Alexandria Engineering Journal (2022) 61, 3571–3585

HOSTED BY

ELSEVIER

Alexandria University

Alexandria Engineering Journal

www.elsevier.com/locate/aej
www.sciencedirect.com

Improved KNN algorithms of spherical regions based on clustering and region division

Haiyan Wang [a], Peidi Xu [b], Jinghua Zhao [b,*]

[a] *College of Computer Science and Technology, Jilin Provincial Key Laboratory of Human Health Status Identification and Function Enhancement, Changchun University, Changchun 130022, China*
[b] *College of Computer, Jilin Normal University, Siping 136000, China*

Received 14 August 2021; revised 2 September 2021; accepted 5 September 2021
Available online 17 September 2021

KEYWORDS

Machine learning;
Equal radius spherical region division;
KNN algorithm;
Tabu search algorithm

Abstract The KNN classification algorithm is one of the most commonly used algorithm in the AI field. This paper proposes two improved algorithms, namely KNN^{TS}, and KNN$^{TS\text{-}PK+}$. The two improved algorithms are based on KNN^{PK+} algorithm, which uses PK-Means + + algorithm to select the center of the spherical region, and sets the radius of the region to form a sphere to divide the data set in the space. The KNN^{PK+} algorithm improves the classification accuracy on the premise of stabilizing the classification efficiency of KNN classification algorithm. In order to improve the classification efficiency of KNN algorithm on the premise that the accuracy of KNN classification algorithm remains unchanged, KNN^{TS} algorithm is proposed. It uses tabu search algorithm to select the radius of spherical region, and uses spherical region division method with equal radius to divide the data set in space. On the basis of the first two improved algorithms, KNN$^{TS\text{-}PK+}$ algorithm combines them to divide the data sets in space. Experiments are carried out on the new data set and the classification results were obtained. Results revealed show that the two improved algorithms can effectively improve the classification accuracy and efficiency after the data samples are cut reasonably.

1. Introduction

KNN(K-nearest neighbor algorithm) classification algorithm is a non-parametric learning method [1]. The advantages of the algorithm are its simple principle and few influencing factors, but it also has many shortcomings, such as too much time consuming and space overhead and difficulty in choosing *K* value. That is because KNN classification algorithm does not preprocess the data before classification, but includes all the data. Therefore, many researchers are still exploring it. Wang Zhihua et al. proposed an improved K-modes KNN algorithm, it calculated the distance from the sample to the center of the cluster through string kernel function iteration, and constantly modified the center of the cluster. After the improved K- modes algorithm was used to cluster the data sets, the KNN classification model was established [2].Wang Yanfei et al. proposed an improved KNN algorithm based

* Corresponding author.
E-mail address: zjh@jlnu.edu.cn (J. Zhao).
Peer review under responsibility of Faculty of Engineering, Alexandria University.

https://doi.org/10.1016/j.aej.2021.09.004
1110-0168

on clustering and density clipping, which obtained evenly distributed data through the density clipping of data, and then carried out clustering, and then divided into several clusters, transforming globular clusters into super spheres, and finally forming a new training sample set for classification [3].Saetern et al. proposed an integrated K-nearest neighbor classification method based on the neural fuzzy method, improved the KNN algorithm through the neural fuzzy method and the new classification paradigm, and achieved good results [4]. F Lu et al. proposed an improved weighted KNN algorithm, incorporating the idea of variance into the KNN algorithm and assigning different weight values to feature items with different distributions. The improved algorithm would take longer operation time, but its classification performance was significantly improved [5]. In 2020, P. Vinaybhushan and T. Hirwarkar proposed a safe KNN classifier for encrypted data in the cloud, which ensured the privacy of data, client's information query and data arrival design [6]. Ashwini Pathak et al. used KNN in intrusion detection system [7]. In the same year, Arthur Ahmad Fauzi et al. applied KNN to the autonomous ground vehicle technology, and effectively obtained accurate classification results according to the most discriminant features [8]. This year, the research on KNN is mainly focused on the application field, such as helping with the recognition of Marathi handwritten characters [9], designing the teaching system [10], or realizing the classification of land use and land cover [11]. In particular, KNN has been applied in the medical field for many times. For example, Vaishali S. Vairale et al. used the KNN method for the recommendation of food for thyroid patients [12], and Hilal Arslan used it for the detection of the new Covid-19 [13].

KNN classification algorithm [14], namely K-nearest neighbor algorithm, is one of the most commonly used classification algorithms in the AI field. Its basic idea is: when entering new data of unknown category to be classified, the category of the data to be classified should be determined according to the category of other samples. Firstly, the characteristics of the data to be classified should be extracted and compared with the characteristics of each known category data in the test set. Then, the nearest neighbor data of K should be taken from the test set to count the categories in which most of the data are located. Finally, the data to be classified should be classified into this category.

KNN classification algorithm is set with N training samples $A = \{x_1, x_2 \ldots, x_n\}$, distributed in S categories $w_1, w_2 \ldots w_s$, N_i ($i = 1,2\ldots s$) training samples. Find K nearest samples $k_1, k_2 \ldots k_s$ out of all the samples, the discriminant function is $g_i(x) = k_i$, $i = 1,2\ldots s$, the category of sample X to be classified is determined by $g_i(x) = \text{Max}(k_i)$. The specific implementation process of KNN classification algorithm is as follows:

Step 1. The data were divided into training sample set and test sample set. The training sample set was A, $A = \{a_1, a_2 \ldots, a_n\}$, the category of the sample is expressed as S, $S = \{w_1, w_2 \ldots w_s\}$, the test sample set is X ,$X = \{x_j | j = 1,2\ldots n\}$.

Step 2. Set the initial k value as the initial neighbor of X.

Step 3. Calculate the distance between test sample points and all other training sample points.

Step 4. Sort the obtained distance in ascending order and select the appropriate k value.

Step 5. Select the closest k known samples.

Step 6. The category with the highest probability among k known samples was counted.

Step 7. Determine the category of test sample points as the category obtained in Step 6statistics.

The rest of this article is organized as follows. Section 2 discusses related work, followed by the description and the experimental analysis of the improved KNN algorithms being researched in Section 3. Section 4 reports the corresponding experimental results and data analysis. Finally, Section 5 concludes the proposed approach, and presents the future research directions.

2. Related works

2.1. PK-means++ algorithm

Clustering algorithm [15] is a kind of unsupervised learning in machine learning, among which the simplest and most basic method is the K-means algorithm in partitioning clustering algorithm. The basic idea of K-means algorithm [16] is: Among n data samples, K samples were randomly selected as the initial centers, and the distance between the other samples and the K centers was calculated. Then, according to the calculated distance, each sample is divided into the set closest to the center, that is, K clusters are formed. After that, the center of the newly formed cluster is calculated, and then the data is divided according to the new center, and the data is iterated until the center of the cluster is no longer changed. Since the k-means algorithm has the problem that the initial clustering center needs to be artificially selected, and different initial clustering may lead to different clustering results, this paper chooses the PK-Means ++ algorithm(probability K-Means ++) [17] that optimizes this problem, which is guided by local probability. It can improve the KNN classification algorithm effectively.

PK-means++ algorithm calculates the probability interval occupied by each sample by using K-means ++ algorithm. The interval here refers to the weight of the distance/total distance value in 1, the farther the distance, the greater the weight. That is the farther the point is, the greater the proportion in (0,1), the higher the probability of randomly picking this interval will be. The algorithm steps are as follows:

Step 1. Randomly select a point in the array as the center point of the first cluster;

Step 2. Iterate over all points in set D, calculate the distance from all points to the nearest cluster center, and record the data into the distance array, denoted as: D [1], D [2]...D [n].

Step 3. Put all $D[i]$ ($i = 1,2,3\ldots n$, $D[i]$ refers to the distance from the ith point to the nearest cluster center). Add the distance and Sum ($D[n]$), and calculate the probability of $D[i]$ in Sum ($D[n]$), denoted as: $P[i]$. Then, the probability $P[i]$ is expressed in (0,1) as a probability segment, and the starting point of the probability segment is stored in the array PK.

Step 4. Take the point in the interval of a random number rP($0 < \text{rP} < 1$) as the next clustering center point.

Step 5. Repeat Step 2 to Step 4 until all the initial centers of K clusters are selected.

Step 6. Continue to use the standard K-means algorithm for the next calculation.

Take the first cluster whose initial cluster center index is 4 as an example, the distance probability of each data point from the first cluster center is expressed on the interval of (0,1), as shown in Fig. 1. Where, the probability segment of the dis-

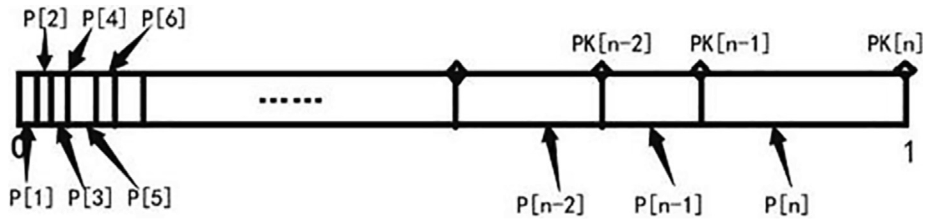

Fig. 1 Schematic of array P and array *PK*.

tance from each point to the first initial clustering center is stored in the array *P* and *P* [4] = 0.The actual point data in the probability segment (0,1) is stored in the array *PK*. If the randomly selected point can be found in the interval (*PK*[*n*-1], *PK*[*n*]), then the next data point is selected in the next clustering center.

2.2. Spherical region division algorithm with equal radius

The main method of equal radius spherical region division [18] is to divide the training set sample points in the space into multiple spherical regions with multiple spherical shapes of equal radius. First, set the radius of each spherical shape to the same R. Secondly, the distance between all sample points in the spherical region and the center of the sphere is set to be less than or equal to R. Each spherical region is then set to contain at least one sample point.

Set the training sample set as *M*, the radius of the spherical region as R, the sample set contained in the spherical region as *N*, and the number of spherical shapes formed in the space as *i* (the initial value of *i* is 1). The specific steps of the equal-radius spherical region division algorithm are as follows:

Step 1. Randomly select a training sample from *M* and make it the center of the *i*th spherical body, then delete it from *M* and add it to *N*.

Step 2. Calculate the distance between the remaining training samples in *M* and the center of the sphere. If the distance is less than the given R, delete it from *M* and add it to *N*; if the distance is greater than R, it stays in *M*.

Step 3. i can be used as the number of iterations, and $i = i + 1$.

Step 4. Loop Step 1 to Step 3 until *M* is empty and the algorithm ends.

2.3. Tabu search algorithm

Tabu Search algorithm (TS) [19,20] is one of combinatorial optimization algorithms in artificial intelligence, and it is an extension of local search algorithm.Tabu search algorithm was first proposed by Glover in the United States, and then gradually improved to form a complete algorithm.Its core idea is to avoid the local optimal situation in local neighborhood search. It is characterized by simulating the memory process of human beings and adopting Tabu technology [21,22], that is, the previous work is prohibited to avoid the local optimal situation in local neighborhood Search [23]. The idea of tabu search algorithm is to select the appropriate candidate set in the initial solution neighborhood with the given initial solution and neighborhood. So first initialize the parameters of the model, and set the tabu table to null, then determine the initial solution. If there is a case in the candidate set that the target value corresponding to the candidate solution is better than the current solution, let it replace the current solution and add the corresponding object to the tabu table for modification; if there is no qualified candidate solution, the non-tabu optimal solution is selected as the new current solution, and the corresponding object is added to the tabu table for modification. Repeat the above search steps until the termination principle (the amnesty rules) are met and then stop to get the optimal result. The flow chart of tabu search algorithm is shown in Fig. 2.

2.4. Experiments settings

In order to prove the effect of improvement, the classical KNN algorithm, KNN^{PK+} algorithm, KNN^{TS} algorithm and $KNN^{TS\text{-}PK+}$ algorithm were compared in this paper. The experiments are based on six data sets selected from the common UCI standard test database [24]. These data sets are Hayes Roth, Iris, Seeds, Pima Indians, Page Blocks and Shuttle respectively. In addition, the number of samples increased in turn. The basic informations of these six data sets are shown in Table 1 below.

In these six data sets, this paper will extract 20% data from each data set as test samples, and the remaining 80% data as training samples. As the number of data samples for each category is different, the proportion of experimental data selected from each category should be close to the proportion of the category in the overall sample number, so as to reduce the incidence of the influence of the classification results caused by the excessive number of samples of a category in the selection process.

3. Methodology

The calculation method of KNN classification algorithm is to calculate the distance between the test sample and each training sample in the sample set, and then determine the *K* nearest neighbors t of the test sample category according to the distance value obtained. However, since KNN algorithm does not process any sample data before classification, when the number of samples is too large, the classification efficiency and accuracy will be lower. Therefore, this paper improves the data pretreatment part of KNN classification algorithm.

3.1. Improved KNN^{PK+} algorithm based on PK-Means ++ spherical region division

3.1.1. Determination of initial classifier

PK-means ++ algorithm is a clustering algorithm, which aims to divide the samples in the sample data set into

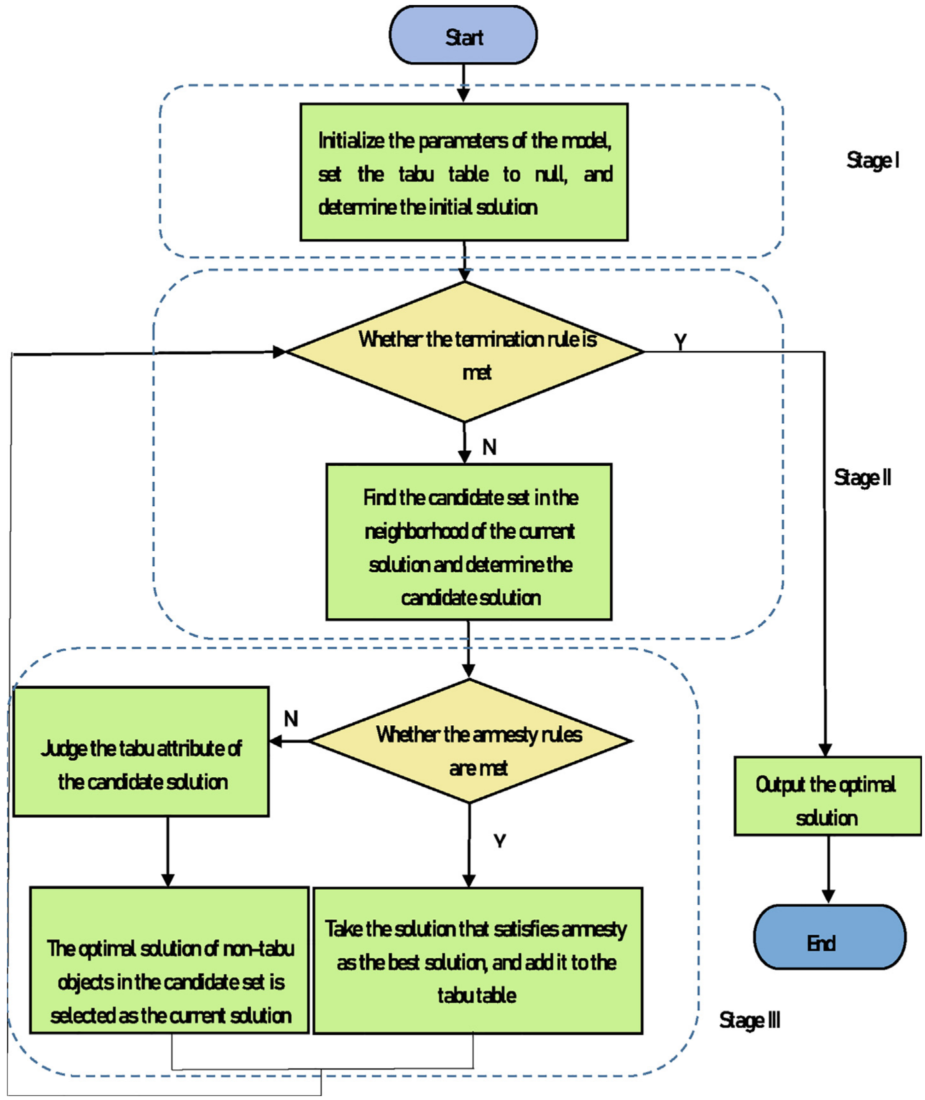

Fig. 2 Flow chart of tabu search algorithm.

Table 1 Information of six data sets.

Data set	The total number of samples	Number of attributes	Number of categories
Hayes Roth	133	6	3
Iris	150	5	3
Seeds	210	8	3
Pima Indians	769	9	2
Page Blocks	5473	11	5
Nursery	12,960	8	3
Census Income	48,842	14	2
Shuttle	58,000	10	7

several clusters, but the shape of the clusters formed by it is not regular. Since the shape of the region formed by the clustering division algorithm is similar to the sphere, in order to facilitate the calculation, this part converts the region of the cluster formed in the sample data set into multiple spherical regions. The determination process of the initial classifier is as follows:

Step 1. Calculate the centroid vectors of each region in the sample data through PK-means+ + algorithm, and select the appropriate initial center.

Step 2. Calculate the distance of all training samples in the data set to each center, and put them into the cluster with the closest distance.

Step 3. Increase the training samples constantly, and update the center point of the cluster timely.

Step 4. Calculate the sum of squared errors. When the sum of squared errors is no longer reduced and the samples contained in the cluster are basically unchanged, the updating of samples in the cluster is ended.

Step 5. Take the centroid vector of each cluster as the centroid of the spherical region, calculate the distance from other samples to the centroid, and take the farthest distance as the radius of the spherical region.

Step 6. Save the samples contained in the formed spherical region and use them as the initial classifier.

3.1.2. Steps of KNN^{PK+} algorithm

First, PK-Means + + algorithm is used to select the center of the spherical region, then an initial classifier is constructed for the training set according to the center and corresponding radius, and then a new training set containing K nearest neighbor training samples is determined through continuous calculation of the classifier. Finally, KNN algorithm is used in the new training set. The steps of KNN^{PK+}[25] algorithm are as follows:

Step 1. The center point of the spherical region is obtained by using PK-Means + + algorithm.

Step 2. Calculate the distance between the center point of each spherical region and other samples, and store it in array D. Then arrange all the values in D in descending order, and take the farthest distance as the radius of the spherical region to form the initial classifier.

Step 3. Calculate the distance of the sample to be tested to each spherical region and record the maximum distance value.

Step 4. The new training set S is initially set to be empty. If the distance is less than 0 in the calculation process, all samples in the region will be added to the new training set.

Step 5. Add all samples contained in the closest spherical region to the new training set S.

Step 6. Decide whether to continue or not. If the distance between the sample to be tested and the adjacent K samples is less than the distance between it and the spherical region without adding the new training set, then the calculation is terminated, otherwise, go to Step 1.

Step 7. The test samples were classified by KNN algorithm in S.

3.1.3. The experimental results of algorithm KNN^{PK+}

The aim of this part is to improve the classification accuracy of KNN on the premise of stabilizing the classification efficiency. Therefore, this section will analyze and compare the running time and classification accuracy of the algorithm and draw the final conclusion. Classification experiments were carried out on the six data sets in UCI. The initial value of K is set to 1, and then it increases by 1 each time. Continuously perform classification calculations and then record classification accuracy. If the K value is still increasing but the accuracy is no longer changing significantly, select the K value. The experimental results are as follows.

It can be seen from Table 2 and Fig. 3 that the classification accuracy of KNN^{PK+} algorithm is significantly higher than that of classical KNN classification algorithm. There has also been a decrease in classification time, but not by much. The reason is that the classical KNN algorithm does not perform any preprocessing on the received data, so some useless or fuzzy data may affect the classification accuracy. By determining the center of the sphere used to divide the spherical region, algorithm KNN^{PK+} processed the data and then obtained a new data set more suitable for classification, which can effectively avoid the error clipping of the effective data in the data set. Therefore, KNN^{PK+} algorithm can effectively improve the accuracy of classification.

Algorithm KNN^{PK+} can remove useless samples and thus reduce the number of training samples. Then, the center of the spherical region is found and all the selected centers are made optimal, which can avoid deleting the effective data of the edge during sample clipping, thus effectively improving the classification accuracy. However, when region division is carried out by Algorithm KNN^{PK+}, the radius of the region selects the farthest distance within the region, which leads to serious overlap between regions. Repeated calculation of many data will increase the amount of calculation. Therefore, although the classification time required by the algorithm is reduced, the classification time is still not optimal.

Table 2 Comparison of classification accuracy of the two algorithms(%).

	Classical KNN algorithm	KNN^{PK+} algorithm
Hayes Roth	93.0	98.1
Iris	94.1	98.2
Seeds	85.6	89.7
Pima Indians	83.7	90.1
Pageblocks	85.7	91.7
Shuttle	83.6	89.6

3.2. Improved KNN^{TS} algorithm based on TS-equal radius spherical region division

In the previous part, although algorithm KNN^{PK+} can effectively improve the accuracy of classification, it does not greatly improve the classification time. Therefore, on the basis of KNN^{PK+} algorithm, the research team considers another way to improve the KNN algorithm. That is, on the premise that the accuracy of KNN classification algorithm is not changed, the classification efficiency of the algorithm is improved.

Through the analysis of algorithm KNN^{PK+} in the previous part, it can be seen that there is too much overlap between multiple spherical regions in the space when dividing the data set. If the classification is carried out directly, a large number of redundant data will be generated, which increases the amount of computation. Therefore, this section will improve the problem of large overlap between regions. By using the spherical region division method with equal radius and the tabu search algorithm to divide the data set, the classification efficiency of KNN algorithm will be improved.

3.2.1. Tabu search method to solve the radius R value

After the simple division of the training samples, an initial classifier should be constructed. The initial classifier is determined by the initial radius. In all spherical regions in space, the center of the spherical region is found and the distance from the center of all sample points is calculated. The radius is set as the farthest distance calculated, that is, the maximum distance to the center in the spherical region is kept as the initial radius and all training samples contained therein as the initial classifier. It is very important to select the radius of the spherical region. Too large or too small radius may directly affect the overlap between spherical regions in the space, as well as the number of samples cropped. Therefore, in the calculation after the initial classifier is determined, the radius R value should be selected as an appropriate value to ensure that the number of spherical objects in the space is reasonable.

To determine the radius R value of a spherical region of equal radius, it is necessary to first find the optimal number of spherical regions in space [26].If the number of training samples in the original training set is n and the number of

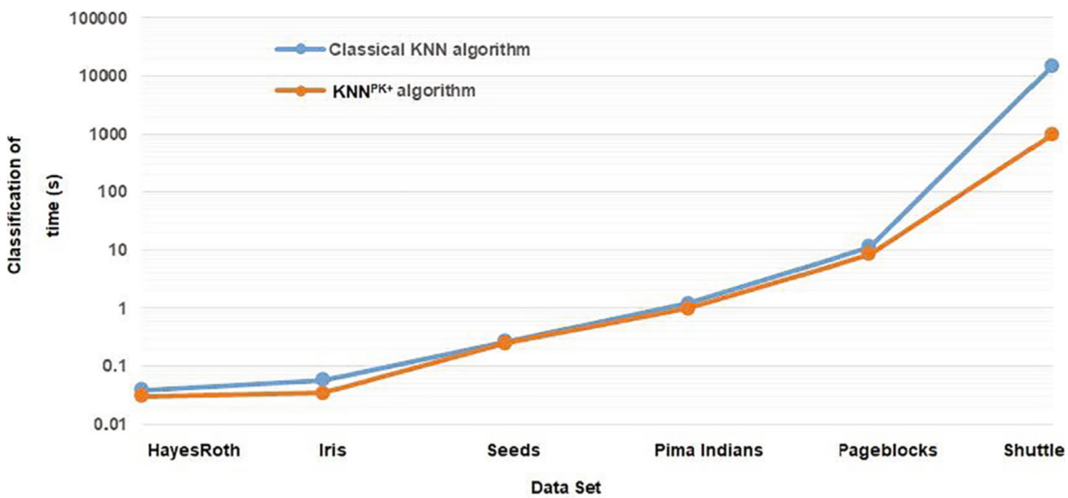

Fig. 3 Classification time comparison of the two algorithms.

spherical regions is s, then the average number of samples contained in each spherical region is n/s. Calculate the distance between the sample to be tested and the center of each spherical region. Calculate s times in total. After calculating the distance, the number of spherical regions that can be added to the new training set is determined. Assuming that the data in m spherical regions can be added to the new training set, the number of samples in the new training set is mn/s. Since the processed new training set samples are used in the classification of KNN algorithm, the new training set samples need to calculate the mn/s sub-distance. If the time required to calculate a distance is 1 s, then the calculation time of KNN classification algorithm is as follows:

$$f(s) = s + mn/s \tag{1}$$

If the optimal number s of spherical body in the region is required, $f(s)$ needs to be minimized. Therefore, the derivative of $f(s)$ can be obtained as follows:

$$f(s) = 1 - mn/s^2 \tag{2}$$

It is calculated that when $S = \sqrt{mn}$, the minimum value of $f(s)$ is 0, and the optimal number of spherical regions in the region is $\sqrt{mn}$. Tabu search algorithm is used to find the optimal R value of the selected spherical regions with equal radius. The objective function is the absolute value of the difference between square root $\sqrt{mn}$ and the actual number of spherical regions. The steps to solve the radius R value using tabu search algorithm are as follows:

Step 1. Set the training set as Y, the number of training samples in the training set as n, the number of iterations as i, the initial feasible solution as R_{now}, and the initial solution as the current solution (namely the current optimal solution R_{best}); initialize tabu T and empty it. The objective function is abs (R_{now}).

Step 2. If the termination rule is satisfied, stop calculating and go to Step 8; otherwise, go to Step 3.

Step 3. In the neighborhood of the initial solution R_{now}, a 2-OPT operation is performed to select an untabu candidate solution or a tabu solution that satisfies the amnesty rule, and the evaluation value of the solution is better than that of the current solution R_{next}, so that $R_{now} = R_{next}$, and the tabu table is updated.

Step 4. In the neighborhood of the initial solution R_{now}, a 3-opt operation is performed to select an untabu candidate solution or a tabu solution that satisfies the amnesty rule, and the evaluation value of the solution is better than that of the current solution R_{next}, making $R_{now} = R_{next}$, and updating the tabu table.

Step 5. Repeat Step 3 and Step 4 until all solutions in the R_{now} neighborhood are taboo and cannot be forgiven.

Step 6. Search for the best solution that is not taboo and is superior to R_{next} in the neighborhood of R_{now}. If the objective function is less than 0 and abs (R_{now}) is superior to abs (R_{best}), make $R_{best} = R_{now}$ and turn to Step 7.

Step 7. Update tabu T to make the number of iterations i = i + 1, and then go to Step 2.

Step 8. Output the optimal solution R_{best} and the algorithm ends.

After the algorithm is finished, the output R_{best} is the optimal radius R value. The contents of Table 1 are the optimal number of spherical regions solved by each data set in the experiment, the radius R value solved by tabu search algorithm, and the error value of the number of spherical regions actually formed and the number of spherical regions during region division.The calculation formula of the error value of the number of spherical regions is as follows:

$$\textit{The error value} = \frac{\textit{The optimal number of spherical bodies} - \textit{The actual number of spherical bodies}}{\textit{The optimal number of spherical bodies}} \tag{3}$$

According to the error value of the number of spherical regions in Table 3, it can be seen that the actual number of spherical regions differs little from the optimal number. Number is ideally in solving the optimal number of spherical region formed in the space of not considering the distribution of actual data, so the actual division may be due to data centralized data more dispersed and the actual number is more than the optimal number, or data aggregation and number less than the optimal number of actual condition. Although there is a little error in the number of spherical regions, the radius R value of spherical regions is the local optimal value, so the sample data obtained after the actual division is relatively accurate.

Table 3 Number of spherical regions and R value of radius.

	The optimal number of spherical regions	The actual number of spherical regions	Radius value R	Error value of number of spherical regions
Hayes Roth	16	15	0.27	0.0625
Iris	16	16	0.243	0
Seeds	24	24	0.43	0
Pima Indians	46	48	0.47	−0.042
Pageblocks	114	113	0.166	0.009
Shuttle	427	420	0.021	0.016

3.2.2. 3.2.2. Steps of KNN^{TS} algorithm

First, an initial classifier is constructed for the training set by using the spherical region division method with equal radius, as shown in Fig. 4. As can be seen from the figure, the overlap has been greatly reduced, and the number of spherical bodies has also been reduced. And then a new training set with redundant data removed is calculated continuously. Then, KNN algorithm is used in the new training set. Suppose the original training set is S_1, the result training set is S_2, the new training set is S_3, the number of samples is n, the number of iterations is i, and the distance matrix is Dis. The steps of the algorithm are as follows:

Step 1. Tabu search algorithm is used to obtain the optimal radius R value;

Step 2. Store the sample from the divided spherical region of equal radius in S_2, and set $i = 0$;

Step 3. After obtaining the spherical region divided according to the radius R value, a point in the spherical region is randomly selected as the center, and the sample distance R_{new} in the ith spherical region is calculated, and R_{new} is taken as the new radius of the spherical region, and $i = i + 1$;

Step 4. repeat Step 3. If $i > n$, then stop the cycle, and store the data sample in the adjusted spherical region in S_2 to form the initial classifier;

Step 5. Calculate the distance between the sample to be tested and each spherical region, store the distance in Dis, and then arrange the data in Dis in ascending order.

Step 6. Determine the spherical region of K near-training samples and add the training samples contained in the region to S_3.

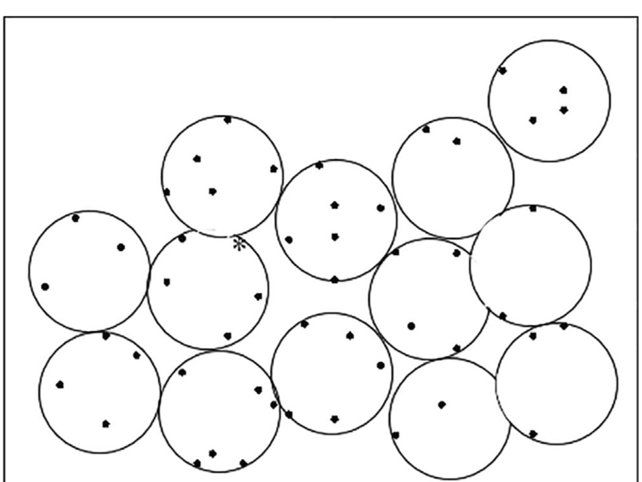

Fig. 4 Example diagram of the initial classifier.

Step 7. In the new data set S_3, KNN algorithm is used to classify the test samples.

The main task of Step 6 and Step 7 above is to determine the training samples of the new training set. The main process is shown in Fig. 5.

3.2.3. 3.2.3. Experimental results of KNN^{TS} algorithm

Based on our assumptions, the purpose of this section is to improve the classification efficiency of KNN algorithm on the premise that the accuracy of KNN classification algorithm remains unchanged. Experiments are still carried out on the six data sets selected from the UCI database. This part discusses the differences in classification time and accuracy between the classical KNN algorithm, the common spherical region division KNN algorithm with equal radius and the KNN^{TS} algorithm. In the course of the experiment, the Classical KNN algorithm receives all the data without any processing, calculates all the data, and then obtains the classification result. Classical KNN algorithm for spherical region division with equal radius performs simple region division processing in advance, and randomly selects the radius and center of spherical region. KNN^{TS} algorithm also performs pre-region division processing on data, but in addition to randomly selecting the center of the sphere, the regional radius is set through the local optimal value determined by tabu search algorithm. After many experiments, the results of classification time and classification accuracy are as follows:

Table 4 to Table 5 respectively show the experimental results of Classical KNN algorithm, Classical KNN algorithm for spherical region division with equal radius, and KNN^{TS}.

As can be seen from Table 4, the classification accuracy of classical KNN classification algorithm is high, but the time required for classification is significantly longer when the number of samples keeps increasing.Especially for Shuttle data set, the number of data samples was 58000, and the classification time reached 14,526.97 s.

Table 5 shows that when Classical KNN algorithm for spherical region division with equal radius is used for classification, the classification time is much less than that of Classical KNN Algorithm, and the accuracy is basically unchanged. However, for the data set Shuttle with a large sample number, the classification time is still long due to the large number of sample data.Due to the inaccuracy of the radius selected by the initial classifier, the number of spheres formed in the space is also uncertain, resulting in many repeated distance calculations during the determination of the new training set, which seriously affects the classification efficiency.

Meanwhile, the radius of the spherical region is calculated when KNN^{TS} algorithm is used, and the required classification

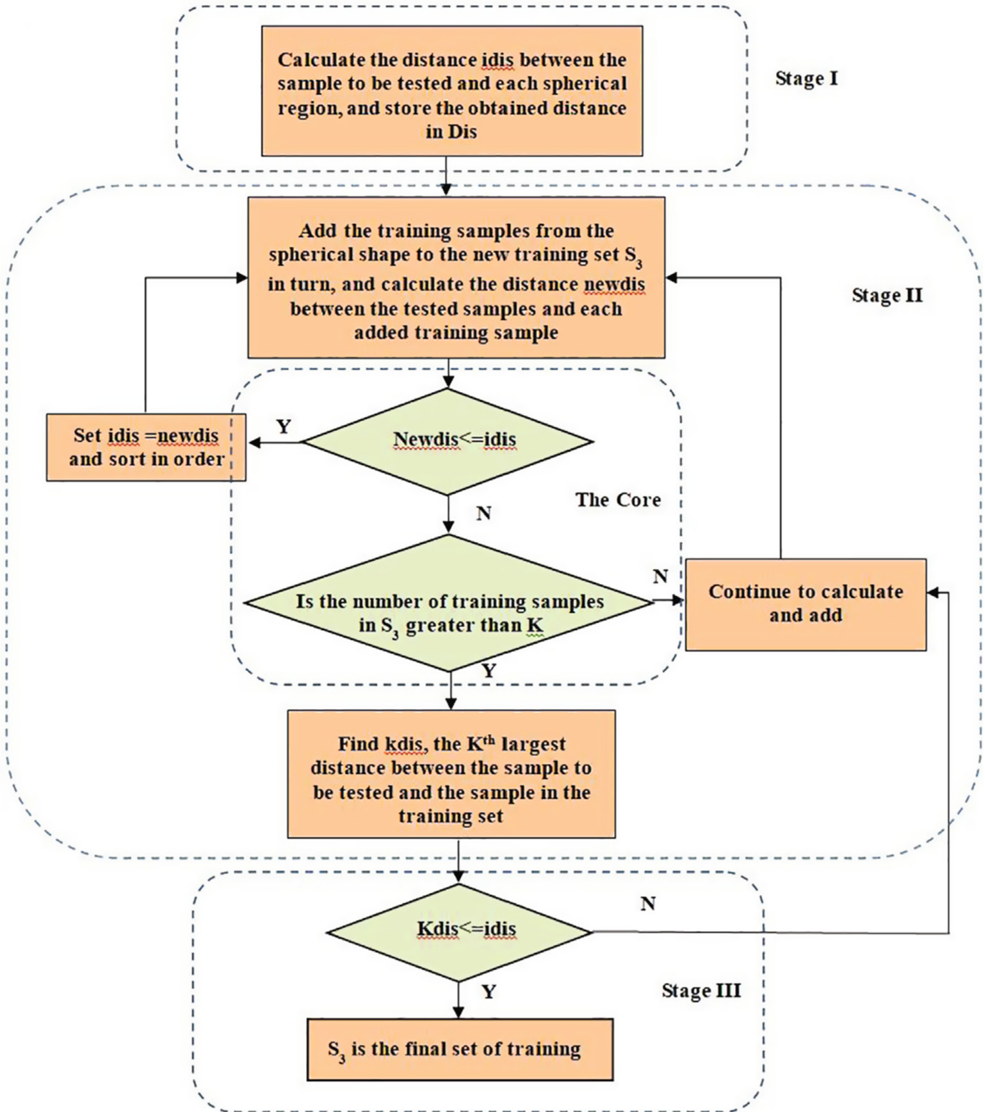

Fig. 5 Flow chart of identify new training sets.

time is significantly reduced. The classification time is about half that of the classical KNN algorithm, and the accuracy fluctuated little.

In order to more intuitively reflect the differences in the classification accuracy and classification time of the three classification methods, we further compared the classification accuracy and classification time, and made Fig. 6 and Table 6. Fig. 6 shows the comparison of classification accuracy of the three algorithms, and Table 6 shows the comparison of classification time. In order to illustrate the advantages of the improved algorithms, Fig. 7 is specially made to show the comparison of the improvement rates of classification time. Each section of these figures respectively records the percentage improvement of the classification time of the common spherical region division KNN algorithm with equal radius compared with that of classical KNN algorithm(left part) and the percentage improvement of the classification time of KNN^{TS} algorithm compared with that of classical KNN algorithm(right part).

It can be clearly seen from Fig. 6 that the classification accuracy of the common spherical region division KNN algorithm with equal radius and KNN^{TS} algorithm is basically unchanged after improvement, which is basically the same as that of classical KNN algorithm. By Table 6, you can see that after using the improved algorithm, the classification times of the common spherical region division KNN algorithm with equal radius and KNN^{TS} algorithm compared with the classical KNN algorithm have greatly improved. Moreover, the percentages of improvement rate are marked as the blue part of

Table 4 Results of Classical KNN Algorithm.

	K-value	Accuracy	Classification time
Hayes Roth	2	0.93	0.039 s
Iris	2	0.941	0.057 s
Seeds	3	0.856	0.263 s
Pima Indians	3	0.837	1.189 s
Pageblocks	3	0.857	11.419 s
Shuttle	6	0.836	14526.97 s

Table 5 Results of Classical KNN algorithm for spherical region division with equal radius and KNN^{TS} Algorithm.

	K-value	Radius	Accuracy	Classification time
Hayes Roth	2	0.3	0.93	0.02 s
	2	0.27	0.93	0.017 s
Iris	2	0.5	0.941	0.041 s
	2	0.243	0.941	0.035 s
Seeds	3	0.3	0.854	0.176 s
	3	0.43	0.856	0.128 s
Pima Indians	3	0.6	0.835	0.512 s
	3	0.47	0.836	0.455 s
Pageblocks	3	0.52	0.857	8.548 s
	3	0.166	0.854	3.359 s
Shuttle	6	0.4	0.835	8645.635 s
	6	0.021	0.836	784.63 s

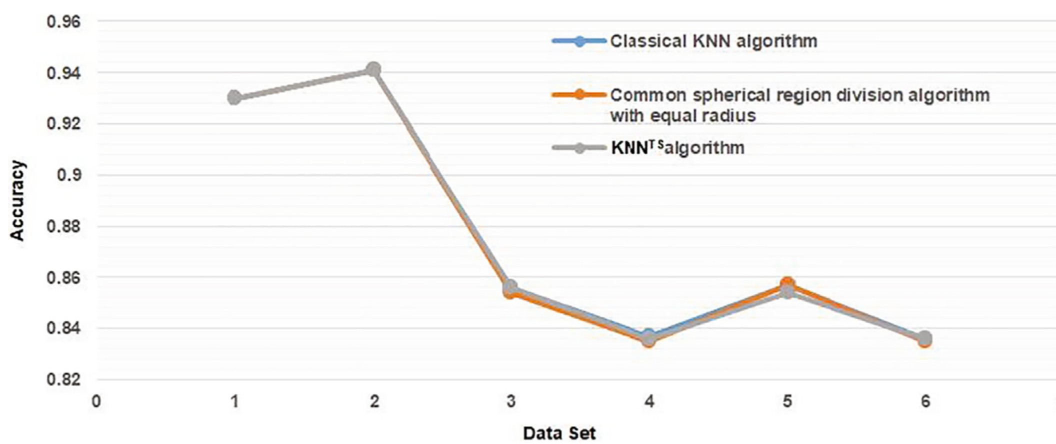

Fig. 6 Classification accuracy comparison of the three algorithms.

Table 6 Classification time comparison of three algorithms (Unit: millisecond, ms).

	Hayes Roth	Iris	Seeds	Pima Indians	Pageblocks	Shuttle
Classical KNN	39	57	263	1189	11,419	14,526,970
Common spherical regions division KNN	20	41	176	512	8548	8,645,635
KNN^{TS}	17	35	128	455	3359	784,630

Fig. 7. From these values, it can be further seen that KNN^{TS} algorithm can save more time than the common spherical region division KNN algorithm with equal radius. Especially when the number of samples keeps increasing, the time required for classification is significantly reduced and the improvement rate gradually increases.

The reason is that the classical KNN algorithm does not do any processing on the data and computations on all the received data, which results in too long classification time. Although KNN algorithm for spherical regions with equal radius can reduce the classification time, it still takes a long time because it cannot determine the appropriate radius and needs to carry out distance calculation for many times. The KNN^{TS} algorithm avoids the above problems and reduces the overlapping between regions, thus reducing the required time. Therefore, KNN^{TS} algorithm can effectively improve the classification efficiency while ensuring the classification accuracy is basically unchanged.

KNN^{TS} algorithm can calculate the appropriate radius required in the classified spherical region, solve the problem of partial data duplication caused by excessive overlap between regions, and effectively improve the classification efficiency of the algorithm without reducing the classification accuracy. However, KNN^{TS} algorithm selects the center of the spherical region randomly in the regional division, which may lead to errors in the effective sample data and affect the accuracy of classification.

3.3. Improved KNN algorithm $KNN^{TS\text{-}PK+}$ based on clustering and regional division

As can be seen from the above two parts, KNN^{PK+} algorithm can remove useless samples and reduce the number of training samples, find the center of the spherical region and make the center selected optimal, and effectively improve the classifica-

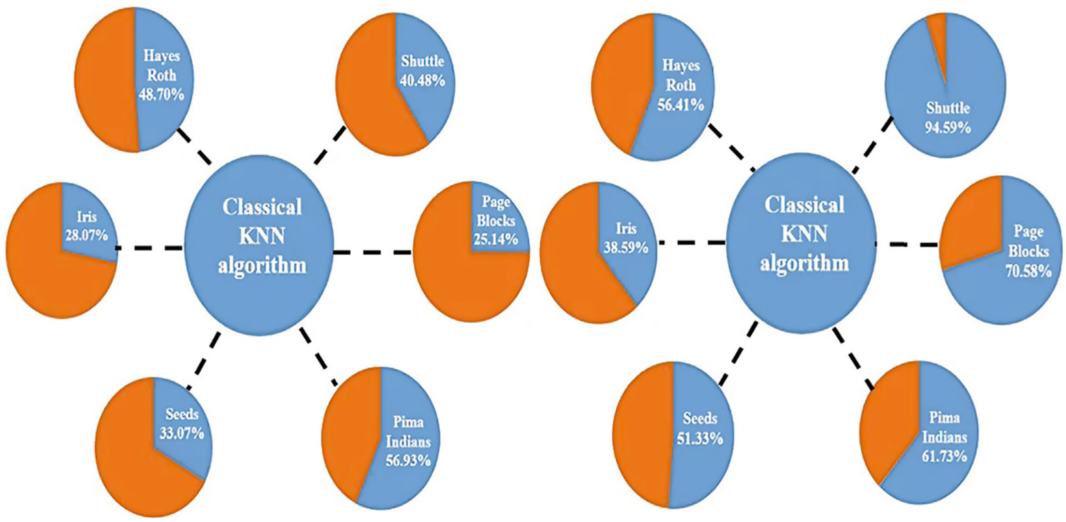

Fig. 7 Comparison of classification time improvement rates of the two algorithms.

tion accuracy. However, the disadvantage is that the radius of the region selects the farthest distance within the region, leading to a lot of overlap between regions, and repeated calculation of many data will increase the amount of calculation. KNN^{TS} algorithm can calculate the appropriate radius required in the classified spherical region, solve the problem of partial data duplication caused by excessive overlap between regions, and effectively improve the classification efficiency of the algorithm without reducing the classification accuracy. However, the disadvantage is that the center of the spherical region is randomly selected, and some valid sample data may be misclipped, thus affecting the accuracy of classification.

Aiming at the above problems, this paper combines the two methods and proposes an improved KNN algorithm KNN$^{TS\text{-}PK+}$ based on clustering and regional division. First, the equal radius spherical region algorithm is used to divide the sample data in the space into several spherical regions, then PK-means+ + algorithm is used to find the center of the spherical region, and then tabu search algorithm is used to find the radius of the spherical region to form an appropriate spherical region. Then, the original sample data set is clipped with iterative operation to form a new sample data set suitable for KNN algorithm. Finally, KNN algorithm is used to classify the processed new sample data set, so as to improve the accuracy of classification and the efficiency of classification.

3.3.1. Steps of KNN$^{TS\text{-}PK+}$ algorithm

Assuming that the original training set is S, the number of samples is N, the number of iterations is i, and the new training set is H. Store the samples in the divided spherical region with equal radius in H, and let $i = 0$. The specific steps of the algorithm are as follows:

Step 1. PK-Means + + algorithm is used to determine the initial center point of the spherical region.

Step 2. Tabu search algorithm was used to calculate the optimal radius value of the spherical region.

Step 3. The R value obtained in Step 2 and the center point obtained in Step 1 were respectively taken as the radius and center of the spherical region division method with equal radius, and the sample data was divided into several regions, and the training samples in the regions were saved in train[i].

Step 4. repeat Step 1 to Step 1. If $i > n$, stop the cycle and stores the data samples in the adjusted spherical region in H to form the initial classifier.

Step 5. Calculate the distance between the samples to be tested and each spherical region, store the distance in D, and arrange the data in D in ascending order.

Step 6. Determine the spherical regions adjacent to the training samples, and add the training samples in these K regions to the new training set.

Step 7. In H, KNN algorithm is used to classify test samples.

By combining the two methods, the overlap probability in sample region partition can be effectively reduced.In addition, because appropriate center points and radius can be selected, the original training sample can be clipped and the problem of partial valid data being clipped by mistake can be avoided.As the number of samples in the final new training set is reduced, the classification accuracy can be improved and the classification time can be reduced.

3.3.2. Experimental results for KNN$^{TS\text{-}PK+}$ algorithm

In order to prove the superiority of the improved KNN algorithm, 10 experiments were conducted on six sample data sets, and the classification time and classification accuracy of each experiment were recorded. Take the average of these 10 accuracy rates for the final comparative analysis. The experimental results are as follows.

As can be seen from Table 7, KNN$^{TS\text{-}PK+}$ algorithm has a high accuracy in classification calculation. Although the accuracy varies from one experiment to another in the ten experiments, the overall fluctuation range is not large, that is, the classification accuracy is relatively stable.

Table 8 lists the comparison of classification accuracy between classical KNN algorithm and KNN$^{TS\text{-}PK+}$ algorithm, and Table 9 shows the classification time proportion of KNN$^{TS\text{-}PK+}$ algorithm and classical KNN algorithm. As can be seen from Table 8, compared with the classical KNN

Table 7 The 10 accuracy rates and average results of $KNN^{TS\text{-}PK+}$ algorithm (%).

	1	2	3	4	5	6
Hayes Roth	97.8	97.7	97.5	96.5	97.9	97.0
Iris	97.3	96.8	96.6	97.0	96.1	96.3
Seeds	86.9	87.0	86.9	88.0	87.2	86.4
Pima Indians	89.2	88.6	89.3	89.0	88.8	89.2
Page Blocks	87.5	87.4	87.1	87.3	87.7	87.4
Shuttle	85.0	84.4	84.3	84.9	84.7	84.6
	7	8	9	10	Average	
Hayes Roth	96.8	98.1	97.7	98.0	97.5	
Iris	96.9	97.1	96.7	96.2	96.7	
Seeds	86.7	87.3	87.5	87.1	87.1	
Pima Indians	88.6	88.2	89.1	89.0	88.9	
Page Blocks	87.3	87.2	86.9	87.2	87.3	
Shuttle	85.0	84.8	84.9	84.4	84.7	

Table 8 Comparison of classification accuracy (%).

	Classical KNN algorithm	$KNN^{TS\text{-}PK+}$ algorithm
Hayes Roth	93.0	97.5
Iris	94.1	96.7
Seeds	85.6	87.1
Pima Indians	83.7	88.9
Pageblocks	85.7	87.3
Shuttle	83.6	84.7

Table 9 Classification time comparison of the two algorithms.

	Classical KNN algorithm	The proportion of $KNN^{TS\text{-}PK+}$ algorithm
Hayes Roth	1	53.85%
Iris	1	70.18%
Seeds	1	74.14%
Pima Indians	1	56.09%
Pageblocks	1	57.59%
Shuttle	1	5.88%

algorithm, the classification accuracy of $KNN^{TS\text{-}PK+}$ algorithm is significantly improved. In the case of a small number of data samples, the classification accuracy of $KNN^{TS\text{-}PK+}$ algorithm is relatively high, while on Pageblocks and Shuttle data set with a large number of data samples, the classification accuracy of $KNN^{TS\text{-}PK+}$ algorithm is higher than that of classical KNN algorithm, but the scale of the improvement is less obvious. It can be seen from Table 9 that the classification time of $KNN^{TS\text{-}PK+}$ algorithm is about more than half of that of the classical KNN algorithm when the data samples are relatively small, but only about one tenth of it when the data samples are relatively large.

The classical KNN algorithm does not discriminate the data to be classified before classification calculation, but calculates all the data, which leads to excessively long classification time. In this way, the received interference data directly leads to the decrease of classification accuracy. $KNN^{TS\text{-}PK+}$ divides and cuts out all the data before calculation, which can effectively remove invalid data. This makes it more likely to select and retain valid data suitable for calculation, so classification accuracy and efficiency are greatly improved.

4. Experimental results and analysis

In this paper, the experimental results of the classical KNN algorithm, the KNN^{PK+} algorithm, the KNN^{TS} algorithm, and the $KNN^{TS\text{-}PK+}$ algorithm are collated and compared. The basic information of these eight data sets is shown in Table 10 below. Classification accuracy and classification efficiency are still taken as measurement standards. Fig. 8 and Fig. 9 can more intuitively show the comparison effect of classification accuracy and classification time of these four algorithms.

Fig. 8 compares the classification accuracy of the novel improved algorithms and the classical KNN algorithm on the same data set. In the previous experimental results, it can be seen that the classification accuracy of KNN^{PK+} algorithm is higher than that of classical KNN algorithm, and the classification accuracy of KNN^{TS} algorithm is basically the same as that of classical KNN algorithm. After combining the two ideas, the $KNN^{TS\text{-}PK+}$ algorithm changes slightly. Compared with the classical KNN algorithm and KNN^{TS} algorithm, its classification accuracy has been further improved, but compared with KNN^{PK+} algorithm, its classification accuracy is slightly lower. On the other hand, starting from the comparative results of classification time, the experimental results are consistent with the previous ones.

As can be seen from Fig. 9, compared with the classical KNN algorithm, the classification time of KNN^{PK+} algorithm is reduced, but not to a great extent. Compared with the classical KNN algorithm, the classification time of KNN^{TS}

Table 10 Information of eight data sets.

Data set	The total number of samples	Number of attributes	Number of categories
Hayes Roth	133	6	3
Iris	150	5	3
Seeds	210	8	3
Pima Indians	769	9	2
Page Blocks	5473	11	5
Nursery	12,960	8	3
Census Income	48,842	14	2
Shuttle	58,000	10	7

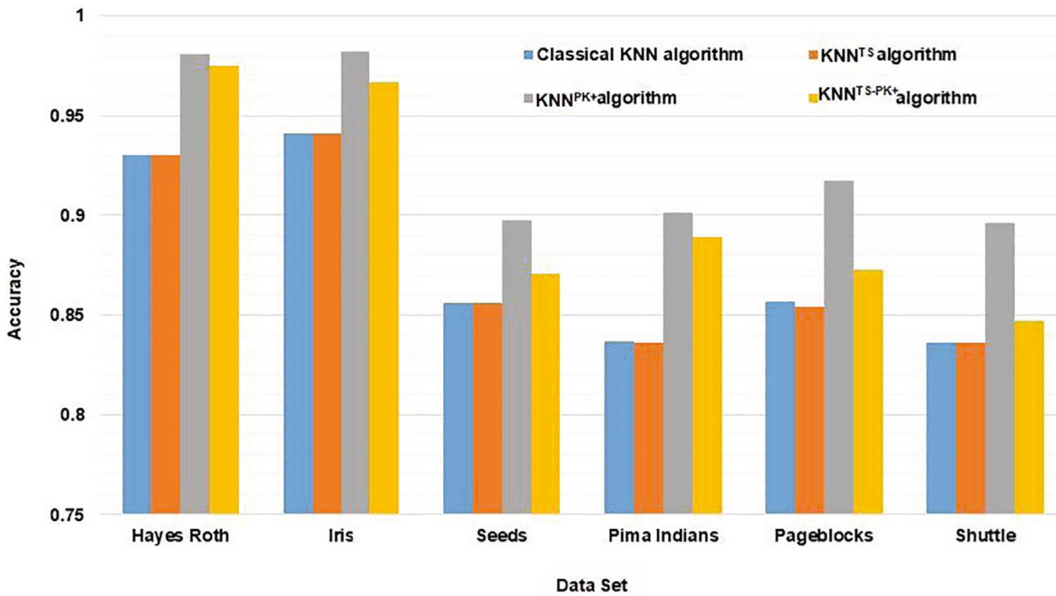

Fig. 8 Compares the classification accuracy of the four algorithms for different data sets.

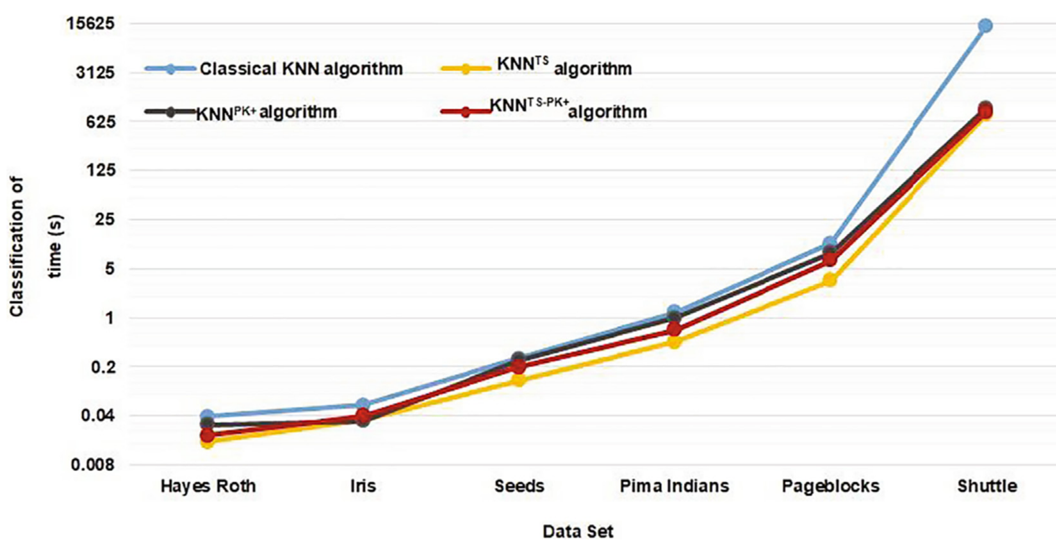

Fig. 9 Time log coordinate comparison diagram of classification of different data sets by four algorithms.

algorithm is significantly reduced. Compared with the classical KNN algorithm and KNN^{PK+} algorithm, the combined $KNN^{TS\text{-}PK+}$ algorithm has significantly shortened the classification time, but compared with KNN^{TS}, the classification time is slightly longer. The experiment was further extended to compare with SVM. The data set used here is the Wisconsin Dataset Breast Cancer Diagnostic (WDBC) from the UCI database, which contains data on 32 features of 10 aspects of the nucleus. These 10 aspects are radius, texture, perimeter, area, smoothness, compactness, concavity, concave points, symmetry, and fractal dimension. Among them, there are three types of data in each of the 10 aspects: sample mean, sample standard deviation, and sample maximum, a total of 569 cases of data.

The data distribution is shown in Table 11. The comparison of classification accuracy and classification time was shown in Fig. 10 and Fig. 11. As can be seen from Fig. 10, the classification accuracy of algorithm $KNN^{TS\text{-}PK+}$ is intermediate between algorithm KNN^{PK+} and algorithm KNN^{TS}, but higher than that of classical KNN algorithm and SVM algorithm. Meanwhile, it can be seen from Fig. 11 that the classification time of algorithm $KNN^{TS\text{-}PK+}$ is still between algorithm KNN^{PK+} and algorithm KNN^{TS}, but it is less than the classification time of classical KNN algorithm and SVM algorithm.

Based on the analysis of the above experimental results, it can be concluded that the classical KNN algorithm does not pre-process the received data, so it cannot accurately process

Table 11 WDBC data distribution.

Aspects	Sample mean	Sample standard deviation	Sample maximum
Radius	6.981–28.11	0.1115–2.873	7.93–36.04
Texture	9.71–39.28	0.3602–4.885	12.02–49.54
Perimeter	43.79–188.5	0.757–21.98	50.41–251.2
Area	143.5–2501	6.802–542.2	185.2–4254
Smoothness	0.05263–0.1634	0.001713–0.03113	0.07117–0.2226
Compactness	0.01938–0.3454	0.002252–0.1354	0.02729–1.058
Concavity	0–0.4268	0–0.396	0–1.252
Concave Points	0–0.2012	0–0.05279	0–0.291
Symmetry	0.106–0.304	0.007882–0.07895	0.1565–0.6638
Fractal Dimension	0.04996–0.09744	0.0008948–0.02984	0.05504–0.2075

some fuzzy defined data, which affects the classification accuracy and classification time. KNN^{PK+} algorithm makes more accurate processing of data by determining the center used to divide the spherical region, and then obtains new data sets more suitable for classification, which can effectively avoid the misclipping of useful data in the data set, so the classification accuracy is improved. KNN^{TS} algorithm reduces the overlap problem between regions when dividing and processing data, and avoids the problem of distance calculation for multiple times due to the uncertainty of appropriate radius, thus significantly shortening the classification time. After $KNN^{TS\text{-}PK+}$ determines the center of spherical region, the number of sample clipping may be less than that of KNN^{TS} algorithm, so the classification time is longer, but the probability of removing useful data is greatly reduced. After determining the radius of the spherical region, the new sample data set may have more fuzzy data than the samples determined by KNN^{PK+} algorithm, so even if the accuracy is not improved greatly, the problem of overlapping spherical regions is greatly

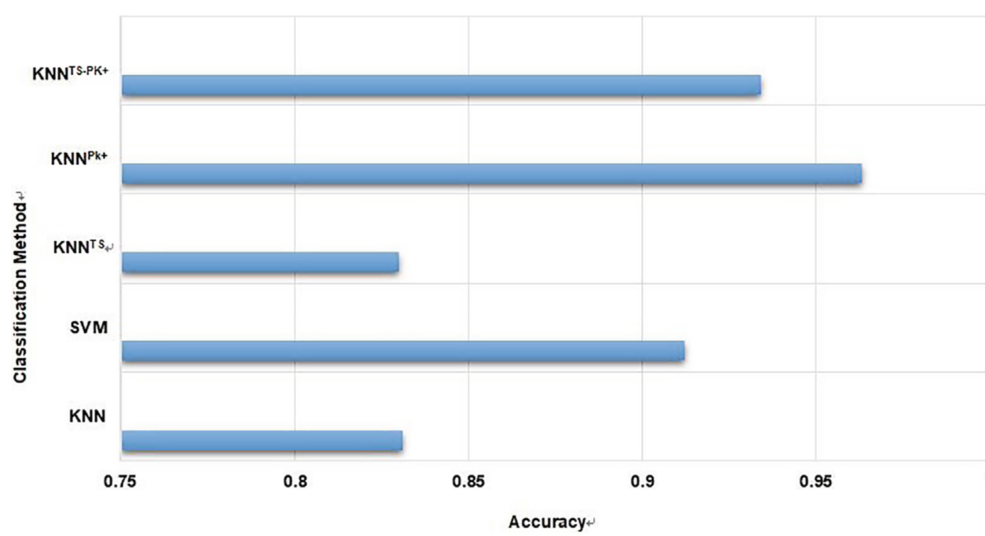

Fig. 10 Comparison of classification accuracy of five algorithms on WDBC dataset.

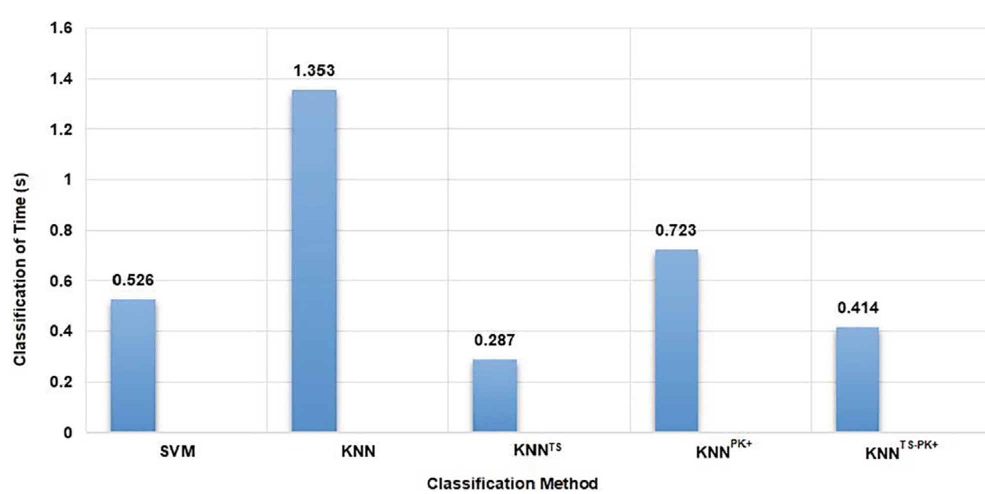

Fig. 11 Comparison of classification time of five algorithms on WDBC dataset.

reduced. The overall classification accuracy and efficiency of $KNN^{TS\text{-}PK+}$ algorithm is higher than that of classical KNN algorithm.

5. Conclusions and future work

To solve the problem that KNN classification algorithm does not pre-process data samples, which leads to a long classification time and a decrease in classification accuracy, two improved algorithms KNN^{TS} and $KNN^{TS\text{-}PK+}$ are proposed on the basis of careful study of the advantages and disadvantages of KNN classification algorithm in this paper. KNN^{PK+} algorithm can greatly improve classification accuracy and effectively reduce classification time. KNN^{TS} algorithm is an improved KNN algorithm for equal-radius spherical region division based on TS, which can divide data set samples by equal-radius spherical region division method, and obtain a reasonable radius R value by tabu search algorithm. Then, with the help of this R value, a new data set more suitable for KNN classification calculation is cropped, and the KNN algorithm is used for further classification in this new data set. KNN^{TS} can greatly reduce the classification time while the classification accuracy is basically unchanged. The second improved method $KNN^{TS\text{-}PK+}$ is an improved KNN algorithm based on clustering and region division. First, it uses PK-Means ++ algorithm to find the appropriate center in the data sample, then obtains the appropriate radius by tabu search algorithm, and then forms an initial classifier combined with the equal-radius spherical region division method to cut out the new data set. Finally, KNN algorithm is used for classification in the new data set. Although the classification accuracy of $KNN^{TS\text{-}PK+}$ algorithm is slightly lower than KNN^{PK+} algorithm, its classification time is effectively higher than KNN^{TS} algorithm. In a word, the improved algorithms proposed in this paper all improve the overall classification effect, and they effectively improve the classification accuracy and efficiency. As a relatively perfect classification method, KNN is required for everything from the recognition of numbers to the recognition of faces. With the continuous development of communication technology [27], especially with the introduction of 5G [28] and blockchain [29], KNN algorithm is bound to serve these developing technologies. The improvement of KNN will also play a supporting role in the increasingly industrialized AI technologies such as big data [30] and data collection [31].In addition, further research on risk assessment [32] and system reliability [33] is needed. In the future, more research emphasis will be placed on the application of optimization algorithms. In particular, KNN can be used in data prediction and analysis, such as disease prediction.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported in part by "the Jilin Province Science and Technology Development Plan" (Grant No. 20190302105GX, JJKH20210786KJ), "the Postgraduate Scientific Research Innovation Plan of Jilin Normal University" (Grant No. 201947), and "the Postgraduate Innovation Project of Jilin Normal University" (Grant No. 202023).

References

[1] T. Cover, P. Hart, Nearest neighbor pattern classification, IEEE Trans. Inf. Theory 13 (1) (1967) 21–27.
[2] Z.H. Wang, S.T. Liu, et al, KNN classification algorithm based on improved k-modes clustering algorithm, Computer Eng. Des. 40 (8) (2019) 2228–2234.
[3] Y.F. Wang, W.J. Hao, et al, Improved KNN algorithm based on clustering and density clipping, J. Qingdao Univ. (Natural Sci. Ed.) 30 (2) (2017) 62–68.
[4] K. Saetern, N. Eiamkanitchat, An ensemble K-nearest Neighbor with neuro-fuzzy method for classification, Adv. Intelligent Syst. Comput. 265 (2014) 43–51.
[5] L. Tian, Research on KNN text classification algorithm, Xi'an University of Technology, Xi'an, China, 2016, M.S. thesis.
[6] P. Vinaybhushan, T. Hirwarkar, Privacy-perserving KNN classification protocol over encrypted relational data in the cloud, Adv. Mathematics Sci. J. 9 (7) (2020) 4589–4596.
[7] A. Pathak, S. Pathak, Study on decision tree and KNN algorithm for intrusion detection system, Int. J. Eng. Res. Technol. 9 (5) (2020) 376–381.
[8] Arthur Ahmad Fauzi; Fitri Utaminingrum et al. "Road surface classification based on LBP and GLCM features using KNN classifier," Bulletin of Electrical Engineering and Informatics, vol. 9, no. 4, pp. 1446-1453, 2020.
[9] Diptee Chikmurge; Shriram R. "Marathi handwritten character recognition using SVM and KNN classifier," A. Abraham et al. (Eds.): Hybrid Intelligent Systems, AISC 1179, pp.319-327, 2021. https://doi.org/10.1007/978-3-030-49336-3_32.
[10] Hao Yifeng. "Design of teaching system of college students based on KNN algorithm," V. Sugumaran et al. (Eds.): Application of Intelligent Systems in Multi-modal Information Analytics, AISC 1234, pp. 708–712, 2021. https://doi.org/10.1007/978-3-030-51556-0_107.
[11] A. Upadhyay, D. Upadhyay, et al, KNN-based classification and comparative study of multispectral LISS-III Satellite ImageD. Gupta et al, in: (eds.): International Conference on Innovative Computing and Communications, Advances in Intelligent Systems and Computing 1165, 2021, pp. 1123–1130, https://doi.org/10.1007/978-981-15-5113-0_94.
[12] Vaishali S. Vairale ; Samiksha Shukla. "Recommendation of food items for thyroid patients using content-based KNN method," D. S. Jat et al. (eds.): Data Science and Security, Lecture Notes in Networks and Systems 132, pp.71-77, 2021. https://doi.org/10.1007/978-981-15-5309-7_8.
[13] Hilal Arslan. "A new COVID-19 detection method from human genome sequences using CpG island features and KNN classifier," Engineering Science and Technology, an International Journal, Available online 9 January 2021. doi: https://doi.org/10.1016/j.jestch.2020.12.026.
[14] X.Y. Huang, An improved KNN algorithm and its application in real-time car-sharing prediction, Dalian University of Technology, Daian, China, 2018, M.S. thesis.
[15] X.D. Chen, Analysis and research of common clustering algorithm in data mining, Digital Technol. Appl. 2017 (4) (2017) 151–152.
[16] L. Jiang, S.L. Xue, K-means algorithm for optimizing initial clustering center and determining K value, Comput. Digital Eng. 46 (1) (2008) 21–24.
[17] H.Y. Wang, W.C. Cui, et al, An optimized k-means ++ algorithm guided by local probability, J. Jilin Univ. (Sci. Ed.) 57 (6) (2019) 1431–1436.

[18] Y. Hu, Research on KNN text fast classification algorithm based on regional division, Shandong University, Jinan, China, 2012, M.S. thesis.

[19] Y. Wang, Y.H. Tai, Research on the location of beer distributors based on tabu search algorithm, Logistics technology 42 (11) (2019) 13–16.

[20] Fred Glover, Artificial intelligence, heuristic frameworks and tabu search, Manag. Decis. Econ. 11 (5) (1990) 365–375.

[21] X.B. Wang, C.Y. Ren, et al, A heuristic algorithm for a class of minimum-maximum vehicle routing problems, Operat. Res. Manage. 22 (6) (2013) 26–33.

[22] F. Garcia, F. Guijarro, et al, Index tracking optimization with cardinality constraints: a performance comparison of genetic algorithms and tabu search heuristics, Neural Computing and Applications 30 (8) (2018) 1–17.

[23] Q.M. Jin, F.F. Li, Application of BFD mixed tabu search in one-dimensional packing problem, Qinghai traffic Sci. Technol. 32 (1) (2020) 34–38.

[24] X. Li, Q.Y. Zhu, Adaboost algorithm improved BP neural network prediction research, Computer Eng. Sci. 35 (8) (2013) 96–102.

[25] H.Y. Wang, P.D. Xu, J.H. Zhao, Improved KNN Algorithm Based on Preprocessing of Center in Smart Cities, Complexity 2021 (5524388) (2021) 1–10.

[26] J.W. Hu, Improved KNN classification algorithm based on regional division, Qingdao University, Qingdao, China, 2016, M.S. thesis.

[27] Zhenyu Zhou, Chuntian Zhang, Jingwen Wang, Bo Gu, Shahid Mumtaz, Jonathan Rodriguez, Xiongwen Zhao, Energy-Efficient Resource Allocation for Energy Harvesting-Based Cognitive Machine-to-Machine Communications, IEEE Trans. Cognit. Commun. Networking 5 (3) (2019) 595–607.

[28] Muhammad Shahmeer Omar, Syed Ali Hassan, Haris Pervaiz, Qiang Ni, Leila Musavian, Shahid Mumtaz, Octavia A. Dobre, Multiobjective Optimization in 5G Hybrid Networks, IEEE Internet Things J. 5 (3) (2018) 1588–1597.

[29] Zhenyu Zhou, Xinyi Chen, Yan Zhang, Shahid Mumtaz, Blockchain-Empowered Secure Spectrum Sharing for 5G Heterogeneous Networks, IEEE Network 34 (1) (2020) 24–31.

[30] Z. Xie, J. Wang, L. Miao, Big data and emerging market firms' innovation in an open economy: The diversification strategy perspective, Technol. Forecast. Soc. Chang. 173 (2021) 1–14.

[31] Shidrokh Goudarzi, Nazri Kama, Mohammad Hossein Anisi, Sherali Zeadally, Shahid Mumtaz, Data collection using unmanned aerial vehicles for Internet of Things platforms, Computers & Electrical Engineering 75 (75) (2019) 1–15.

[32] M. Zhang, F. Conti, H. Le Sourne, D. Vassalos, P. Kujala, D. Lindroth, S. Hirdaris, A method for the direct assessment of ship collision damage and flooding risk in real conditions, Ocean Eng. 237 (2021) 109605.

[33] Xingwang Li, Mengle Zhao, Ming Zeng, Shahid Mumtaz, Varun G. Menon, Zhiguo Ding, Octavia A. Dobre, Hardware Impaired Ambient Backscatter NOMA Systems: Reliability and Security, IEEE Trans. Commun. 69 (4) (2021) 2723–2736.