PERGAMON

Pattern Recognition 35 (2002) 1481–1490

PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY

www.elsevier.com/locate/patcog

Optimal reference subset selection for nearest neighbor classification by tabu search☆

Hongbin Zhang*, Guangyu Sun

Computer Institute of Beijing Polytechnic University, West San Huan North Road 56,6#9, Beijing, 100044, People's Republic of China

Received 22 June 2000; accepted 29 May 2001

Abstract

This paper presents an approach to select the optimal reference subset (ORS) for nearest neighbor classifier. The optimal reference subset, which has minimum sample size and satisfies a certain resubstitution error rate threshold, is obtained through a tabu search (TS) algorithm. When the error rate threshold is set to zero, the algorithm obtains a near minimal consistent subset of a given training set. While the threshold is set to a small appropriate value, the obtained reference subset may have reasonably good generalization capacity. A neighborhood exploration method and an aspiration criterion are proposed to improve the efficiency of TS. Experimental results based on a number of typical data sets are presented and analyzed to illustrate the benefits of the proposed method. The performances of the result consistent and non-consistent reference subsets are evaluated. © 2002 Pattern Recognition Society. Published by Elsevier Science Ltd. All rights reserved.

Keywords: Nearest neighbor classification; Tabu search; Reference set; Prototype selection

1. Introduction

Nearest Neighbor (NN) classification is one of the important non-parametric classification methods and has been studied at length. It is well known that the main drawbacks of NN classifiers in practice have been their computational demands and memories. Numerous studies have been carried out to overcome these limitations. Dasarathy provides an excellent survey on nearest neighbor techniques in his book [1].

In order to reduce the computational demands, one may appropriately organize the given data and use efficient search algorithm. Another approach advocated over the years has been the selection of a representative subset of the original training data, or generating a new prototype reference set from available instances. The objective of reducing the number of reference samples is of course the computational efficiency in the operational phase (when classifying unlabeled samples by using a reference set), or/and making the resulting classification and generalization more reliable. The very early study of this kind was probably the "condensed nearest neighbor rule"(CNN) presented by Hart [2]. His method aims to ensure that the condensed subset is consistent with the original data set, i.e., all of the original samples are correctly classified by the condensed subset under the NN rule. Hart's method indeed ensures consistency, but as admitted by the author, the condensed subset is not minimal, and is sensitive to the randomly picked initial selection and to the order of consideration of the input samples. Under the same idea of picking appropriate samples from the original data set onto the reference subset by adding and deleting samples, there are "reduced nearest neighbor rule" of Gates [3], and "iterative

☆This work was supported by the National Science Foundation of China (69675007) and Beijing Municipal Natural Science Foundation (4972008).

* Corresponding author. Tel.: +86-10-6890-7155.

E-mail address: zhb@public.bta.net.cn (H. Zhang).

PII: S0031-3203(01)00137-6

condensation algorithm" of Swonger [4]. All of these algorithms aim at reducing the size of the condensed subset. However, these and other methods, though obtaining a smaller subset than Hart's algorithm at a higher computational cost, do not realize the minimality of the reference subset. The method proposed by Chang created a reference set by generating new representative prototypes [5]. These prototypes are generated by merging two nearest neighbors of the same class at each step as long as such merging does not increase the error rate. This is actually a bootstrap method in statistics. The editing algorithm MULTIEDIT [6], developed by Devijver and Kittler, aims at editing the training samples to make the resulting classification more reliable, especially the ones located near the boundaries between classes. MULTIEDIT has been proven to be asymptotically Bayes-optimal, i.e., when the number of samples and the number of repetitions of the editing process tend to infinity, the 1-NN classification on the edited reference subset will lead to Bayesian decision. However, but in practice, we usually have finite samples, and the MULTIEDIT performance needs investigation. Considering the above reasons, in this paper, we will use Hart's algorithm as a basis for comparisons.

In 1994, Dasarathy presented a condensing algorithm for selecting an optimal consistent subset based on his concept of the nearest unlike neighbor subset (NUNS) [7]. The algorithm introduced a voting mechanism to select the minimal consistent set (MCS) based on the samples representative significance (in the following, we call the algorithm as an MCS algorithm also, the meaning of MCS can be distinguished from the context). Dasarathy's algorithm is the best known algorithm in terms of consistent subset size and the selected samples' representative nature. However, his conjecture of the minimality of obtained MCS (also cited in Ref. [8]) later is proven not to be true by Kuncheva and Bezdek [9] and Cerveron and Fuertes [10] for the popular IRIS data set. In this paper, we will further illustrate, based on a number of experiments, that the MCS obtained by the algorithm generally has less samples, but it is not minimal. We will also give a counterexample to show that the consistent subset of the MCS algorithm is not always monotonically reducing.

In this paper, we treat the reference subset selection as an optimization problem, that is to minimize the number of the reference samples while constrained to some error rate of classification. We use TS to solve this discrete optimization problem and propose a neighborhood exploration method and an aspiration criterion to improve the performance of the general TS. In Section 2, the algorithm for the optimal reference subset selection is described. Experimental data sets are given in Section 3. This is followed by the experimental results and analyses in Section 4. A conclusion is provided in Section 5.

2. Optimal reference subset selection by tabu search

2.1. Definitions and notations

The optimal reference subset selection can be described as following an optimization problem. Let $X=\{x_1,x_2,\ldots,x_N\}$ be the given training data set for NN classification. Each sample has a known class label from the set $\Omega=\{\omega_1,\omega_2,\ldots,\omega_m\}$. Let $P(X)$ be the solution space, it denotes the power set of X, and $S\in P(X)$ be a selected reference subset. $Card(S)$ denotes the cardinality of S. Let the error rate be $e(S)$ when classifying X using the nearest prototype classifier and S as the reference subset. Let t be the tolerable resubstitution error rate threshold.

The problem can be formalized as follows:

Find a reference subset S^* that satisfies

$$Card(S^*)=\min Card(S),\ \text{s.t. } S\in P(X), e(S)\leqslant t. \quad (1)$$

When S satisfies the condition $e(S)\leqslant t$, we call S a feasible solution, otherwise an infeasible solution.

2.2. Tabu search

Tabu search, proposed by Glover and Laguna [11,12], is a meta-heuristic method that can be used to solve combinatorial optimization problems. It has received widespread attention recently. Its flexible control framework and several spectacular successes in solving NP-hard problems caused rapid growth in its application. The method of neighborhood exploration and the use of short- and long-term adaptive memories distinguish tabu search from local search and other heuristic search methods, and result in lower computational cost and better space exploration.

Tabu search involves a lot of techniques and strategies, but it mainly comes from the use of short-term memories (tabu list) that keep track of recently examined solutions intending to avoid cycling in the solution space exploration (this is usually called the first level of heuristics in TS). Tabu search scheme can be outlined as follows: start with an initial (current) solution x, called a configuration, evaluate the criterion function for that solution. Then, follow a certain set of candidate moves, called the neighborhood $N(x)$ of the current solution x. After a move is performed (i.e., a solution is picked), the move is declared tabu for a predetermined number l of moves, i.e., this move cannot be reversed until a tabu tenure l expires. This means that TS is a dynamic neighborhood method, where the neighborhood of x can change according to the history of the search. However, a tabu move is admissible if it is compliant with an aspiration criterion, usually that of improving the best current solution. At each step, we select the best non-tabu move (may be an ascending move in some situation of the search) from those available moves, and use the improved-best aspiration criterion to allow a move to be considered admissible in spite of its tabu status, i.e., if the tabu move results in a value of the objective function that is better than that of the best-solution known so far, then the aspiration criterion is satisfied and the tabu restriction is relieved. Tabu search saves the best current solution at any time and proceeds iteratively until a chosen termination criterion is satisfied (usually a predefined number of iterations or/and when the best solution was not improved in some predetermined number of iterations). The short-term memory is usually implemented with a first in first out list. Its size equals the tabu tenure l.

Tabu search provides a flexible framework for discrete optimization problem solving. In the following, we use TS to solve the optimal reference subset selection problem, and propose a heuristic neighborhood exploration method and an aspiration criterion to enhance the efficiency of TS.

2.3. Application of tabu search to the optimal reference subset selection

The reference subset is represented by a 0/1 bit string, the kth bit 0 or 1 denotes the absence or presence of the kth sample in the reference subset. Let S_{curr}, S_{next} and S_{best} be the current, next and the best reference subsets, respectively. TL is a first in first out tabu list. It has a predefined length l (tabu tenure).

In our implementation of TS, the neighborhood of each solution $S_{curr} \in P(x)$ is defined as $N(S_{curr})$ consisting of all subsets that differ from S_{curr} in only one sample addition or deletion, that is, all subsets resulting by adding or deleting a sample to or from the current solution S_{curr}. We denote all the subsets resulting by adding a sample to S_{curr} as $N^+(S_{curr})$, and all the subsets resulting by removing a sample from S_{curr} as $N^-(S_{curr})$. We may define a larger neighborhood, such as adding or deleting two or more samples to or from S_{curr}, and randomly pick out a part of them as the candidate moves. However, this will increase the computational cost and decrease the intensification of the space search. We did not adopt this kind of neighborhood in this paper.

In adding or deleting a sample to or from S_{curr}, we consider the following three properties of the resulted reference subset:

(1) The change of the classification error rate before and after adding or deleting a sample.
(2) After adding a sample into S_{curr}, the change of the classification of the samples, which are wrongly classified by S_{curr}.
(3) The distance between the original data set and the resulted reference subset, which is defined as the sum of the distance between each sample in the original data set and its nearest sample of the same class in the reference subset. The intention of doing so is to select the representative samples that are near to the cluster center of the same class samples.

Based on the above demands, in searching the best solution S_{next} among all the candidate solutions in $N^+(S_{curr})$ generated by adding a sample to S_{curr}, we use the following two heuristic criteria:

Criterion 1: Search the S_{next} of the minimal error rate in the candidate subsets $N^+(S_{curr})$. If $e(S_{next}) < e(S_{curr})$, then S_{next} is the best solution in the candidate subsets. If there is more than one solution having the minimal error rate, then select the one that has the minimal distance from the original data set. The distance is defined as above.

Criterion 2: For the minimal error rate S_{next} in the candidate solutions $N^+(S_{curr})$, if $e(S_{next}) \geqslant e(S_{curr})$, then consider selecting such solutions in $N^+(S_{curr})$ which could correctly classify at least one of the samples that are wrongly classified by S_{curr}. Among such candidate solutions, select the solution with minimal error rate or minimal distance. If there are no such candidate solutions, then aspirate the best (minimal error rate) one in TL and start a new search process.

The purpose of criterion 2 is to prevent adding many redundant samples. If only based on criterion 1, many redundant samples may be added. Although they do not deteriorate the classification, they do not help. The above aspiration operation was adopted to avoid the meaningless exchange of samples between the feasible and infeasible regions of the solution space. We will explain this in more detail in Section 4.

The case of deleting a sample from S_{curr} is relatively easy. We may use a criterion similar to the above criterion 1 to select the sample to be deleted based on the minimal error rate and minimal distance criteria between the two subsets.

The reference subset selection algorithm based on tabu search is as follows:

```
(1) Input the original training data set $X$, specify the tabu list length $l$, the error rate threshold $t$.
(2) Generate an initial solution $S_{init}$, set $S_{curr} = S_{init}$, $S_{best} = X$. Let $TL = \phi$, $k = 0$.
(3) (a) Find the best solution $S_{next}$ in the neighborhood of $S_{curr}$. There are two cases:
        If $e(S_{curr}) > t$, then search the best solution $S_{next}$ in $N^+(S_{curr})$ according to criterion 1 and criterion 2;
        If $e(S_{curr}) \leqslant t$, then search the best solution $S_{next}$ among all the solutions in $N^-(S_{curr})$ according to the minimal error rate and minimal distance criterions.
    (b) If $S_{next}$ is in TL and does not satisfy the aspiration criterion, then let $N^+(S_{curr}) = N^+(S_{curr}) - \{S_{next}\}$
```

or $N^-(S_{curr}) = N^-(S_{curr}) - \{S_{next}\}$, respectively, goto 3(a); Otherwise, let $S_{curr} = S_{next}$.
If $e(S_{curr}) \leqslant t$ and $Card(S_{curr}) < Card(S_{best})$, or $Card(S_{curr}) = Card(S_{best})$ and $e(S_{curr}) < e(S_{best})$, then let $S_{best} = S_{curr}$.

(c) If termination condition is satisfied, stop and output the S_{best}; otherwise insert the S_{curr} into TL, $k = k + 1$, Goto (3).

Termination condition is a predefined number of iterations or when there is no improvement of the best solution after a given number of successive rounds. The initial reference subset of tabu search may be null set, or randomly generated subset, or the result of other reference subset selection algorithms. The use of the full original data set is not recommended, as it will take more time to converge.

3. Test data sets

Seven data sets, which have broad spectrum in property, were used to test the proposed reference subset selection methodology. These data sets are as follows:

(1) *The Iris data set* (*IRIS*): The Fisher's Iris data set contains 150 four-dimensional feature vectors from three classes: Setosa, Virginica and Versicolor. Each class contains 50 samples.

(2) *The I-I data set* (*I-I*): The I-I data set, used by Fukunaga and Hamamoto [13,14], was generated from two classes of n-dimensional normal distributions $N(\mu_i, \sum_i)$, $i = 1, 2$. The parameters are

$$\mu_1 = [0, \ldots, 0]^{\mathrm{T}}, \quad \mu_2 = [\mu, 0, \ldots, 0]^{\mathrm{T}}, \quad \textstyle\sum_1 = \sum_2 = I_n,$$

where μ_1 the n-dimensional zero vector and I_n the $n \times n$ identity matrix. The value μ controls the overlap between the two distributions. We used $\mu = 2.56$ in the experiments, which results in a Bayes error rate of 10%. When the dimensionality of the data changes, the Bayes error rate stays the same for a fixed μ.

(3) *The ring shaped data set* (*RING*): This is a two-class problem defined in two-dimensional plane. The classes are circumscribed by three circles of radius r_1, r_2 and r_3, respectively (Fig. 1). One class is represented by the gray areas, and the other class by a white ring. Samples are uniformly distributed over the corresponding areas.

(4) *The diagonal data set* (*DIAGONAL*): It is a two-class, two-dimensional data set. Each class consists of two normal distributions as follows:

$$p_1(x) = \tfrac{1}{2}N(\mu_{11}, I_n) + \tfrac{1}{2}N(\mu_{12}, I_n),$$

$$p_2(x) = \tfrac{1}{2}N(\mu_{21}, I_n) + \tfrac{1}{2}N(\mu_{22}, I_n),$$

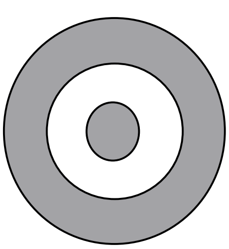


Fig. 1. Ring data set.

where $\mu_{11} = [0, 0]^{\mathrm{T}}$, $\mu_{12} = [\mu, \mu]^{\mathrm{T}}$, $\mu_{21} = [\mu, 0]^{\mathrm{T}}$, $\mu_{22} = [0, \mu]^{\mathrm{T}}$. The Bayes error rate of this data set is determined by μ. $\mu = 3.5$ is used in the experiments.

(5) *The interval data set* (*INTERVAL*): It is a two-class data set taken from Ref. [13] of Fukunaga. Each class consists of two normal distributions as follows:

$$p_1(x) = \tfrac{1}{2}N(\mu_{11}, I_n) + \tfrac{1}{2}N(\mu_{12}, I_n),$$

$$p_2(x) = \tfrac{1}{2}N(\mu_{21}, I_n) + \tfrac{1}{2}N(\mu_{22}, I_n),$$

where $\mu_{11} = [0, 0, \ldots, 0]^{\mathrm{T}}$, $\mu_{12} = [6.58, 0, \ldots, 0]^{\mathrm{T}}$, $\mu_{21} = [3.29, 0, \ldots, 0]^{\mathrm{T}}$, $\mu_{22} = [9.87, 0, \ldots, 0]^{\mathrm{T}}$. Even when the dimensionality of the data changes, the Bayes error rate of this data set remains 7.5%.

(6) *The Ness data set* (*NESS*): This data set was used in Ref. [15] by Ness. The samples were independently generated from two n-dimensional normal distributions $N(\mu_i, \sum_i)$ with the following parameters:

$$\mu_1 = [0, \ldots, 0]^{\mathrm{T}}, \quad \mu_2 = [\Delta/2, 0, \ldots, 0, \Delta/2]^{\mathrm{T}},$$

$$\textstyle\sum_1 = I_n, \quad \sum_2 = \begin{pmatrix} I_{n/2} & O \\ O & \frac{1}{2}I_{n/2} \end{pmatrix},$$

where Δ is the Mahalanobis distance between class ω_1 and class ω_2. The Bayes error rate varies depending on the value of Δ as well as n.

(7) *The VMD data set* (*VMD*): This data set was independently generated from two n-dimensional normal distributions $N(\mu_i, \sum_i)$, $i = 1, 2$. The mean vector of the second class is decreased by degrees

$$\mu_1 = [0, \ldots, 0]^{\mathrm{T}}, \quad \mu_2 = \left[\mu, \frac{\mu}{2}, \frac{\mu}{3}, \ldots, \frac{\mu}{n}\right]^{\mathrm{T}},$$

$$\textstyle\sum_1 = \sum_2 = I_n.$$

Table 1 summarized the seven data sets including the dimension, number of classes, number of samples, and the values of parameters. In the data set RING, the samples for the two classes are 120 and 60, respectively. In other data sets (except IRIS), the number of training samples for the two classes are equal.

Table 1
Data sets

Data	Dimension	Classes	Number of samples	Parameters
IRIS	4	3	150	—
I-I	6	2	300	$n=6,\ \mu=2.56$
RING	2	2	180	$r_1=1,\ r_2=2,$ $r_3=3$
DIAGONAL	2	2	100	$n=2,\ \mu=3.5$
INTERVAL	5	2	300	$n=5$
NESS	10	2	300	$n=10,\ \Delta=2.0$
VMD	10	2	200	$n=10,\ \mu=3.0$

4. Experimental results and analyses

We have carried out two kinds of experiments on these data sets. In the first kind of experiment, we set the error rate threshold to zero. Thus, the resulting reference subsets are the consistent subsets of the original data sets. The second kind of experiment uses a small non-zero error rate threshold and independent training and test data sets. The sizes of the obtained reference subsets and error rates on independent test data sets are used to compare the proposed algorithm with the CNN and MCS algorithms.

4.1. The optimal consistent subsets by tabu search

Setting the error rate threshold to zero, tabu search can select the near optimal (minimal) consistent subsets. The experimental results are shown in Table 2. Euclidean distance is used in these experiments. The classifications are made with the nearest prototype classifier. For comparison, we also implemented CNN and MCS algorithms and the results are also shown in Table 2. For CNN method, we show the best and the average results of 10 runs. The initial solutions of tabu search were null set or randomly generated subsets. For randomly generated initial solutions, 10 runs were executed on each data set. Table 2 lists the best, the worst, and the average results, along with the standard deviations. For null set initial solution, tabu search only runs once, as the solution is unique according to our TS-based algorithm. In experiments, the length of the TL is set to equal the number N of samples in the training set, respectively. The termination condition is $2N$ times of iterations or termination after N times of iterations without improvement of the solutions.

From Table 2, we see that the resulted consistent subsets by CNN have more samples than that of MCS and TS. As previously mentioned, CNN is also very sensitive to the initial samples and to the order of consideration of samples. Meanwhile MCS method resulted in smaller consistent subsets than CNN (except DIAGONAL data set). However, the resulting consistent sets of MCS are not minimal. The TS method obtained even smaller consistent subsets than that of MCS on all the seven data sets. Although to some extent TS is sensitive to the initial solutions, even in the worst case, the consistent subsets are still smaller than that of MCS (for IRIS, the same number).

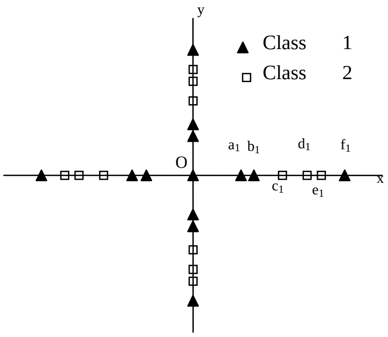
Based on the above experimental results, we analyze the MCS method and the proposed aspiration criterion further:

(1) In Ref. [7], the author believed that though no formal mathematical proof has been established, the MCS algorithm realized the minimality of the consistent subset size. Ref. [8] also cited this minimality. However, counter examples to the author's conjecture are later given by Kuncheva and Bezdek in Ref. [9] and Cerveron and Fuertes in [10] independently. They obtained a 12-element consistent subset [9] and even an 11-element one [10] for the popular IRIS data set, while the minimal consistent subset found by MCS algorithm was a 15-element subset. Our experiments also found an 11-element consistent subset many times. The experiments also demonstrate that the minimality goal in consistent subset selection of the MCS algorithm does not realize not only for the IRIS data set, but also for the other data sets. Moreover, the differences in the size of the resulting consistent subsets by MCS algorithm and the TS-based method are obvious.

(2) In Ref. [7], the author thought the fact that the consistent subsets is monotonically reducing is assured by

Table 2
Consistent subsets by CNN, MCS and tabu search

Data set	Original samples	CNN		MCS	TS (null initial set)	TS (Random m samples)			
		Best	Average			m	Best	Worst	Average (±s.d.)
IRIS	150	18	19.8	15	15	15	11	15	14.0 (±0.8)
I-I	300	90	97.4	74	62	30	55	71	63.1 (±5.0)
RING	180	44	51.0	43	28	18	26	35	30.7 (±3.1)
DIAGONAL	100	12	16.2	13	6	10	6	10	7.5 (±1.3)
INTERVAL	300	98	104.1	89	58	30	57	72	68.8 (±4.4)
NESS	300	67	72.6	46	29	30	26	39	33.9 (±3.9)
VMD	200	29	34.4	23	4	20	4	13	7.7 (±2.6)



Coordinates

sample point	class	x	y
o	1	0.0	0.0
a1	1	1.0	0.0
b1	1	1.3	0.0
c1	2	1.9	0.0
d1	2	2.4	0.0
e1	2	2.7	0.0
f1	1	3.2	0.0

Fig. 2. A counterexample to the monotonically reducing MCS.

using the MCS algorithm. In the following, we first briefly introduce the MCS algorithm, then provide a counterexample to illustrate that this is not always true.

The MCS method is based on the concept of NUNS, the nearest unlike neighbor subset [7]. The NUN subset is defined as the unique set of all samples which are the nearest unlike neighbors of one or more of the given samples. Based on this concept, for every given sample, the sufficient condition for its correct classification, i.e., for consistency, is the presence of a sample from its own class that is closer than its NUN within MCS. MCS algorithm employed a mechanism of vote of confidence cast by the given sample and received by such closer-than-NUN samples. The sample with the most such votes represents the prime candidate for inclusion in MCS. Once this is picked, all the samples which were the voters contributing to the selection of the candidate for MCS can be disregarded from further consideration and the vote counts of other candidates are reduced to reflect this. The candidate with the maximum votes after this update becomes the next most effective MCS sample. This process is repeated till all the voters have been taken into account. It is possible that in some cases, the samples may have only one vote of itself. In such cases, these automatically become MCS candidates. It is also possible that the voters to another sample may themselves become candidates for MCS.

Once a candidate MCS set has been identified, the algorithm reexamines the problem as the effective NUN distances are now likely to be larger than before since some NUNs are no longer in the subset under consideration. Thus, there is now scope for reducing the candidate MCS further. For the process to be monotonically reducing, the algorithm has to ensure that the candidate list will only include samples (other than the last MCS candidates) that will not create any new inconsistencies. Thus, the algorithm maintains a candidate consistent set consisting of all samples which are either (a) already present in the current consistent set or (b) whose inclusion will not create an inconsistency (step 5 in the algorithmic procedure of Ref. [7]). According to the algorithms the samples in the consistent set are monotonically reducing. However, in our experiments, we found that this is not always true, and the number of samples in the resulted consistent set increases sometimes. Since the sample distributions are complex in practical problems, and after recounting the NUN distances of each sample and re-voting, the order of the most voted sample may change. This will effect the consistent set not to be monotonically reducing. In the following we construct an example to illustrate this situation. In the experiments we found that cases like this example occurred often.

Table 3

Point	o	a_1	b_1	c_1	d_1	e_1	f_1
Received votes	1	3	3	2	3	2	1
NUN distance	1.9	0.9	0.6	0.6	0.8	0.5	0.5

Fig. 2 is a two class data set. The samples are located on the x- and y-axis. The coordinate values of the samples on the positive x-axis are shown in the table of Fig. 2. The other samples are rotated ones of the samples on the positive x-axis. For the sake of convenience group the points that are symmetrical about the origin to form a set, and call them as O, A, B, C, D, E, F, respectively. According to MCS algorithm, in the first iteration each point in the sets O, A through F has a NUN distance and receives votes are given in Table 3 (the same for the samples in the same set).

So the algorithm obtains $A \cup D \cup F$ as the consistent set, its size is 12. The second iteration recounts the NUN distances and re-votes, at this time each point of A votes o, so the received votes of o are 5. The votes and NUN distances of each point are given in Table 4.

Then, the candidate consistent set becomes $O \cup A \cup D \cup F$. According to the MCS algorithm, the most voted

Table 4

Point	o	a_1	b_1	c_1	d_1	e_1	f_1
Received votes	5	3	—	—	3	—	1
NUN distance	2.4	1.4	1.1	0.9	0.8	0.5	0.8

sample o in the candidate consistent set should be designated as a member of a newly selected consistent set, as this will not create any new inconsistencies. So the second iteration results in a consistent set of $O \cup A \cup D \cup F$. Its size is 13, one larger than the previous size. This conflicts with the concept of Ref. [7].

We noted that in most cases, the MCS was monotonically reducing, but there were exceptions sometimes. This constructive example gives us some hints as to why MCS algorithm failed to attain the minimal consistent subset. The overall analysis of the reasons for failure is beyond the scope of this paper. It will be an interesting work.

(3) In solving constrained optimization problems, the search process often wanders between the feasible and infeasible regions in the solution space. This decreases the efficiency of the search algorithm. As described in Section 2.3, we use an aspiration criterion to avoid the meaningless exchange of samples between the candidate consistent subset and the rest of the samples. Here, we give a simplified example to illustrate the benefit of introducing this aspiration criterion.

Suppose $\boldsymbol{a}$ is such a sample that it will not be correctly classified unless it is in the reference subset by itself. $\boldsymbol{B} = \{\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots, \boldsymbol{b}_k\}$ is a cluster of samples of a class. Provided any element of $\boldsymbol{B}$ is within the reference subset, all the samples of $\boldsymbol{B}$ will be correctly classified. Suppose now that tabu search obtains a reference subset $\boldsymbol{S}$, which satisfies the error rate threshold condition, and $\boldsymbol{a}, \boldsymbol{b}_1 \in \boldsymbol{S}$. The next step of TS will be to try to remove a sample from $\boldsymbol{S}$. After calculation, removing $\boldsymbol{a}$ will result in a minimal error rate. So, TS obtains the subset $\boldsymbol{S} - \{\boldsymbol{a}\}$ (signs '$-$' and '$+$' denote deleting or adding a sample). Suppose the classification error rate with this reference subset becomes already larger than the threshold, then TS will add a sample to the reference subset. If there was no aspiration criterion, TS would add a redundant sample $\boldsymbol{b}_2$ as $\boldsymbol{S}$ is tabu ($\boldsymbol{a}, \boldsymbol{b}_1 \in \boldsymbol{S}$ is in the tabu list). After this the error rate still does not satisfy the threshold, it needs to add further samples, then $\boldsymbol{a}$ is added, and a subset $\boldsymbol{S} + \{\boldsymbol{b}_2\}$ results. It satisfies the error rate threshold. Afterwards, TS algorithm deletes $\boldsymbol{b}_1$, obtains $\boldsymbol{S} + \{\boldsymbol{b}_2\} - \{\boldsymbol{b}_1\}$ (it is not tabu). So, the search process might be

$$\boldsymbol{S} \to \boldsymbol{S} - \{\boldsymbol{a}\} \to \boldsymbol{S} - \{\boldsymbol{a}\} + \{\boldsymbol{b}_2\} \to \boldsymbol{S} + \{\boldsymbol{b}_2\}$$
$$\to \boldsymbol{S} + \{\boldsymbol{b}_2\} - \{\boldsymbol{b}_1\} \ldots .$$

Table 5
Subset sizes by TS with and without aspiration criterion for IRIS (Random initial subset)

Size	With aspiration criterion	Without aspiration criterion
Best	11	14
Worst	16	17
Average	14.2	15.5

Table 6
Subset size differences of TS with and without aspiration criterion for IRIS

Size difference	0	1	2	3
Occurrence time	5	6	7	3

These steps only replace $\boldsymbol{b}_1$ with $\boldsymbol{b}_2$, and such meaningless replacements might continue. Since $\boldsymbol{S} + \{\boldsymbol{b}_2\} - \{\boldsymbol{b}_1\}$ satisfies the threshold, the next step might remove $\boldsymbol{a}$, and the process may be possible as follows:

$$\boldsymbol{S} \to \cdots \to \boldsymbol{S} + \{\boldsymbol{b}_2\} - \{\boldsymbol{b}_1\} \to \cdots \to \boldsymbol{S} + \{\boldsymbol{b}_3\} - \{\boldsymbol{b}_1\}$$
$$\to \cdots \to \boldsymbol{S} + \{\boldsymbol{b}_k\} - \{\boldsymbol{b}_1\} \ldots .$$

Obviously, these processes would decrease the efficiency of the algorithm, especially when the tabu list is short, the algorithm would be trapped in meaningless exchanges of samples between feasible and infeasible solution regions.

In these cases, the search process should add $\boldsymbol{a}$ again and try to remove another sample after removing of $\boldsymbol{a}$ failed. By introducing the aspiration criterion described in Section 2.3, we may achieve the desired search process. For example, when TS obtains $\boldsymbol{S} - \{\boldsymbol{a}\}$, as adding any $\boldsymbol{b}_i \in \boldsymbol{B}$ $(i = 1, \ldots, k)$ cannot decrease the error rate and correctly classify any wrongly classified sample by $\boldsymbol{S} - \{\boldsymbol{a}\}$, the algorithm will aspirate $\boldsymbol{S}$ not to be tabu, then start a new search path. Since at this time $\boldsymbol{S} - \{\boldsymbol{a}\}$ becomes tabu, the algorithm will try to remove some other sample from $\boldsymbol{S}$, and avoid the meaningless exchanges of samples.

In our experiments, situations like this simplified illustrative example often take place. After introducing the aspiration criterion, the efficiency of the algorithm is obviously improved. In order to demonstrate the role of the aspiration criterion, we conducted the following experiments. Using the same initial subsets (20 random initial subsets and a null subset), we implemented the TS algorithms with and without aspiration criterion on IRIS data set, respectively. The sizes of resulting consistent subsets of the random initial subsets are shown in Table 5. For the null initial subset, the sizes of resulted reference subsets are 15 and 17, respectively.

During 21 times of runs, the differences in sizes of the resulted consistent subsets between TS with and without the aspiration criterion are shown in Table 6.

Table 7

		Data set							
Algorithms		IRIS1	IRIS2	I-I	RING	DIAGONAL	INTERVAL	NESS	VMD
(A) *Data sets and the error rate of 1-NN classification*									
Number of training/testing samples		30/120	75/75	300/300	180/3000	100/2000	300/3000	300/3000	200/2000
Error rate of 1-NN (%)		4.17	4.00	17.73	9.44	8.75	13.07	9.10	6.75
(B) *Resulting reference subset sizes*									
CNN (average)		7.1	12.4	97.4	51.0	16.2	104.1	72.6	34.4
MCS		6	9	74	43	13	89	46	23
Tabu ($t = 0.00$)(average)		4.0	8.2	63.0	30.4	7.4	67.8	33.4	7.3
Tabu ($t = 0.05$)(average)		4.0	3.0	11.4	13.6	4.0	12.2	2.0	2.0
Tabu ($t = 0.10$)(average)		3.0	3.0	2.6	9.8	4.0	4.0	2.0	2.0
(C) *Error rates on independent test data sets* (%)									
CNN (average)		5.12	8.43	20.54	12.17	11.94	16.71	14.24	10.48
MCS		6.67	12.00	21.27	11.73	10.60	18.73	15.50	9.45
Tabu ($t = 0.00$)	Best	3.33	6.67	19.23	10.17	6.55	15.57	12.10	4.90
	Worst	5.00	12.00	21.03	12.93	12.90	18.30	14.97	7.55
	Average	3.67	8.27	20.23	11.19	10.28	17.04	13.80	6.36
Tabu ($t = 0.05$)	Best	3.33	4.00	12.47	10.07	6.50	9.73	7.70	5.45
	Worst	5.00	4.00	16.60	14.20	8.70	13.60	7.70	5.45
	Average	4.00	4.00	14.63	12.75	7.55	11.23	7.70	5.45
Tabu ($t = 0.10$)	Best	10.00	4.00	11.33	13.40	5.50	9.33	7.70	5.45
	Worst	12.50	4.00	12.67	19.97	7.50	12.40	7.70	5.45
	Average	11.33	4.00	11.95	16.36	6.57	10.54	7.70	5.45

From Tables 5 and 6, we can see that the effect of the aspiration criterion is notable. The size difference of the minimal consistent subsets is three samples. In the 21 runs of TS with aspiration criterion (termination criterion is 300 iterations or after 150 iterations without improvement), the average steps of obtaining the optimal reference subsets are 49.55 steps, and the average number of activation of the aspiration criterion before the optimal subsets result is 9.90, and the average number per run of activating the aspiration criterion is 67.2. This shows that the aspiration criterion is more often than not practically working and this is helpful to obtain better reference subsets.

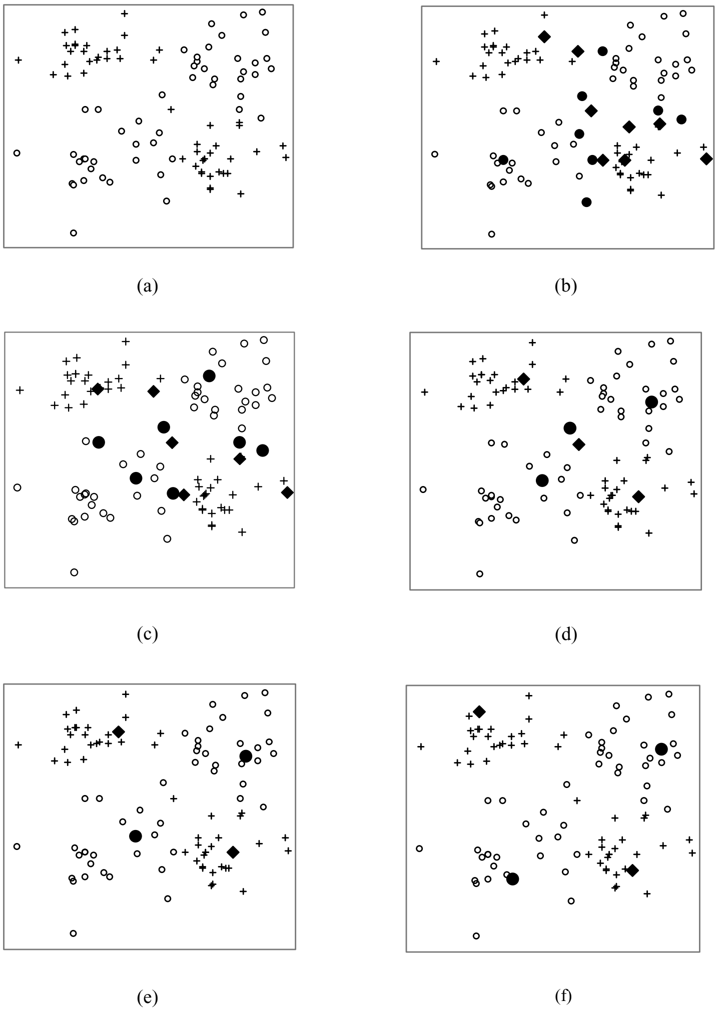
4.2. *Classification performance of the reduced non-consistent reference subsets*

In the previous section, we set the error rate threshold to zero and obtained the consistent subsets of the original data sets. However, in practice due to the finite sample size, the performance of the consistent reference subset may not necessarily be the best in the operational phase. Fukunaga and Hummels show that the 1-NN estimates may be severely biased even for the large sample size if the dimensionality of the data is large [16]. They recommend a decision threshold r to take into account the bias in density estimation [17]. That is, the decision rule can be modified as: classify x into class ω_k if

$$\hat{p}(x \mid \omega_k) > \hat{p}(x \mid \omega_j) + r, \quad \text{for all } j = 1, \ldots, m, \quad j \neq k,$$

where $\hat{p}(x \mid \bullet)$ denotes the estimated density. However, it is difficult to determine the optimal threshold r because of its complexity. We know that the basis of the NN classification comes from the NN density estimation. Therefore, the consistent subset of the original training data set is not necessarily accurate on unknown data. It overfits the training data, and may lack generalization ability. In this section, we set the error rate threshold t of Eq. (1) to be a small nonzero value, investigate the reference subset sample reduction rate and the classification performance of the reduced inconsistent reference subsets on independent test data sets.

In experiments, the error rate thresholds are set to $t = 0.00, 0.05$ and 0.10, respectively. The training and test data sets are independently generated (Table 7(A)). In the training set of RING, the samples of two classes are 120 and 60 and 2000, 1000 in the test set, respectively. In other data sets (except IRIS), the number of training samples and test samples for two classes is equal. The IRIS1 and IRIS2 data sets are two random partitions of the IRIS data set, their numbers of training/test samples are 30/120 and 75/75, respectively. For different error rate thresholds, we repeat the TS-based reference subset selection algorithm five times, one of them uses null set


(a)

(b)

(c)

(d)

(e)

(f)

Fig. 3. Distribution of DIAGONAL and its condensed subsets. (a) diagonal data set. (b) result of CNN. (c) result of MCS. (d) result of TS($t = 0.00$). (e) result of TS($t = 0.05$). (f) result of TS($t = 0.10$).

as an initial set, the other four times use random an initial sets. By using the resulted reference subsets, classify the samples of the test data sets, respectively. The resulting reference subset sizes and the corresponding error rates are tabulated in Table 7(B) and (C), respectively. For comparison, we also conducted the 1-NN classification on the whole (training plus test) data sets, respectively. The error rates are listed in Table 7(A). They provide bases for comparison with other algorithms. For CNN and MCS methods, the resulting reference subset sizes and corresponding error rates are also shown in Table 7(B) and (C), respectively.

From Table 7 the following observations can be drawn

(1) The sizes of resulting reference subsets by MCS algorithm are smaller than that of CNN algorithm. While when $t = 0.00$, the sizes of the resulting reference subsets (they are consistent subsets) by TS are smaller than that of CNN and MCS. When $t = 0.05$ and 0.01, the sizes of the resulting reference subsets by TS are even more small.

(2) For independent test data sets, the error rates of the condensed subsets of MCS are comparable to that of CNN's, but worse than the error rates of 1-NN algorithm. Meanwhile, when $t = 0.00$, the average error rates of the resulting consistent subsets by TS are smaller than that of MCSs and CNNs. This indicates that the consistent subsets resulting from TS have better representative property and generalization ability than that of MCSs and CNNs.

(3) When $t = 0.05$, the average error rates of the resulting reference subsets by TS are smaller than that of 1-NN algorithm. At the same time, the resulting reference subsets of NESS and VMD data sets are rather rational, their error rates approach the Bayes error rates. This confirms that the 1-NN estimates may be biased due to the finite

sample size, and that the performance of the consistent subsets may not necessarily be the best in the operational phase of classifying independent test data sets.

(4) When $t = 0.10$, the I-I, DIAGONAL and INTERVAL data sets obtain quite rational reference subsets, their sizes and classification performances are superior to those of $t = 0.00$ and 0.05.

From the experimental results, we observe that for different data sets, the appropriate thresholds t are also different. It depends on the data distribution. For example, the RING data set needs more samples as reference prototypes. Therefore, as t increases, the number of reference samples becomes smaller, and this may incur the increase of the error rate. For IRIS1 data set, the classification performance at $t = 0.10$ also deteriorates.

In the experiments, we also observe that the sample distributions of the resulting reference subsets by tabu search are quite rational. Fig. 3 shows the sample point distributions of the original DIAGONAL data set and the resulting reference subsets by CNN, MCS and TS ($t = 0.00, 0.05$ and 0.10), respectively.

5. Conclusion

We use TS to select the near optimal reference subset for the nearest neighbor classification. The performance of the proposed algorithm is demonstrated on several data sets of broad spectrum. It shows that the proposed algorithm outperforms the classical and other respectable algorithms in the reference sample reduction rate and classification performance. We also demonstrated that the consistent reference subsets are generally not accurate on independent test data sets. It shows that the TS-based selection method significantly reduces the size of the reference subset and get good generalization capacity. Therefore, it should be considered as a promising tool in the NN classifier design.

References

[1] B.V. Dasarathy, Nearest Neighbor (NN) Norms: NN Pattern Classification Techniques, IEEE Computer Society Press, Los Alamitos, CA, 1991.

[2] P.E. Hart, The condensed nearest neighbor rule, IEEE Trans. Inf. Theory 14 (3) (1968) 515–516.

[3] G.W. Gates, The reduced nearest neighbor rule, IEEE Trans. Inf. Theory 18 (3) (1972) 431–433.

[4] C.W. Swonger, Sample set condensation for a condensed nearest neighbor decision rule for pattern recognition, in: S. Watanade (Ed.), Frontiers of Pattern Recognition, Academic Press, New York, 1972, pp. 511–519.

[5] C.L. Chang, Finding prototypes for nearest neighbor classifiers, IEEE Trans. Comput. 23 (11) (1974) 1179–1184.

[6] P.A. Devijver, J. Kittler, On the edited nearest neighbor rule, Proceedings of the Fifth International Conference on Pattern Recognition, Miami, Florida, 1980, pp. 72–80.

[7] B.V. Dasarathy, Minimal consistent set (MCS) identification for optimal nearest neighbor decision systems design, IEEE Trans. Syst. Man Cybern. 24 (3) (1994) 511–517.

[8] L.I. Kuncheva, Fitness functions in editing k-NN reference set by genetic algorithms, Pattern Recognition 30 (6) (1997) 1041–1049.

[9] L.I. Kuncheva, J.C. Bezdek, Nearest prototype classification: clustering, genetic algorithms, or random search, IEEE Trans. Syst. Man and Cybern. 28 (1) (1998) 160–164.

[10] V. Cerveron, A. Fuertes, Parallel random search and Tabu search for the minimal consistent subset selection problem, Lecture Notes in Computer Science, Vol. 1518, Springer, Berlin, 1998, pp. 248–259.

[11] F. Glover, M. Laguna, Tabu search, in: R.C. Reeves (Ed.), Modern Heuristic Techniques for Combinatorial Problems, McGraw-Hill, Berkshire, pp. 70–150.

[12] F. Glover, M. Laguna, Tabu Search, Kluwer Academic Publishers, Dordrecht, 1997.

[13] K. Fukunaga, Introduction to Statistical Pattern Recognition, 2nd Edition, Academic Press, New York, 1990.

[14] Y. Hamamoto, S. Uchimura, S. Tomita, A bootstrap technique for nearest neighbor classifier design, IEEE Trans. Pattern Anal. Mach. Intell. 19 (1) (1997) 73–79.

[15] J. Van Ness, On the dominance of non-parametric Bayes rule discriminant algorithms in high dimensions, Pattern Recognition 12 (3) (1980) 355–368.

[16] K. Fukunaga, D.M. Hummels, Bias on nearest neighbor error estimates, IEEE Trans. Pattern Anal. Mach. Intell. 9 (1) (1987) 103–112.

[17] K. Fukunaga, D.M. Hummels, Bayes error estimation using Parzen and k-NN procedures, IEEE Trans. Pattern Anal. Mach. Intell. 9 (5) (1987) 634–643.

About the Author—HONGBIN ZHANG received his B.S. degree in Automation in 1968, and M.S. degree in Pattern Recognition and Intelligent System in 1981, both from Tsinghua University, China. From 1986 to 1989 he was an invited researcher in the Department of Information Science of Kyoto University, Japan. From 1993 to 1994 he was a visiting scholar of RPI, USA. Since 1993, he has been a professor of the Institute of Computer, Beijing Polytechnic University, China. His current research interests include pattern recognition, computer vision, neural networks and image processing.

About the Author—GUANGYU SUN received his B.S. degree in Geology from Peking University in 1992 and his M.S. degree from Computer Institute, Beijing Polytechnic University in 1999. His current research interests include pattern recognition and computer vision.