Predicting the Number of Nearest Neighbor for kNN Classifier

Yanying Li, Youlong Yang, Jinxing Che and Long Zhang

***Abstract*—The k nearest neighbor (kNN) rule is known as its simplicity, effectiveness, intuitiveness and competitive classification performance. Selecting the parameter k with the highest classification accuracy is crucial for kNN. There's no doubt that the leave-one-out cross validation (LOO-CV) is the best method to do this work as its almost unbiased property. However, it is too time consuming to be used in practice especially for large data. In this paper, we propose a new algorithm for selecting an optimal neighborhood size k. We found that the classification accuracy of LOO-CV is approximate concave for the parameter k. And a search method is proposed to pick out the optimal value of k. An empirical study conducted on 8 standard databases from the UCI repository shows that the new strategy can find the optimal k with significantly less time than the LOO-CV method.**

***Index Terms*—k nearest neighbor, leave-one-out cross validation, selecting the parameter k.**

I. Introduction

THE k nearest neighbor (kNN) rule [1] is a typical non-parametric algorithm for pattern classification problems [2], [3], [4]. Fix and Hodges [5] firstly introduced the nearest neighbor rule in 1951 and then Cover and Hart [6] generalized some important characters such as the upper bound of asymptotic error rate of kNN in 1967. After then kNN gained wide attention and rapid development. The condensed nearest neighbor (CNN) rule [7] proposed by Hart in 1968 is the first algorithm to reduce the model size of kNN. CNN dose not stop the iteration until all instances that are classified incorrectly are added to the select set. In order to improve the effectiveness of CNN, a number of variations of CNN had been developed, such as the reduced nearest neighbor (RNN) algorithm, the selective nearest neighbor (SNN) algorithm, the modified condensed nearest neighbor (MCNN) algorithm, the pairwise opposite class-nearest neighbor (POC-NN) algorithm and fast condensed nearest neighbor (FCNN) algorithm et al [8], [9], [10], [11], [12]. Another class of improved kNN algorithms is based on the fuzzy sets theory. As an extension of fuzzy sets theory, the fuzzy nearest neighbor classifiers use fuzziness to find the neighbors set of a query point [13], [14], [15], [16], [17]. A detailed survey about the fuzzy nearest neighbor classification can be found in ref. [3]. Recently, some latest kNN-based algorithms are developed, such as the probabilistic group nearest neighbor (PGNN) query algorithm [18], the decrease the size of the training set for kNN regression (DISKR)[19]and the local mean-based pseudo nearest neighbor (LMPNN) rule [20].

Although the kNN has its advantages, such as simplicity, effectiveness, intuitiveness and competitive classification performance, it also has its drawbacks. To improve the performance of kNN-based rule, the scholars' research mainly focuses on the selection of the appropriate similarity measures, feature subsets and their weights in the case of the retrieval step [21], [22], [23], [24], [25], [26], [27], [28]. However, how many cases should be combined to generate the label of a query point? Many researchers believe that finding the optimal parameter k is crucial to improve the performance of kNN rule [26], [28], [29], [30], [31]. When parameter k is too small, the corresponding kNN classifier may has the following disadvantages: sensitive to noise points, data sparseness and data imbalance [20], susceptible to overfitting [31] and inefficient for representing the general pattern of the data [28]. When parameter k is too large, the corresponding kNN classifier may has the following disadvantage: inclusive many irrelevant or distorted points from other classes [28]. All of these disadvantages lead to a low classification accuracy. Hence, the selection of parameter k is a crucial issue. Unfortunately, there have been few works that try to optimize the value of k.

Zhang and Song summarize the existing methods and divide them into four different types [31]. The first one is the constant value method which is the most straightforward method, the closest neighbor is used to assign the class for the query point. For example, the WEKAs default setting is k = 1. This setting suffers from noise, and cannot reflect the diversity of different data sets. The second and the most widely used method is cross-validation that can obtain different k values for different data sets but consumes much time. The different k values can be obtained according to the number of instances in training set or the distribution of class [32]. The third is the heuristic method. Okamoto and Satoh point out that the optimum k value increases gradually as the number of training instances increases [32], they did not provide a uniform relationship model. And Wettschereck confirms that a single k value suffices to classify all queries [33]. The fourth method is the evolutionary parameter optimization algorithms [26], [28]. These methods detect k values by genetic algorithm , which is much time consuming as well. Zhang and Song propose a novel method of using back-propagation neural networks to explore the relationship

Manuscript received April 29, 2019; revised September 6, 2019. The research was supported by the National Natural Science Foundation of China (Grant No. 61573266,71971105,11801013) and the Foundation of the Doctoral Scientific Research of Baoji University of Arts and Sciences (Grant No.ZK2017020) and the Foundation of Baoji City Guidance plan for scientific and technological innovation(Grant No.2017JH2-23,2018JH-19). Scientific Research Program Funded by Shaanxi Provincial Education Department (Grant No. 18JK0045). The research was supported by Natural Science Major Research Plan in Shaanxi Province of China(Grant No.2018ZKC-115, 2018ZKC-023), and supported by Shaanxi social science major theoretical and practical issues research project in 2019(Grant No.2019C003).

Yanying Li is with the School of Mathematics and Information Science, Baoji University of Arts and Sciences, Baoji 721013, Shaanxi, China.

YouLong Yang is School of Mathematics and Statistics, Xidian University, 266 Xinglong Section of Xifeng Road, Xi'an, Shaanxi, 710126, China

JinXing Che is with the College of Science, NanChang Institute of Technology, NanChang 330099, JiangXi, China.

Long Zhang is with the Shaanxi Baoguang Vacuum Electric Device Co., Ltd., Baoji 721016, Shaanxi, China.

between data set characteristics and the optimal values of k, then the relationship and the data set characteristics of a new data set are used to recommend the value of k for this data set.

However, the most important thing for a classifier is a good classification accuracy, and all methods mentioned above did not consider the direct relationship between classification accuracy and the parameter k. It is well known that the leave one out cross validation (LOO), a special case of cross validation (CV), is proximately unbiased. That is that LOO can evaluate the model with the best accuracy. However, LOO is a costing-time process, conventional methods use 5-fold cross validation or 10-fold cross validation instead of LOO for reducing the complex, but this sacrifices the main objective of the classification accuracy.

Fortunately, we have found that there is a close relationship between the LOO accuracy(LOO-A) and the parameter k through the experiment. That is the LOO accuracy is approximation concave on the parameter k. This paper proposed a search strategy to select the optimal k according to the LOO-A according to the approximation concave relationship. This search strategy not only greatly reduces the time complexity, but also ensures that the k value corresponding to the highest accuracy is guaranteed when the step size is small enough. The experimental results also show that on the 8 datasets, the new method stays LOO in time and has the same classification accuracy as LOO.

The rest of this paper is structured as follows: Section 2 introduces some related basic knowledge. A detailed description of the new algorithm is listed in Section 3. The experimental results is given in Section 4. Section 5 concludes the paper with a brief summary and propose the further works.

II. The related basic knowledge

The kNN rule is one of top ten data mining algorithms. kNN Algorithm does its work by finding the k nearest neighbors of the query point in the training data. Generally speaking, the most frequently label in the k nearest neighbors is the result of classification. The pre-defined category labels are essential requirement for kNN rule. Three main core elements of kNN are the training data, similarity measure or distance between objects and the number of the nearest neighbors.

In a general classification problem, let $T = \{x_m \in R^n\}_{m=1}^M$ be a training set with M training samples in n-dimensional feature space, and there are l class labels, each sample x_n corresponds to its class label c_n, where $c_n \in \{w_1, w_2, \cdots, w_l\}$. For a query point x, the kNN rule is implemented as follows:

(1) Compute the distances between the unknown query point y and samples in training data. Then find k nearest neighbors set $Neighbor\{x_1, x_2, \cdots, w_k\}$ of y according to these distances. The distance between y and the neighbor x_i is measured by the Euclidean distance metric by Eq.(1)

$$d(y, x_i) = \sqrt{(y - x_i)^T (y - x_i)} \tag{1}$$

(2)To assign a label to the query point, the following two strategies are widely used.

$$c = \arg\max_i \sum_{x_j \in Neighbor} sign(x_j, c_i) \tag{2}$$

$$c = \arg\max_i \sum_{x_j \in Neighbor} Sim(y, x_j) sign(x_j, c_i) \tag{3}$$

where x_j is one of the neighbors in the training set, $sign(x_j, c_i) \in \{0, 1\}$ is a indicator function that implies x_j whether belongs to class c_i. $Sim(y, x_j)$ is a function that is used to measure the similarity of y and x_j. Eq.(2) means that the most frequently class label in the nearest neighbors set will be assigned to the predication. Whereas Eq.(3) means the class with maximal sum of similarity will be assigned to the predication. The former is used in this paper as it is the most classical one.

III. The explicit process of the new algorithm

A. motivation

kNN is famous for its simplification, high efficiency and classification accuracy. The unique parameter k is closely related to the classification results. How to select a optimal parameter k which can make kNN get a high classification accuracy for all datasets is our ultimate goal. It is well known that the Leave-One-Out Cross Validation (LOO-CV) is a very good way to evaluate the performance of a classifier, its best advantage lies in the classification error (LOO-CV-E) is almost unbiased, and Luntz and Brailovsky give a detailed proof [34]. If the time complexity is not taken into account, LOO-CV is the best choice to evaluate the performance of the nearest neighbor classifier. In this case, we need to perform a traversal test for k within its range D, that is, for each k$\in D$, we must calculate the corresponding nearest neighbor classifier classification accuracy, and then select the highest precision corresponding to the k, as the final parameter selection. Obviously this ensures that the classifier has the highest classification accuracy, but it takes a lot of time. If the time required for this process can be effectively reduced, then a nearest neighbor classifier with the highest classification accuracy can be obtained. We study the relationship between the parameter k and the LOO-CV-E in the nearest neighbor method and find that the LOO-CV-E is approximation convex on k, so that the LOO-CV accuracy (LOO-CV-A) is approximation concave on k. 8 representative data in the UCI database are selected to show the approximation convex relationship. They are Australian Data, Glass Data, Ionosphere Data, Iris Data, Sonar Data, Wine Data, Breast Cancer Wisconsin Data and Diabetic Retinopathy Debrecen Data, abbreviated as Breast Data and Diabetic Data separately. We draw the images of LOO-CV accuracy of k nearest neighbor classifier with respect to k on these datasets, see Figure 1 and Figure 2. These figures show that the two are roughly concave relationship, so choose a good optimization search method can avoid traversal search of k in its range. Inspired by reference [35], we propose a similar search strategy that can fast converge to the k values with the highest classification accuracy.

B. Data preprocessing and the range of k

If the value ranges of some attribute variables are very large, and others are very small, and the differences between these ranges are significant disparity, then the attribute variables with big ranges might dominant in leaning and

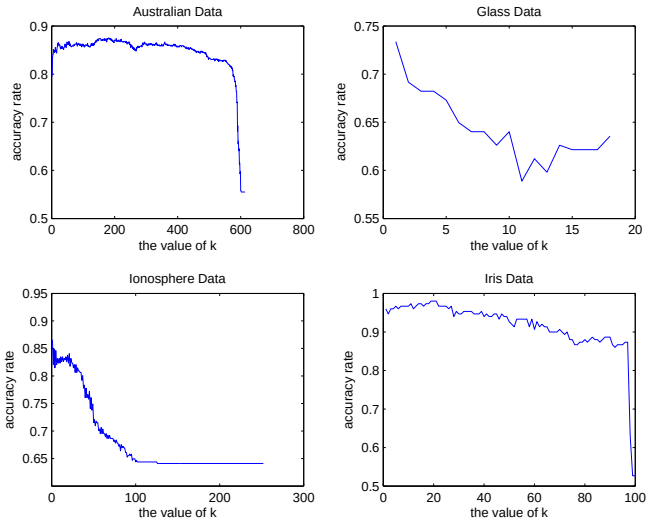

Fig. 1. Classification accuracy of LOO-CV on Australian, Glass, Ionosphere, Iris

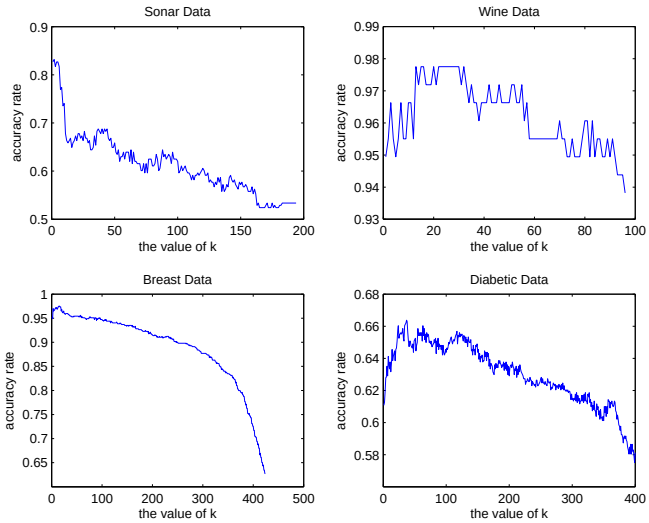

Fig. 2. Classification accuracy of LOO-CV on Sonar,Wine,Breast,Diabetic

TABLE I
VALUES OF ATTRIBUTE VARIABLE FOR AUSTRALIAN

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
minimum	0	13.75	0	1	1	1	0	0	0	0	0	1	0	1
maximum	1	80.25	28	3	14	9	28.5	1	1	67	1	3	2000	100001
difference	1	66.5	28	2	13	8	28.5	1	1	67	1	2	2000	100000

result an unsatisfactory classification result. For example, the value ranges of the 14 attribute variables in Australian Data are listed in table I, the first two rows are the minimum and maximum of variables, the last row is the difference of maximum and minimum. To avoid this situation, we use regularization method to process data.

From the table I, the difference of the 14th variable is 10000, and the differences of the first, 4th, 8th, 9th, 11th and 12th variables are 1 or 2, this situation makes the 14th variable play a dominant role in the process of determining the nearest neighbors, and the other first, 4th, 8th, 9th, 11th and 12th variables have little effect in the same process. But it's possible that these variables are closely related to the classification. According to the above analysis, some disparate data should be pre-processed in order to get a better classification effect. In this paper, a linear normalization method is adopted. We map the data linearly within a specific range. Let $y\prime_{max}$ and $y\prime_{min}$ are the target maximum and minimum values of the variables to be normalized, respectively. Thus each variable y to be normalized can be transformed by the following procedure:

$$y\prime = P(y) = A \times y + B \tag{4}$$

$y\prime$ is the normalized value of y. The coefficients A and B are jointly controlled by y_{max}, $y\prime_{max}$, y_{min} and $y\prime_{min}$. Calculated as follows:

$$\begin{cases} A = \{y\prime_{max} - y\prime_{min}\}/\{y_{max} - y_{min}\} \\ B = y\prime_{max} - \{\{y\prime_{max} - y\prime_{min}\}/\{y_{max} - y_{min}\}\} \times y_{max} \end{cases} \tag{5}$$

Experiments show that the classification accuracy of LOO-CV on Australian Data is 71.01%, the run time is 5454.5606 seconds; and the classification accuracy of LOO-CV on normalized Australian Data is 87.54%, the run time is 898.9572 seconds.

Since the class imbalance has a great influence on the k nearest neighbor algorithm, in order to prevent the influence of big class on classification, we control the range of k as following:

$$k_{max} = card(minclass) * 2 \tag{6}$$

$card(minclass)$ is the sample size of the smallest class, that is the maximum of k is no larger than 2 times $card(minclass)$. The range of k is [1, k_{max}].

C. Approximate concave optimization framework for selecting the optimal k

Let $T = \{(x_1, c_1), \cdots, (x_n, c_n)\}$ is the raw data set, x_i is the value of attribute variable, and c_i is the corresponding class label. When the query X_i is input, the corresponding $\widetilde{c}_i$ is the classification results of the algorithm [35], [36].

The optimal k value means that the nearest k neighbors contain the most information of the query, which can be abstracted as the following optimization problem.

$$\max_{k \in N^+} F(k) = \lambda * \frac{k}{n} + (1 - \lambda) * (1 - MAE(D, k)) \tag{7}$$

where n is the size of the training set, k is the nearest neighbor number,$MAE(D, k)$ is the average absolute error of the k nearest neighbor classifier for data set D, λ is a trade-off between k value and classification accuracy. In Eq.(7), the first term is the item that controls the size of the k value, a too large k will result in a significant increase in the complexity of the algorithm; the second term is the classification accuracy under k. The model should first consider the model accuracy, so λ generally takes a smaller value (close to 0 and no more than 0.5). If set $\lambda = 0$, the model will have the highest classification accuracy, which is more suitable for the small sample data. In general, the useful information increases with the increase of k value, but at the same time the redundant information will increase, that is, as k from small to large, the initial useful information dominated, and then reversed. If Eq.(7) is considered as a continuous function of the variable k, it is clear that there will be the following properties [35]:

$$\frac{\partial^2 MAE(k,D)}{\partial k^2} \geq 0 \tag{8}$$

Use this property for $F(k)$, then:

$$\frac{d^2 F(k)}{dk^2} \leq 0 \tag{9}$$

According to the above analysis we can see that $F(k)$ is approximately concave on k, so the following search strategy is proposed to solve the optimal k. It is worth noting that, because k only takes integer values, the problem is equivalent to solving integer convex optimization problem, which also makes the problem more complex. First, we select three initial k values, denoted as k_1, k_2, k_3, and evaluate the corresponding MAE of the corresponding k nearest neighbor model by LOO-CV. Then we calculate the corresponding objective function value $F(k)$. Shorthand this search strategy for LOO-O-K(the optimal k based on LOO-CV). The specific process consists of two subroutines and the main program. The subroutine 1 listed in table II is the kNN-routine. The subroutine 2 listed in table III calculates the accuracy and the MAE of LOO-kNN. Table IV shows the proposed concave optimization search strategy in detail.

TABLE II
SUBROUTINE 1 KNN-ROUTINE

subroutine 1 kNN
Input: training set D_{train} ; query variable x; k
1 Find the top k nearest neighbors of x on D_{train} by Eq.(1)
2 Assign a label to x according to Eq.(2)
3 If the label assigned to x is right,then result=1;otherwise result=0
Output: result

TABLE III
SUBROUTINE 2 LOO-KNN-ACCURACY

subroutine 2 LOO-kNN-Accuracy
Input: data set D; the number of samples n; k
1 correct=0
2 for $i = 1 : n$
3 select the ith sample as the inquiry x, the other samples are the training set D_{train}
4 correct= correct+kNN(D_{train}, x,k)
5 end
6 Accuracy = correct$\div n \times 100\%$
Output: MAE=1- Accuracy

TABLE IV
THE PROPOSED CONCAVE OPTIMIZATION SEARCH STRATEGY FOR OPTIMAL K

routine LOO-O-K
input: data set D; the number of samples n; the upper bound M of k; threshold α
1 select three initial values k_1, k_2, k_3
2 put k_1, k_2, k_3 in to LOO-kNN-Accuracy respectively, calculate $MAE_{k_1}, MAE_{k_2}, MAE_{k_3}$
3 Calculate $F(k_1), F(k_2), F(k_3)$ according to Eq.(7)
4 step=0.5
5 while $max\{F(k_1), F(k_2), F(k_3)\} - min\{F(k_1), F(k_2), F(k_3)\} >= \alpha$
6 if $F(k_3) >= F(k_2)$ and $F(k_2) >= F(k_1)$
7 $Optim - K = k_3$
8 $Optim - F = F(k_3)$
9 $step1 = k_2 - k_1, step2 = k_3 - k_2,$
10 $k_1 = k_1 + round(step * step1), k_2 = k_3,$ $k_3 = k_3 + round(step * step2)$
11 if $k_3 > M$
12 $k_3 = M$
14 end
15 else if $F(k_2) >= F(k_3)$ and $F(k_2) >= F(k_1)$
16 $Optim - K = k_2$
17 $Optim - F = F(k_2)$
18 $step1 = k_2 - k_1, step2 = k_3 - k_2,$
19 $k_1 = k_1 + round(step * step1),$ $k_3 = k_3 - round(step * step2)$
20 else
21 $Optim - K = k_1$
22 $Optim - F = F(k_1)$
23 $step1 = k_2 - k_1, step2 = k_3 - k_2,$
24 $k_3 = k_3 - round(step * step2), k_2 = k_1,$ $k_1 = k_1 - round(step * step1)$
25 if $k_1 < 1$
26 $k_1 = 1$
27 end
28 end
29 repet line 2 and 3 to calculate the current $F(k_1)$,$F(k_2)$,$F(k_3)$
30 end
Output: the optimal value $Optim-K$ of k and the optimal result $Optim-F$

Our iterative strategy is to go close to the optimal k value with respect to the $F(k)$ valuation function each time. The step size is set in line 9, line 10, line 18, line 19, line 23 and line 24 in main program LOO-O-K. The advantage of the step-size setting is that it across executes rough and fine search in the process of model selection. At the beginning of the search, the three initial values are chosen to be scattered, and the search step size is large. It is a rough search process, which is helpful to quickly close to the optimal value and reduce the time and computational complexity. With the three k values constantly gather, step change is gradually narrowed into an accurate search, the three k value converge to a k value at last.

Figure 1 and Figure 2 show trend of LOO-CV-A with respect to k. The image is an approximately concave relationship with a zigzag shape, and we suspect that the sawtooth may be caused by a small sample size, which may disappear at a sufficiently large sample size. If the value of k is separated by a small segment in its value space, for example, interval 4, we got Figure 3 and Figure 4. From these figures, the graphs are approximate smooth concave functions, so in the initial rough search process, k must tend to the optimal value, and will not fall into the local optimum. It is worth noting that the three initial k values should be distributed in the lower bound, the middle value, the maximum of the range of k, respectively. If the values of k are too close, the convergence rate will be reduced.

- We found the approximate concave relationship between LOO-CV-A and k.
- We given a fast search strategy which across executes rough and fine search.

IV. EXPERIMENTS

In order to show the advantages of our method, we selected 8 representative data in the UCI Machine Learning Repository Data Sets(http://archive.ics.uci.edu/ml/index.php), namely: Australian Data, Glass Data, Ionosphere Data, Iris Data, Sonar Data, Wine Data, Breast Data, Diabetic Data. The information for the 8 datasets is shown in the Table V. Where the column 2 is the number of samples per data, the column 3 is the number of class values, referred to as categories, the column 4 is sample size of the smallest category, referred to

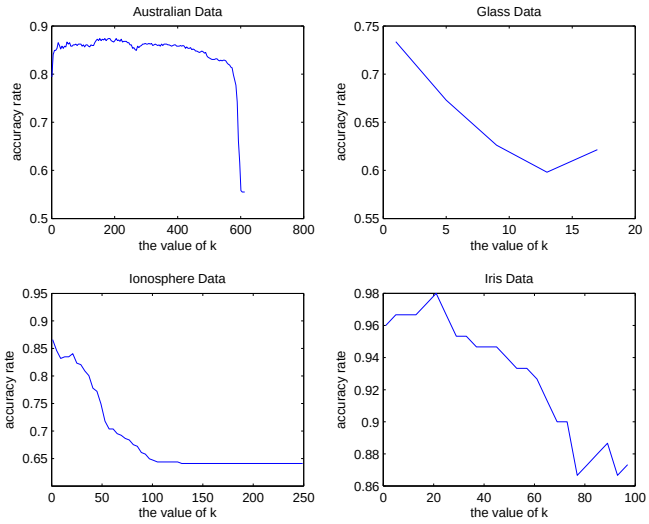

Fig. 3. LOO-CV accuracy for 4 intervals of k on Australian, Glass, Ionosphere, Iris

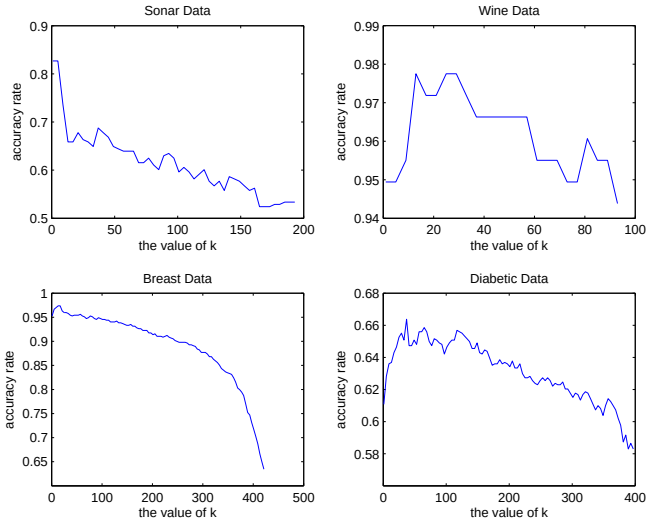

Fig. 4. LOO-CV accuracy for 4 intervals of k on Sonar,Wine,Breast,Diabetic

as the s-c size, the last column is sample size of the largest category, referred to as the l-c size. Glass data is unbalanced.

TABLE V
INFORMATION FOR THE 8 DATASETS

	sample size	categories	s-c size	l-c size
Australian	690	2	307	383
Glass	214	6	9	76
Ionosphere	351	2	126	225
Iris	150	3	50	50
Sonar	208	2	97	111
Wine	178	3	48	71
Breast	569	2	212	357
Diabetic	1151	2	540	611

In the experiment, the range of k is set to be [1, 2× (s-c size)] for each data, that is, the upper bound of k is $M = 2\times$ (s-c size). The s-c size of Diabetic is set to be 200 as the size is too big. The termination condition of the iteration is that the differences of $F(k)$ among the three values of the k are small. We set the threshold to $\alpha = 1/10 \times n$, where n is the sample size. In general, we recommend that the three initial values be selected as: $0.2 \times M$, $0.5 \times M$, $0.8 \times M$, or as close as possible, respectively. Since no particular data is selected, the parameter λ in Eq.(7) is set $\lambda = 0$ in this experiment. If there is large data, λ should be set appropriately to control k to be too large. The data Australian, wine, Breast and Diabetic were normalized and the attribute variables were linearly mapped to the unit interval.

A. Time and accuracy comparison

Experiments show that the new method can quickly converge, and achieve the same accuracy (mean absolute error) with LOO-CV. Table VI gives the time consumption, where t_{LOO-CV} (column 2) gives the time consumptions of LOO-CV, and t_{K1}, t_{K2} and t_{K3} (columns 3, 4, and 5) are the consumptions of our method in three different initial values, $t_{average}$ (column 6) is the average time of LOO-O-K. t_{ratio}(column 7) is the ratio of t_{LOO-CV} to $t_{average}$. The table displays that the new method can save a lot of time, especially in the relatively large data set. The timings in Table VI reported in seconds.

TABLE VI
TIME

	t_{LOO-CV}	t_{K1}	t_{K2}	t_{K3}	$t_{average}$	t_{ratio}
Iris	7.24	1.30	1.71	1.94	1.65	4.39
Sonar	29.36	4.43	3.58	3.55	3.85	7.63
Ionosphere	105.55	10.42	10.45	10.51	10.46	10.09
Glass	2.55	2.13	2.13	2.14	2.13	1.20
Australian	906.31	50.38	46.02	25.13	40.51	22.37
Wine	10.10	1.52	2.13	1.22	1.62	6.23
Breast	523.4303	34.42	36.21	35.64	35.42	14.78
Diabetic	1798.22	165.76	165.54	173.06	168.12	10.70

Table VII lists the optimal k and optimal accuracy of LOO-CV and LOO-O-K respectively. Where column 2 and 3 is the optimal k of LOO-CV and LOO-O-K respectively, and column 4 and 5 is the optimal accuracy of LOO-CV and LOO-O-K respectively. LOO-O-K gets the optimal accuracy of each data.

TABLE VII
THE OPTIMAL VALUE OF K AND ACCURACY

	k	k'	ac.	ac.'
Australian	182	182	0.8754	0.8754
Glass	1	1	0.7336	0.7336
Ionosphere	1	1	0.8661	0.8661
Iris	19/20/21	29	0.9800	0.9800
Sonar	2	2	0.8317	0.8317
Wine	13/15/1620 22-30/32	20	0.9775	0.9775
Breast	10/12/17/22/42	15	0.9754	0.9736
Diabetic	37	37	0.6638	0.6638

B. The stability of LOO-O-K

According to the experiments, LOO-O-K has quick convergence, this part discuss the stability of the quick convergence. We select three initial values for each dada set, see Table VIII. Where each bracket is a set of initial values in the columns 2, 3 and 4, the three values in the column 5 are the iteration counts of convergence according to the initial values separately. We draw the iteration process on different data, see the following graphs: Figure 5 - Figure

12. The three subgraphs in each graph correspond to three sets of initial values, respectively. The horizontal axis is the number of iterations in each subgraph, the vertical axis is k_1, k_2, k_3 and the optimal value k of each generation. These graphs shows that we can reach the optimal value under all situations. The 24 sub-graphs strongly illustrate the stability of the LOO-O-K.

TABLE VIII
THREE INITIAL VALUES AND THE CORRESPONDING ITERATIVE NUMBER

	NO.1	NO.2	NO.3	iterative number
Australian	(123,307,491)	(61,246,368)	(31,184,491)	(12,9,6)
Glass	(4,9,14)	(2,7,11)	(1,5,14)	(5,5,5)
Ionosphere	(50,126,202)	(25,101,151)	(13,76,202)	(9,9,9)
Iris	(20,50,80)	(10,40,60)	(5,30,60)	(6,8,9)
Sonar	(39,97,155)	(19,78,116)	(10,58,155)	(10,8,8)
Wine	(19,48,77)	(10,38,58)	(2,29,77)	(5,7,4)
Breast	(42,212,339)	(21,254,382)	(64,170,360)	(10,10,11)
Diabetic	(40,240,360)	(80,200,320)	(60,220,380)	(12,12,12)

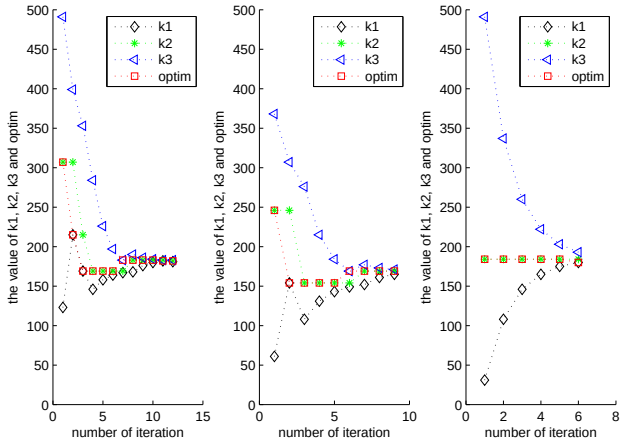

Fig. 5. Sensitive Dependence on Initial Condition for Australian Data

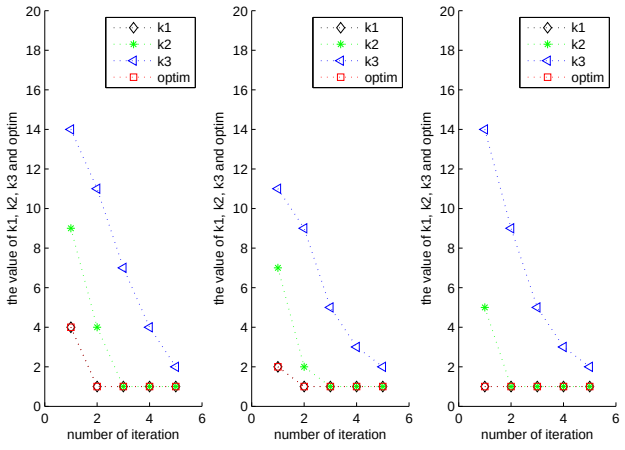

Fig. 6. Sensitive Dependence on Initial Condition for Glass Data

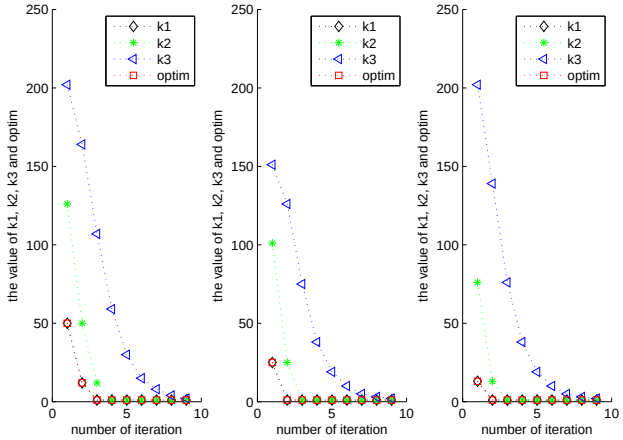

Fig. 7. Sensitive Dependence on Initial Condition for Ionosphere Data

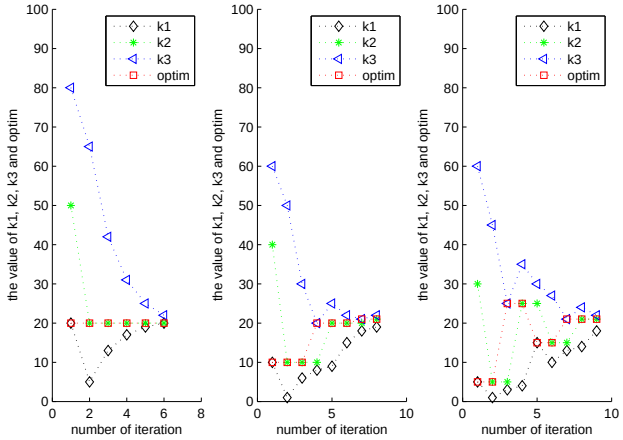

Fig. 8. Sensitive Dependence on Initial Condition for Iris Data

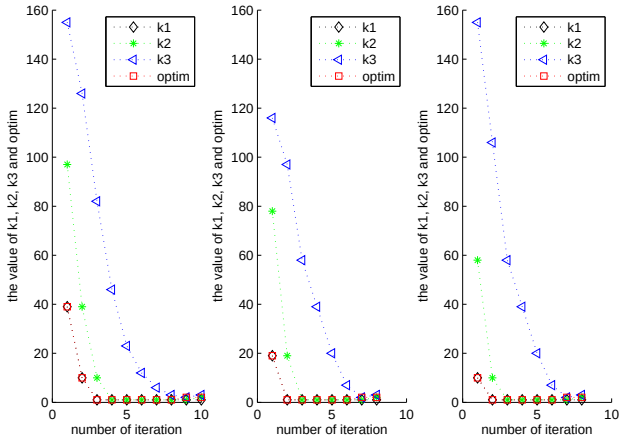

Fig. 9. Sensitive Dependence on Initial Condition for Sonar Data

V. CONCLUSION

As k is the only parameter in the kNN algorithm, which closely affects the effect of kNN. Therefore, this paper focuses on the selection of k values in k nearest neighbor algorithm. We proposed a new strategy to search the optimal parameter k according to the approximate concave relationship between the LOO-CV-A and k. The amount of experiments show the advantages in time, accuracy and stability of LOO-O-K.

ACKNOWLEDGMENT

The authors declare that there is no conflict of interest regarding the publication of this paper. The authors thank the editors and the anonymous reviewers for helpful comments and suggestions.

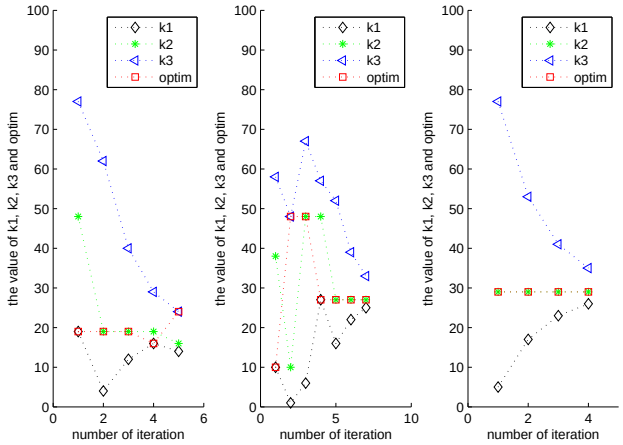

Fig. 10. Sensitive Dependence on Initial Condition for Wine Data

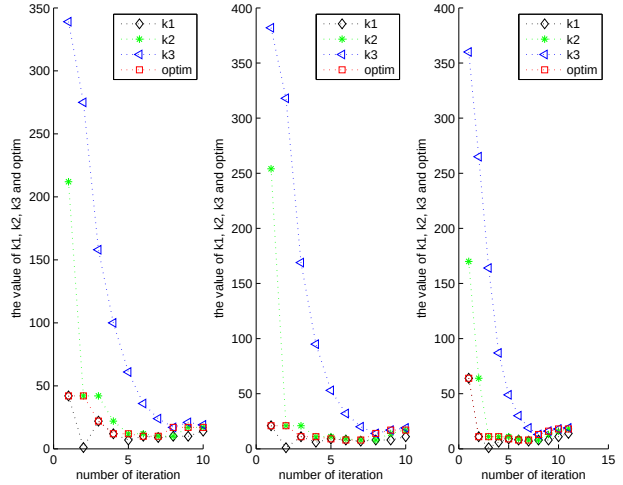

Fig. 11. Sensitive Dependence on Initial Condition for Breast Data

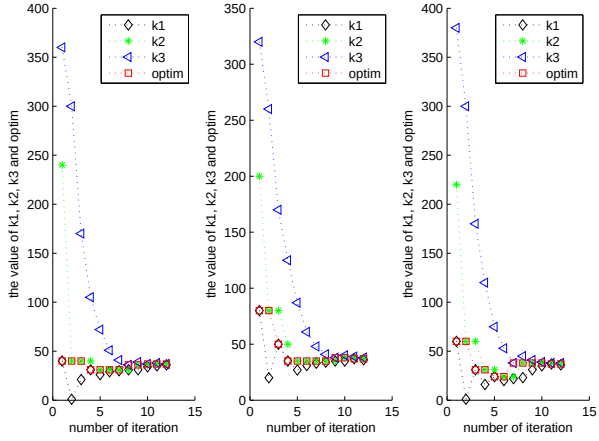

Fig. 12. Sensitive Dependence on Initial Condition for Diabetic Data

REFERENCES

[1] T.M. Cover, P.E. Hart, "Nearest neighbor pattern classification," *IEEE Trans. Inform. Theory*, 13.1(1967):21-27.

[2] T. Hastie, R. Tibshirani, J. Friedman, "The Elements of Statistical Learning: Data Mining, Inference and Prediction," seciond ed., Springer, New York, Berlin, Heidelberg, 2009.

[3] J. Derrac, S. García, F. Herrera, "Fuzzy nearest neighbor algorithms: Taxonomy, experimental analysis and prospects," *Information Sciences* 260 (2014): 98-119.

[4] Zu B, Xia K, Pan Y, et al." A Novel Graph Constructor for Semisupervised Discriminant Analysis: Combined Low-Rank and k-Nearest Neighbor Graph, " *Computational intelligence and neuroscience*, 2017: 9290230.

[5] E. Fix, J. Hodges, Discriminatory analysis, "nonparametric discrimination: Consistency properties," Technical Report 4, USAF School of Aviation Medicine, 1951.

[6] T.M. Cover, P.E. Hart,"Nearest neighbor pattern classification," *IEEE Transactions on Information Theory*, 13 (1967): 21-27.

[7] P. Hart, "The condensed nearest neighbor rule," *IEEE Transactions on Information Theory*, 14 (1968): 515-516.

[8] W. Gates, "The reduced nearest neighbor rule," *IEEE Transactions on Information Theory*, 18 (1972): 431-433.

[9] G. Ritter, H. Woodruff, S. Lowry, T. Isenhour, "An algorithm for a selective nearest neighbor decision rule," *IEEE Trans. Inform. Theory*, 21 (1975): 665-669.

[10] F. Devi, M. Murty, "An incremental prototype set building technique," *Pattern Recognition*, 35 (2002): 505-513.

[11] T. Raicharoen, C. Lursinsap, "A divide-and-conquer approach to pairwise opposite class (poc) nearest neighbor algorithm," *Pattern Recognition Lett*, 26 (2005): 1554-1567.

[12] F. Angiulli, "Fast condensed nearest neighbor rule", In: ICML 05: Proc. 22nd Internat. Conf. on Machine Learning. ACM Press, New York, NY, USA, (2005): 25-32.

[13] T. Denoeux, "A k-nearest neighbor classification rule based on Dempster-Shafer theory," *IEEE Transactions on Systems, Man and Cybernetics*, 25.5(1995):804-813.

[14] L.M. Zouhal, T. Denoeux, "An evidence-theoretic k-NN rule with parameter optimization," *IEEE Transactions on Systems, Man and Cybernetics Part C*, 28 (2) (1998): 263-271.

[15] H. Zhu, O. Basir, "An adaptive fuzzy evidential nearest neighbor formulation for classifying remote sensing images," *IEEE Trans. Geosci. Rem. Sens.*, 43 (8) (2005): 1874-1889.

[16] Z. G. Su, P. H. Wang, "Improved adaptive evidential k-NN rule and its application for monitoring level of coal powder filling in ball mill," *Journal of Process Control*, 19.10(2009):1751-1762.

[17] J. Xu, J. Yang, "A nonnegative sparse representation based fuzzy similar neighbor classifier," *Neurocomputing*, 99 (2013): 76-86.

[18] Li J, Xia X, Liu X, et al, "Probabilistic Group Nearest Neighbor Query Optimization Based on Classification Using ELM," *Neurocomputing*, 277(2017): 21-28.

[19] Song Y, Lu J, Lu J, et al, "An efficient instance selection algorithm for k nearest neighbor regression," *Neurocomputing*, 251(C)(2017):26-34.

[20] J. Gou, Y. Zhan, Y. Rao et al., "Improved pseudo nearest neighbor classification," *Knowledge-Based Systems*, 70 (2014): 361-375.

[21] Y. Wang, N. Ishii, "A method of similarity metrics for structured representations," *Expert Systems with Applications*, 12 (1997): 89-100.

[22] Mateos-García D, García-Gutiérrez J, Riquelme-Santos J C," On the Evolutionary Weighting of Neighbours and Features in the k-Nearest Neighbour Rule," *Neurocomputing*, 326C327(2019): 54-60.

[23] K. Kim, I. Han, "Maintaining case-based reasoning systems using a genetic algorithms approach," *Expert Systems with Applications*, 21 (2001): 139-145.

[24] C. Chiu, P. C. Chang, N. H. Chiu, "A case-based expert support system for due-date assignment in a water fabrication factory," *Journal of Intelligent Manufacturing*, 14 (2003): 287-296.

[25] H. Ahn, K. J. Kim, I. Han, "Hybrid genetic algorithms and case-based reasoning systems for customer classification," *Expert Systems*, 23, 3, (2006a): 127-144.

[26] H. Ahn, K. J. Kim, I. Han, "Global optimization of feature weights and the number of neighbors that combine in a CBR system," *Expert Systems*, 23(5) (2006b): 290-301.

[27] H. Ahn, K. J. Kim, I. Han, "A case-based reasoning system with the two-dimensional reduction technique for customer classification," *Expert Systems with Applications*, 32(4) (2007): 1011-1019.

[28] H. Ahn, K. J. Kim, "Using genetic algorithms to optimize nearest neighbors for data mining," *Annals of Operations Research*, 163 (2008): 5-18.

[29] J. M. Garrell I Guiu, E. Golobardes i Rib, E. Bernad i Mansilla, et al., "Automatic diagnosis with genetic algorithms and case-based reasoning," *Artificial Intelligence in Engineering*, 13 (1999): 367-372.

[30] H. Y. Lee, K. N. Park, "Methods for Determining the optimal number of cases to combine in an effective case based forecasting system," *Korean Journal of Management Research*, 27 (1999): 1239-1252.
[31] X. Zhang, Q. Song, "Predicting the number of nearest neighbors for the k-NN classification algorithm," *Intelligent Data Analysis*, 18 (2014): 449-464.
[32] S. Okamoto and K. Satoh, "An average-case analysis of k-nearest neighbor classifier," *Case-Based Reasoning Research and Development*, 1010 (1995): 253-264.
[33] D. Wettschereck, "A study of distance-based machine learning algorithms," PhD dissertation, Department of Computer Science, Oregon State University, Corvallis (1994).
[34] Luntz A, Brailovsky V. "On estimation of characters obtained in statistical procedure of recognition," *Technicheskaya Kibernetica*,3(1969): 6-12.
[35] Che J X,Support vector regression based on optimal training subset and adaptive particle swarm optimization algorithm," *Applied Soft Computing*, 13,8(2013):3473-3481.
[36] Che J X, Wang J Z, "Short-term load forecasting using a kernel-based support vector regression combination model," *Applied Energy*, 132, 11(2014):602-609.