy weights		Category 1	58	1	0
		Category 2	2	60	9
		Category 3	0	1	47

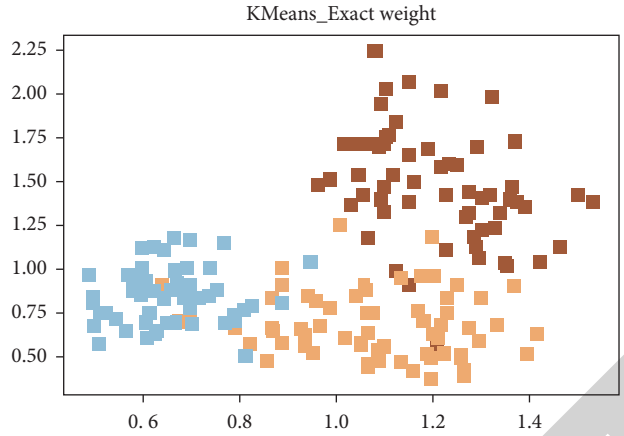

Figure 9: Effect diagram of K-means clustering of the wine data after the exact solution method is used to determine the weights.

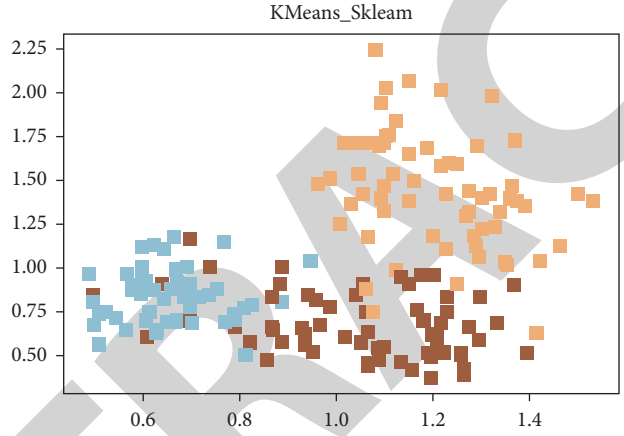

Figure 10: Effect diagram of K-means without weights on wine data clustering.

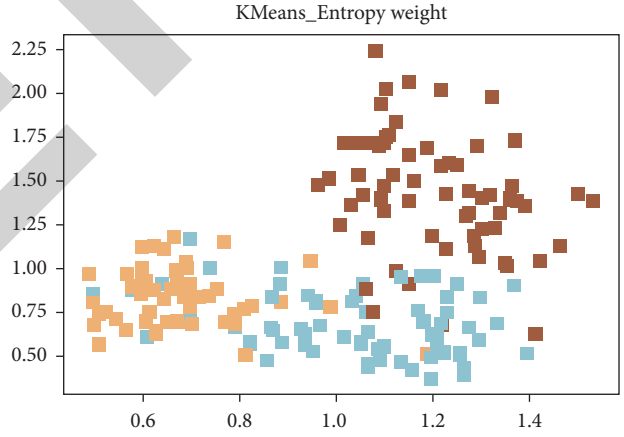

Figure 11: Effect diagram of K-means clustering of the wine data after the entropy weight method is used to determine the weights.

Table 18: NMI of wheat seed data aggregated by the three methods after classification.

	Normalized mutual information (NMI)
K-means with the exact weight	0.673
K-means without weights	0.524
K-means with entropy weight	0.621

Table 19: Confusion matrix of the three methods for clustering the wheat seed data set.

Confusion matrix		Category 1	Category 2	Category 3
Accurate method	Real category y	Category 1	55	2	13
		Category 2	9	61	0
		Category 3	1	0	69
Without weight		Category 1	58	11	1
		Category 2	10	60	0
		Category 3	20	0	50
Entropy weights		Category 1	60	10	0
		Category 2	12	57	9
		Category 3	3	2	65

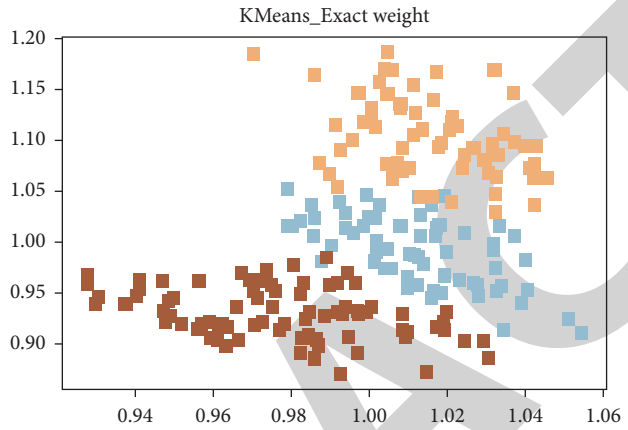

Figure 12: K-means clustering effect diagram of wheat seed data after the exact solution method is used to determine the weights.

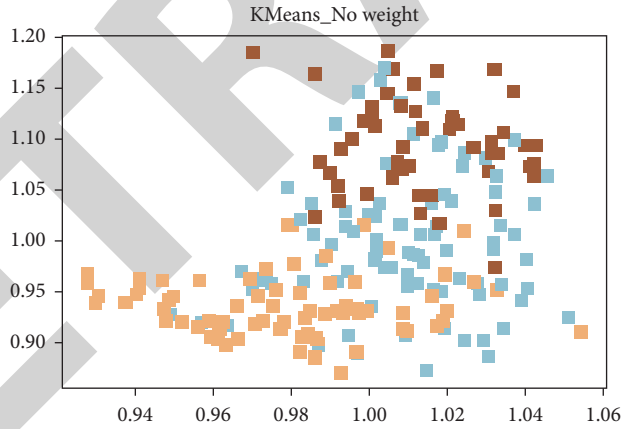

Figure 13: Effect diagram of K-means without weights on wheat seed data clustering.

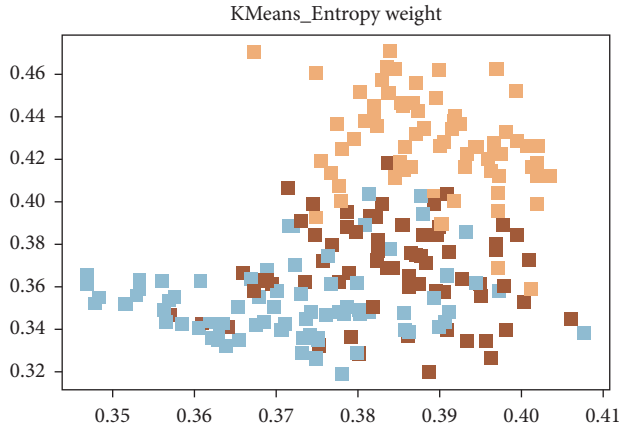

Figure 14: The K-means clustering effect of wheat seed data after the entropy weight method is used to determine the weights.

visualization, the clustering effects of the accurate solution method and entropy weight method are significantly better than that of traditional K-means. Samples of different categories are divided into different clusters.

5. Discussion and Conclusions

Class distance-based data classification algorithms are used to deal with different scenarios, where determining weights is an important and difficult problem. Based on the data value itself, this paper proposes a precisely determined distance weight, which makes the method more objective. The weight is determined only by solving the minimum function, and methods such as the entropy weight method and principal component analysis (PCA) are not needed. After determining the minimum Euclidean distance between the attribute vector of each category and the center point vector of the category to determine the weight, the obtained result is applied to the K-means clustering algorithm. Experiments were conducted using normalized mutual information as an evaluation criterion and a confusion matrix to evaluate clustering details. In this paper, we cluster the iris data set, wine data set, and wheat seed data set. The results show that, using the weight determination method proposed in this paper, confusion matrix and normalized mutual information results are better than the other two methods. Based on entropy and traditional K-means, the solution method is proven to be effective. Finally, the method is compared with the entropy weight method, to compare clustering results. The effect of the entropy weight method in determining class weight is not as good as that of the method proposed in this paper, which proves the accuracy and efficiency of this method. However, this paper only uses Euclidean distance as a distance function to measure each sample point and the center point to which it belongs. There are other distance functions in addition to Euclidean distance. Validating this approach with other distance functions is our next step. In addition, three classic machine learning data sets are taken as examples to demonstrate the effectiveness and efficiency of this method for determining weights. However, different weight determination methods are suitable for different data sets, and more verification is required for different scenarios and different data sets. For other methods, such as neural networks, further verification is required in future work. The distance-based weight determination method proposed in this paper still needs to be improved in the future, but the distance-based weight determination method is different from the subjective, entropy, and variance methods and provides a new idea for future weight determination.

Data Availability

The simulation experiment data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors have no conflicts of interest to declare.

Acknowledgments

This work was supported by the General Topics of Shanghai Philosophy and Social Science Planning (2020BGL007) and the National Natural Science Foundation of China (71832001).

References

[1] S. Zeraatkar and F. Afsari, "Interval-valued fuzzy and intuitionistic fuzzy-KNN for imbalanced data classification," *Expert Systems with Applications*, vol. 184, Article ID 115510, 2021.

[2] S. Rengasamy and P. Murugesan, "K-means–laplacian clustering revisited," *Engineering Applications of Artificial Intelligence*, vol. 107, Article ID 104535, 2022.

[3] S. Huang, Z. Kang, Z. Xu, and Q. Liu, "Robust deep K-means: an effective and simple method for data clustering," *Pattern Recognition*, vol. 117, Article ID 107996, 2021.

[4] F. D. Bortoloti, E. D. Oliveira, and P. M. Ciarelli, "Supervised kernel density estimation K-means," *Expert Systems with Applications*, vol. 168, Article ID 114350, 2021.

[5] K. S. Gyamfi, J. Brusey, A. Hunt, and E. Gaura, "A dynamic linear model for heteroscedastic LDA under class imbalance," *Neurocomputing*, vol. 343, pp. 65–75, 2019.

[6] Y. Donyatalab, F. Kutlu Gündoğdu, F. Farid, S. A. Seyfi-Shishavan, E. Farrokhizadeh, and C. Kahraman, "Novel spherical fuzzy distance and similarity measures and their applications to medical diagnosis," *Expert Systems with Applications*, vol. 191, Article ID 116330, 2021.

[7] G. Kovács, B. Nagy, and B. Vizvári, "Weighted distances on the truncated hexagonal grid," *Pattern Recognition Letters*, vol. 152, pp. 26–33, 2021.

[8] A. Panda, R. B. Pachori, and N. D. Sinnappah-Kang, "Classification of chronic myeloid leukemia neutrophils by hyperspectral imaging using Euclidean and Mahalanobis distances," *Biomedical Signal Processing and Control*, vol. 70, Article ID 103025, 2021.

[9] E. Azhir, N. Jafari Navimipour, M. Hosseinzadeh, A. Sharifi, and A. Darwesh, "An efficient automated incremental density-based algorithm for clustering and classification," *Future Generation Computer Systems*, vol. 114, pp. 665–678, 2021.

[10] Y.-L. Xu, S. Chen, and B. Luo, "A weighted locally linear KNN model for image recognition," *Communications in Computer and Information Science*, Springer, Singapore, pp. 567–578, 2017.

[11] H. R. Pourghasemi, B. Pradhan, and C. Gokceoglu, "Application of fuzzy logic and analytical hierarchy process (AHP) to landslide susceptibility mapping at Haraz watershed, Iran," *Natural Hazards*, vol. 63, no. 2, pp. 965–996, 2012.

[12] C. Lin and G. Kou, "A heuristic method to rank the alternatives in the AHP synthesis," *Applied Soft Computing*, vol. 100, Article ID 106916, 2021.

[13] M. Xia and Z. Xu, "Entropy/cross entropy-based group decision making under intuitionistic fuzzy environment," *Information Fusion*, vol. 13, no. 1, pp. 31–47, 2012.

[14] P. Chen, "Effects of the entropy weight on TOPSIS," *Expert Systems with Applications*, vol. 168, Article ID 114186, 2021.

[15] J. E. Amaya, E. Camargo, J. Aguilar, and M. Tarazona, "A proposal for a cooperative cross-entropy method to tackle the unit commitment problem," *Computers & Industrial Engineering*, vol. 162, Article ID 107764, 2021.

[16] C. Lu, D. Liang, S. Wang, L. Zeng, and Y. Zhao, "Pre-cut KNN algorithm based on threshold of distance," in *Proceedings of*

the 2018 International Conference on Network Infrastructure and Digital Content (IC-NIDC), pp. 309–314, Guiyang, China, August 2018.

[17] N. Li, H. Kong, Y. Ma, G. Gong, and W. Huai, "Human performance modeling for manufacturing based on an improved KNN algorithm," *International Journal of Advanced Manufacturing Technology*, vol. 84, no. 1–4, pp. 473–483, 2016.

[18] M. B. Afousi and M. R. Zoghi, "Wi-Fi RSS indoor positioning system using online layer clustering and weighted DCP-KNN," in *Proceedings of the Iranian Conference on Electrical Engineering (ICEE)*, pp. 710–715, Mashhad, Iran, May 2018.

[19] Q. V. Bui, K. Sayadi, S. B. Amor, and M. Bui, "Combining latent dirichlet allocation and K-means for documents clustering: effect of probabilistic based distance measures," *Intelligent Information and Database Systems*, vol. 10191, pp. 248–257, 2017.

[20] D. Wang, H. Liu, and Y. Li, "Intelligent weight generation algorithm based on binary isolation tree," *Engineering Applications of Artificial Intelligence*, vol. 109, Article ID 104604, 2022.

[21] B. Manavalan, T. H. Shin, M. O. Kim, and G. Lee, "AIPpred: sequence-based prediction of anti-inflammatory peptides using random forest," *Frontiers in Pharmacology*, vol. 9, 2018.

[22] M. Rahimi and M. A. Riahi, "Reservoir facies classification based on random forest and geostatistics methods in an offshore oilfield," *Journal of Applied Geophysics*, vol. 201, Article ID 104640, 2022.

[23] Y. Liu, Y. Li, Z. Zhang, Y. Xu, and Y. Dong, "Classification-based strategic weight manipulation in multiple attribute decision making," *Expert Systems with Applications*, vol. 197, Article ID 116781, 2022.

[24] X. Chen, S. Kar, and D. A. Ralescu, "Cross-entropy measure of uncertain variables," *Information Sciences*, vol. 201, pp. 53–60, 2012.

[25] B. Santosa, "Multiclass classification with cross entropy-support vector machines," *Procedia Computer Science*, vol. 72, pp. 345–352, 2015.

[26] M. Charrad, N. Ghazzali, V. Boiteau, and A. Niknafs, "NbClust: AnRPackage for determining the relevant number of clusters in a data set," *Journal of Statistical Software*, vol. 61, no. 6, pp. 1–36, 2014.

[27] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Transactions on Information Theory*, vol. 13, no. 1, pp. 21–27, 1967.

[28] A. C. Paola Patricia, O. C. Ana Isabel, and D.-L.-H.-F. Emiro, "Discovering similarities in Landsat satellite images using the K-means method," *Procedia Computer Science*, vol. 170, pp. 129–136, 2020.

[29] X. Zhao, F. Nie, R. Wang, and X. Li, "Improving projected fuzzy K-means clustering via robust learning," *Neurocomputing*, vol. 491, pp. 34–43, 2022.

[30] M. P. Uddin, M. A. Mamun, M. I. Afjal, and M. A. Hossain, "Information-theoretic feature selection with segmentation-based folded principal component analysis (PCA) for hyperspectral image classification," *International Journal of Remote Sensing*, vol. 42, no. 1, pp. 286–321, 2021.

[31] A. Aradnia, M. A. Haeri, and M. M. Ebadzadeh, "Adaptive explicit kernel minkowski weighted K-means," *Information Sciences*, vol. 584, pp. 503–518, 2022.

[32] A. Kumar, Y. Zhou, and J. Xiang, "Optimization of VMD using kernel-based mutual information for the extraction of weak features to detect bearing defects," *Measurement*, vol. 168, Article ID 108402, 2020.

[33] Y. Wu, J. He, Y. Ji et al., "Enhanced classification models for iris dataset," *Procedia Computer Science*, vol. 162, pp. 946–954, 2019.

[34] D. Chicco and G. Jurman, "The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation," *BMC Genomics*, vol. 21, no. 1, pp. 6–13, 2020.

[35] M. Lin, Z. Chen, H. Liao, and Z. Xu, "ELECTRE II method to deal with probabilistic linguistic term sets and its application to edge computing," *Nonlinear Dynamics*, vol. 96, no. 3, pp. 2125–2143, 2019.

[36] X. Li, Z. Wang, L. Zhang, C. Zou, and D. D. Dorrell, "State-of-health estimation for Li-ion batteries by combing the incremental capacity analysis method with grey relational analysis," *Journal of Power Sources*, vol. 410-411, pp. 106–114, 2019.

[37] B. Cao, W. Zhang, X. Wang, J. Zhao, Y. Gu, and Y. Zhang, "A memetic algorithm based on two_Arch2 for multi-depot heterogeneous-vehicle capacitated Arc routing problem," *Swarm and Evolutionary Computation*, vol. 63, Article ID 100864, 2021.

[38] J. Mou, P. Duan, L. Gao, X. Liu, and J. Li, "An effective hybrid collaborative algorithm for energy-efficient distributed permutation flow-shop inverse scheduling," *Future Generation Computer Systems*, vol. 128, pp. 521–537, 2022.

[39] B. Li, Y. Feng, Z. Xiong, W. Yang, and G. Liu, "Research on AI security enhanced encryption algorithm of autonomous IoT systems," *Information Sciences*, vol. 575, pp. 379–398, 2021.

[40] L. Zhang, H. Zheng, G. Cai, Z. Zhang, X. Wang, and L. H. Koh, "Power-frequency oscillation suppression algorithm for AC microgrid with multiple virtual synchronous generators based on fuzzy inference system," *IET Renewable Power Generation*, 2022.

[41] L. Zhong, Z. Fang, F. Liu, B. Yuan, G. Zhang, and J. Lu, "Bridging the theoretical bound and deep algorithms for open set domain adaptation," *IEEE Transactions on Neural Networks and Learning Systems*, 2021.

[42] Y. Zhang, F. Liu, Z. Fang, B. Yuan, G. Zhang, and J. Lu, "Learning from a complementary-label source domain: theory and algorithms," *IEEE Transactions on Neural Networks and Learning Systems*, 2021.

[43] Z. Wu, C. Li, J. Cao, and Y. Ge, "On scalability of association-rule-based recommendation," *ACM Transactions on the Web*, vol. 14, no. 3, pp. 1–21, 2020.

[44] Z. Wu, A. Song, J. Cao, J. Luo, and L. Zhang, "Efficiently translating complex SQL query to mapreduce jobflow on cloud," *IEEE transactions on cloud computing*, vol. 8, no. 2, pp. 508–517, 2020.

[45] W. Zheng, J. Cheng, X. Wu, R. Sun, X. Wang, and X. Sun, "Domain knowledge-based security bug reports prediction," *Knowledge-Based Systems*, vol. 241, Article ID 108293, 2022.

[46] W. Zheng, T. Shen, X. Chen, and P. Deng, "Interpretability application of the Just-in-Time software defect prediction model," *Journal of Systems and Software*, vol. 188, Article ID 111245, 2022.

[47] X. Gong, L. Wang, Y. Mou et al., "Improved four-channel PBTDPA control strategy using force feedback bilateral teleoperation system," *International Journal of Control, Automation and Systems*, vol. 20, no. 3, pp. 1002–1017, 2022.