Optimal Selection of Reference Set for the Nearest Neighbor Classification by Tabu Search

ZHANG Hongbin (张鸿宾) and SUN Guangyu (孙广煜)

Computer Institute, Beijing Polytechnic University, Beijing 100044, P.R. China

E-mail: zhb@public.bta.net.cn

Received October 25, 1999; revised March 27, 2000.

Abstract In this paper, a new approach is presented to find the reference set for the nearest neighbor classifier. The optimal reference set, which has minimum sample size and satisfies a certain error rate threshold, is obtained through a Tabu search 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 set may compensate the bias of the nearest neighbor estimate. An aspiration criterion for Tabu search is introduced, which aims to prevent the search process from the inefficient wandering between the feasible and infeasible regions in the search space and speed up the convergence. Experimental results based on a number of typical data sets are presented and analyzed to illustrate the benefits of the proposed method. Compared to conventional methods, such as CNN and Dasarathy's algorithm, the size of the reduced reference sets is much smaller, and the nearest neighbor classification performance is better, especially when the error rate thresholds are set to appropriate nonzero values. The experimental results also illustrate that the MCS (minimal consistent set) of Dasarathy's algorithm is not minimal, and its candidate consistent set is not always ensured to reduce monotonically. A counter example is also given to confirm this claim.

Keywords nearest neighbor classification, Tabu search, reference set

1 Introduction

Nearest Neighbor (NN) classification is one of the important nonparametric classification methods and has been studied at length. It is well known that the main drawbacks of NN classifiers in practice are their computational demands and requiring a lot of memory. Numerous studies have been carried out to overcome these limitations. Dasarathy gives 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 the available instances, which is called bootstrap method in statistics. The objective of reducing the number of reference samples is of course the computational efficiency of the classification phase, or/and making the resulted classification and generalization more reliable. The very early study of this kind was probably that of Hart[2], who presented the "Condensed Nearest Neighbor Rule" (CNN). His method aims to ensure that the condensed set is consistent with the original set, i.e., all of the original samples are correctly classified by the condensed set under the NN rule. Hart's method indeed ensures consistency, but the condensed subset is not minimal, and is sensitive to the initial ordering of the input samples. Under the same idea of picking appropriate samples

Supported by the National Natural Science Foundation of China (No.69675007) and Beijing Municipal Natural Science Foundation (No.4972008).

from the original data set onto the reference set by adding and deleting samples, there are "Reduced Nearest Neighbor Rule" of Gates[3], and "Iterative Condensation Algorithm" of Swonger[4]. All these algorithms aim at reducing the size of the condensed set. The method proposed by Chang created a reference set by generating new representative prototypes[5]. These prototypes were not selected from the original set. They were generated by merging the nearest neighbors of the same class as long as such merging did 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 resulted classification more reliable, especially those 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 editing process tend to infinity, the 1-NN classification on the edited reference set will lead to Bayesian decision. But in practice, we usually have some finite samples, and the MULTIEDIT performance needs to be studied.

Recently Dasarathy presented a condensing algorithm based on the concept of the Nearest Unlike Neighbor Subset (NUNS)[7]. The algorithm introduced a voting mechanism to select the Minimal Consistent Set (MCS). Dasarathy claimed that the candidate consistent set is monotonically reducing during the iterative process, and conjectured that the cardinality of the obtained MCS is the smallest one among all the consistent subsets. This optimality of the smallest size of the attained MCS is also cited in [8]. In this paper we illustrate that this is not true. Though the MCS obtained by Dasarathy's algorithm generally has less samples, but it is not minimal. We will also give a counter example to show that the consistent subset is not always monotonically reducing.

In this paper we treat the selection of reference set as an optimization problem, that is to minimize the number of the reference samples while constrained by some error rate of classification. We use Tabu Search (TS) to solve this constrained combinatorial optimization problem. In Section 2, the TS algorithm for the optimal selection of reference set is described. The experimental data sets are given in Section 3. This is followed by the experimental results and analyses in Section 4. A conclusion is given in Section 5.

2 Optimal Selection of Reference Set by Tabu Search

2.1 Optimal Selection of Reference Set for Nearest Neighbor Classification

The optimal selection of reference sample set can be described as the following optimization problem.

Let $X = \{x_1, x_2, \ldots, x_N\}$ be the original training set for NN classification. Each sample has a known class label from the set $\Omega = \{\omega_1, \omega_2, \ldots, \omega_M\}$. Let $P(X)$ denote 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 S as a reference set, and t be the tolerable error rate threshold. The problem can be formalized as follows:

$$\text{Find } S^* \text{ such that } Card(S^*) = \min Card(S), \quad \text{s.t. } S \in P(X), e(S) \leq t. \tag{1}$$

In the following we will use Tabu search to solve this constrained optimization problem.

2.2 Tabu Search

Tabu search, proposed by Glover[9], is a 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 led to the rapid growth of its application. It differs from the local search in the sense that Tabu search allows moving to a new solution which makes the objective function worse in

the hope that it will not be trapped in suboptimal solutions. Tabu search uses a short-term memory, called tabu list, to record and guide the process of the search. In addition to the tabu list, we can also use long-term memories and *a priori* information about the solutions to improve the intensification and diversification of the search.

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. If the best of these moves is not tabu, or if the best is tabu but satisfies the aspiration criterion, then pick that move and consider it to be the new current solution. Repeat the procedure for a certain number of iterations. On termination the best solution obtained so far is the solution of the Tabu search. Note that the solution that is picked at a certain iteration is put in the tabu list (TL) so that it is not allowed to be reversed in the next l iterations. l is the size of TL. When the length of tabu list reaches that size, then the first solution on TL is freed from being tabu and the new solution enters that list. TL acts as a short-term memory. By recording the history of the searches, Tabu search can control the direction of the following searches. The aspiration criterion could reflect the value of the objective function, i.e., if the tabu solution results in a value of the objective function that is better than the best known so far, then the aspiration is satisfied and the tabu restriction is relieved.

The framework of a Tabu search algorithm can be summarized as follows:

(1) Generate an initial solution x_{init}. Set $x_{\text{curr}} = x_{\text{best}} = x_{\text{init}}$. Let $k = 1$, $TL = \emptyset$.

(2) Pick out a certain number of solutions from the neighborhood of x_{curr} to form a candidate solution set $N(x_{\text{curr}})$.

(3) a) If $N(x_{\text{curr}}) = \emptyset$, goto (2) to regenerate the candidate set. Otherwise, find out the best solution y in $N(x_{\text{curr}})$.

b) If $y \in TL$ and y doesn't satisfy the aspiration condition, let $N(x_{\text{curr}}) = N(x_{\text{curr}}) - \{y\}$, then goto a). Otherwise, let $x_{\text{curr}} = y$. If y is better than x_{best}, let $x_{\text{best}} = x_{\text{curr}}$.

c) If termination condition is satisfied, stop and output the x_{best}, otherwise insert x_{curr} to the tail of TL. If TL reaches a predefined size, free the first one. Let $k = k+1$. Goto (2).

For more details on Tabu search, the reader is encouraged to refer to Glover[9].

2.3 Application of Tabu Search to Optimal Selection of Reference Set

In this section, we present our Tabu search-based algorithm for reference set selection problem.

The reference set is represented by a 0/1 bit string, the k-th bit denotes the presence or absence of the k-th sample in the reference set. Let S_{curr}, S_{next} and S_{best} be current, next and the best reference sets respectively. TL is a first in first out tabu list. It has a predefined length l.

The reference set selection algorithm based on Tabu search is as follows:

(1) Generate an initial solution S_{init}, set $S_{\text{curr}} = S_{\text{init}}$, $S_{\text{best}} = X$ (the original data set). Let $TL = \emptyset$.

(2) Insert the new solution to the tail of the tabu list, $TL = TL \cup \{S_{\text{curr}}\}$.

(3) Modify the best solution. If $e(S_{\text{curr}}) \leq t$ and $Card(S_{\text{curr}}) < Card(S_{\text{best}})$, or $Card(S_{\text{curr}}) = Card(S_{\text{best}})$ and $e(S_{\text{curr}}) < e(S_{\text{best}})$, then let $S_{\text{best}} = S_{\text{curr}}$.

(4) Search the optimal solution in the neighborhood of S_{curr}. There are two cases:

If $e(S_{\text{curr}}) \leq t$, i.e., the S_{curr} satisfies the error rate threshold, search the optimal solution S_{next} among all the sets that simultaneously satisfy the following conditions: 1) $S_{\text{next}} \subset S_{\text{curr}}$, 2) $Card(S_{\text{next}}) = Card(S_{\text{curr}}) - 1$, and 3) $S_{\text{next}} \notin TL$. That is all the non-tabu sets generated by removing a sample from S_{curr} respectively.

If $e(S_{\text{curr}}) > t$, i.e., the S_{curr} exceeds the error rate threshold, search the optimal solution S_{next} among all the sets that simultaneously satisfy the following conditions: 1) $S_{\text{curr}} \subset S_{\text{next}}$, 2)

$Card(S_{\text{next}}) = Card(S_{\text{curr}}) + 1$, and 3) $S_{\text{next}} \notin TL$. That is all the non-tabu sets generated by adding a sample to S_{curr} respectively.

The criteria of optimality in adding or removing a sample will be explained in detail afterwards.

(5) Let $S_{\text{curr}} = S_{\text{next}}$, goto (2).

The termination condition is a predefined number of iteration rounds or/and that there is no improvement of the solutions in some successive rounds.

In adding or deleting a sample, the following three properties of the condensed reference subset are considered:

1) The change of the classification error rate before and after adding or deleting a sample.

2) The change of the classification of the samples which are wrongly classified before adding a sample into the condensed reference subset.

3) The distance between the original data set and the resulted reference subset from or into which a sample is deleted or added. The distance between the resulted reference subset and the original data set is defined as the sum of the distances 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 which are near to the cluster center of the samples.

In searching the optimal S_{next} among all the candidate sets generated by adding a sample to S_{curr}, we use the following two criteria.

Criterion 1. Search the S_{next} of the minimal error rate in the candidate sets. If $e(S_{\text{next}}) < e(S_{\text{curr}})$, then S_{next} is the optimal solution in the candidate sets. If there are two or more solutions 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 sets, if $e(S_{\text{next}}) \geq e(S_{\text{curr}})$, then consider selecting such samples which could correctly classify at least one of the samples that are wrongly classified by S_{curr}. Among such candidate samples, select the sample with minimal error or minimal distance. If all the candidate samples are in TL, then aspirate the best (minimal error) one among them. The purpose of Criterion 2 is preventing adding many redundant samples. If only based on Criterion 1, many redundant samples may be added. Though they do not deteriorate the classification, but have no help. Adopting aspiration operation is to avoid the meaningless exchange of samples between the feasible and infeasible areas 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 between the two sets.

The initial reference set of Tabu search may be null set, randomly generated set, or the result of other algorithm. It is not recommended to use the full original data set, that will take more time to converge.

3 Test Data Set

Seven data sets were used to test the new TS-based methodology. These data sets have broad scope in property as shown in Table 1. We have carried out two types of experiments on these data sets. In the first type of experiments, we set the error rate threshold equal to zero. Thus the resulted reference sets are the consistent subsets of the original data sets. The second type of experiments uses a small nonzero error rate threshold, and independent training and test data sets. The sizes of the obtained reference sets and error rates for independent test data sets are used to compare the proposed algorithm with the CNN and Dasarathy's algorithms. (In the following we will call Dasarathy's algorithm as MCS algorithm also. The meaning of MCS can be found from the context.)

Table 1. Data Set

Data	Dimension	Classes	Number of Samples (Training/Test)	Parameters
IRIS	4	3	150	—
I-I	6	2	300/3000	$n = 6, \mu = 2.56$
RING	2	2	180/3000	$r_1 = 1, r_2 = 2, r_3 = 3$
DIAGONAL	2	2	100/2000	$n = 2, \mu = 3.5$
INTERVAL	5	2	300/3000	$n = 5$
NESS	10	2	300/3000	$n = 10, \Delta = 2.0$
VMD	10	2	200/2000	$n = 10, \mu = 3.0$

The seven data sets are as follows.

(1) The Iris data set (IRIS)

The Fisher's Iris data set contains 150 4-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 was used by Fukunaga and Hamamoto in [10, 11]. The samples were independently generated from two classes of n-dimensional normal distributions $N(\mu_i, \Sigma_i)$, $i = 1, 2$. The parameters are:

$$\mu_1 = [0, \ldots, 0]^{\mathrm{T}}, \mu_2 = [\mu, 0, \ldots, 0]^{\mathrm{T}}, \Sigma_1 = \Sigma_2 = I_n.$$

Here μ_1 is the n-dimensional zero vector and I_n is 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 led to a Bayes error rate of 10%. When the dimensionality of the data changes, the Bayes error rate remains unchanged for a fixed μ.

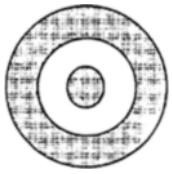


Fig.1. Ring data set.

(3) The ring-shaped data set (RING)

This is a two-class problem defined in 2-dimensional plane. The classes are circumscribed by three circles of radii r_1, r_2 and r_3 respectively (Fig.1). One class is represented by the gray areas, and the other class by white ring. Samples are uniformly distributed over the corresponding areas.

(4) The diagonal data set (DIAGONAL)

It is a two-class, 2-dimensional data set. Each class consists of two normal distributions as follows:

$$p_1(x) = \frac{1}{2}N(\mu_{11}, I_n) + \frac{1}{2}N(\mu_{12}, I_n),$$
$$p_2(x) = \frac{1}{2}N(\mu_{21}, I_n) + \frac{1}{2}N(\mu_{22}, I_n),$$

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. We used $\mu = 3.5$ in the experiments.

(5) The interval data set (INTERVAL)

It is a two-class data set taken from [10]. Each class consists of two normal distributions as follows:

$$p_1(x) = \frac{1}{2}N(\mu_{11}, I_n) + \frac{1}{2}N(\mu_{12}, I_n),$$
$$p_2(x) = \frac{1}{2}N(\mu_{21}, I_n) + \frac{1}{2}N(\mu_{22}, I_n),$$

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 at 7.5%.

(6) The Ness data set (NESS)

This data set was used in [12] by Ness. The samples were independently generated from two n-dimensional normal distributions $N(\mu_i, \Sigma_i)$ with the following parameters:

$$\mu_1 = [0, \ldots, 0]^{\mathrm{T}}, \mu_2 = [\Delta/2, 0, \ldots, 0, \Delta/2]^{\mathrm{T}},$$
$$\Sigma_1 = I_n, \Sigma_2 = \begin{pmatrix} I_{n/2} & 0 \\ 0 & \frac{1}{2} I_{n/2} \end{pmatrix}$$

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 $N(\mu_i, \Sigma_i)$, $i = 1, 2$. The mean vector of the second class is decreased by degrees:

$$\mu_1 = [0, \ldots, 0]^{\mathrm{T}}, \mu_2 = \left[\mu, \frac{\mu}{2}, \frac{\mu}{3}, \ldots, \frac{\mu}{n}\right]^{\mathrm{T}}, \Sigma_1 = \Sigma_2 = I_n.$$

Table 1 summarizes the seven data sets including the dimension, number of classes, number of samples, and the values of parameters. In the training set of RING, the numbers of samples of two classes are 120 and 60 respectively, and 2000,1000 in the test set. In other data sets, the numbers of training samples and test samples are equal.

4 Experimental Results and Analyses

4.1 The Optimal Consistent Set Obtained by Tabu Search

Setting the error rate threshold to zero, Tabu search can select the optimal (minimal) consistent sets. The results corresponding to these seven data sets are shown in Table 2. The Euclidean distance is used in these experiments. The classifications are made by 1-NN. For comparison, we implemented the CNN and MCS algorithms and the results are also shown in Table 2. For CNN method, we show the best and the average results in 10 runs. The initial solutions of Tabu search were null set or randomly generated. 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 lengths of the TL are set to equal the number N of samples in the training sets respectively. The termination condition is $2N$ times of iterations or that during N times of iterations. There is no improvement of the solutions.

Table 2. Consistent Set Obtained 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)

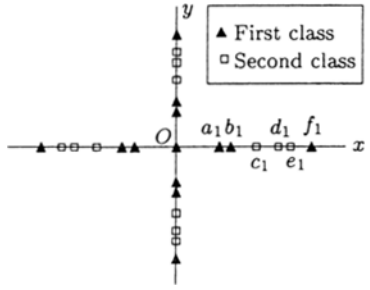
From Table 2, we see that the derived consistent sets by CNN have more samples than those by MCS and TS. As previously mentioned, CNN is also sensitive to the order of samples. Meanwhile MCS method resulted in smaller consistent sets than CNN (except DIAGONAL data set). But the resulted consistent sets of MCS are not minimal. The TS method obtained even smaller sets than those of MCS for all of the seven data sets.

Though to some extent TS is sensitive to the initial solutions, but even in the worst case, the consistent set is still smaller than that of MCS (for IRIS, it is the same).

Based on the above experimental results, we analyze the MCS method and the aspiration criterion of Tabu search further.

1) In [7] Dasarathy claimed that though no formal mathematical proof was established, but he tended to consider the MCS to be the optimal, i.e., the MCS could attain the true minimality of the consistent subset size. Kuncheva also quoted this minimality in [8]. However, the above experiments show that the consistent subsets obtained by MCS are not minimal. For example, for IRIS data set the MCS obtained definitely consistent subset containing 15 samples (see [7] and our Table 2), but our TS-based method obtained a subset containing only 11 samples.

2) In MCS method, Dasarathy maintained a candidate consistent set consisting of all samples either (a) which were already present in the current consistent set, or (b) whose inclusion would not create inconsistency. He asserted that the samples in the consistent set are monotonically reducing. In our experiments we found that Dasarathy's effort is not always effective, and the number of samples in the consistent set increases sometimes. Since after recounting the NUN (Nearest Unlike Nerghbor) distances of each instance and revoting, the order of the most voted sample may change. This will cause the consistent set not to reduce monotonically. In the following we give an example to illustrate this situation.



Coordinates

Sample point	Class	x	y
O	1	0.0	0.0
a_1	1	1.0	0.0
b_1	1	1.3	0.0
c_1	2	1.9	0.0
d_1	2	2.4	0.0
e_1	2	2.7	0.0
f_1	1	3.2	0.0

Fig.2. A counter example to monotonically reducing.

Fig.2 is a data set of two classes. The samples are located on the x and y axes. The coordinate values of the samples on the positive x-axis are shown in the table of Fig.2. The other samples are rotated images of the samples on the positive x-axis. For 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 Dasarathy's algorithm, in the first iteration each point in the set O, A ~ F has an NUN distance and gets a vote as follows (identical for the samples in the same set).

Point	O	a_1	b_1	c_1	d_1	e_1	f_1
The Vote	1	3	3	2	3	2	1
NUN Distance	1.9	0.9	0.6	0.6	0.8	0.5	0.5

So the algorithm obtains $A \cup D \cup F$ as the candidate consistent set, and its size is 12. The second iteration recounts the NUN distances and revotes. At this time each point in A votes O, so the vote of O is 5. The votes and NUN distances of each point are as follows:

Point	O	a_1	b_1	c_1	d_1	e_1	f_1
The Vote	5	3	3	2	3	3	1
NUN Distance	2.4	1.4	1.1	0.9	0.8	0.5	0.8

Then the candidate consistent set becomes $O \cup A \cup D \cup F$. According to Dasarathy's algorithm, the most voted sample O in this candidate consistent list should be designated as a member of a newly selected consistent set, and this will not create any new inconsistencies.

So the second iteration gives a consistent set of $O \cup A \cup D \cup F$. Its size is 13, larger than the previous size. This conflicts with the claim of [7].

Since the above reason, the sizes of the MCS listed in Table 2 are not the final results, they are the minimum ever obtained in iterations. We found that in most cases the MCS is monotonically reducing, but there are exceptions sometimes.

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

Suppose a is such a sample that it won't be correctly classified unless it by itself is in the reference set. $B = \{b_1, b_2, \ldots, b_k\}$ is a cluster of samples of a class. Provided anyone of B is in the reference set, it can classify all samples of B. Suppose Tabu search obtains a reference set S, which satisfies the error rate threshold, and $a, b_1 \in S$. The next step of TS will try to remove a sample from S. After calculation removing a will lead to the minimal error rate. So TS obtains the subset $S - \{a\}$ (signs '$-$' and '$+$' denote deleting or adding a sample). Suppose the classification error rate of this set becomes larger than the threshold, then TS will add a sample to the reference set. If there is not aspiration criterion, TS will add a redundant sample b_2 as S is tabu ($a, b_1 \in S$ is in the tabu list). After this the error rate still doesn't satisfy the threshold, it is needed to add sample further, then a is added, and a subset $S + \{b_2\}$ is obtained. It satisfies the threshold. Afterwards, TS algorithm deletes b_1, obtains $S + \{b_2\} - \{b_1\}$ (it is not tabu). So the search process would be

$$S \to S - \{a\} \to S - \{a\} + \{b_2\} \to S + \{b_2\} \to S + \{b_2\} - \{b_1\} \cdots$$

These search steps only replace b_1 with b_2, and such meaningless replacements will continue. Since $S + \{b_2\} - \{b_1\}$ satisfies the threshold, the next step will remove a, and the process may be as follows:

$$S \to \cdots \to S + \{b_2\} - \{b_1\} \to \cdots \to S + \{b_3\} - \{b_1\} \to \cdots \to S + \{b_k\} - \{b_1\} \cdots$$

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

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

4.2 Classification Performance of the Reduced Reference Set

In the above section, we set the error rate threshold to be zero and obtained the consistent set of the original data set. However in practice due to the finite sample size the performance of the consistent set may not necessarily be the best in the operational phase of classifying an independent test data set. 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[13]. They recommend a decision threshold t to take into account the bias in density estimation[14]. That is, the decision rule can be modified as: classify x into class ω_k if

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

where $\hat{p}(x|\bullet)$ denotes the estimated density. But the optimal selection of the decision threshold t is difficult because of its complexity. In this section we set the error rate threshold to be a small nonzero value, and investigate the reference sample reduction rate and the classification performance of the condensed reference subset.

In the following experiments, the error rate thresholds were set to $t = 0.00$, 0.05 and 0.10 respectively. The training data sets and test data sets are independent. For different thresholds and data sets, repeat TS five times, one of them uses null set as initial set, the other four times use randomly selected samples as initial sets. Use the reference sets resulted from the training sets, classify the samples of the test data sets respectively, and the experimental results are summarized in Table 3. For comparison, we also conducted the 1-NN classification on all data sets, and the CNN, MCS methods which used independent training and test data sets. These experimental results are also listed in Table 3. The IRIS1 and IRIS2 are two random partitions of the IRIS data set. Their numbers of training/test samples are 30/120 and 75/75 respectively.

Table 3. Classification Performance of the Condensed Sets

Algorithms \ Data Sets			IRIS1	IRIS2	I-I	RING	DIA-GONAL	INTER-VAL	NESS	VMD
Original Training Set Size			30	75	300	180	100	300	300	200
1-NN	Error rate (%)		4.17	4.00	17.73	9.44	8.75	13.07	9.10	6.75
CNN	Cond. subset size (average)		7.1	12.4	97.4	51.0	16.2	104.1	72.6	34.4
	Error rate (average) (%)		5.12	8.43	20.54	12.17	11.94	16.71	14.24	10.48
MCS	Cond. subset size		6	9	74	43	13	89	46	23
	Error rate (%)		6.67	12.00	21.27	11.73	10.60	18.73	15.50	9.45
Tabu ($t = 0.0$)	Cond. subset size (average)		4.0	8.2	63.0	30.4	7.4	67.8	33.4	7.3
	Error rate (%)	Best	3.33	6.67	19.23	10.17	12.90	15.57	12.10	4.90
		Worst	5.00	12.00	21.03	12.93	6.55	18.30	14.97	7.55
		Average	3.67	8.27	20.23	11.19	10.28	17.04	13.80	6.36
Tabu ($t = 0.05$)	Cond. subset size (average)		4.0	3.0	11.4	13.6	4.0	12.2	2.0	2.0
	Error rate (%)	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 ($t = 0.10$)	Cond. subset size (average)		3.0	3.0	2.6	9.8	4.0	4.0	2.0	2.0
	Error rate (%)	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 Table 3 we see that when $t = 0.00$, the sizes of the condensed consistent sets are smaller than those of CNN and MCS. When $t = 0.05$, the sizes of the condensed sets are much smaller than those of $t = 0.00$, and the condensed subsets for NESS and VMD are rather rational, their classification performances approach the Bayes errors. When $t = 0.10$, the DIAGONAL, I-I and INTERVAL obtain quite rational reference sets, their sizes and classification performances are superior to those of $t = 0.00$.

From the experimental results we also observe that for different data sets the appropriate threshold t is also different. It depends on the data distribution. For example, the RING data set needs more samples as reference prototypes. Therefore as t increasing, 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 $t = 0.10$ is also deteriorated. Through experiments, we observe that for the data sets used above the appropriate thresholds are as follows:

Data Set	IRIS	I-I	RING	DIAGONAL	INTERVAL	NESS	VMD
Threshold	0.05	0.10	0.00	0.10	0.10	0.05	0.05

In experiments, it was also shown that the sample distributions of the condensed reference

subsets by Tabu search are quite rational. Fig.3 shows the sample point distributions of the original DIAGONAL data set and the condensed reference subsets by CNN, MCS and TS ($t = 0.00$, 0.05 and 0.10) respectively.

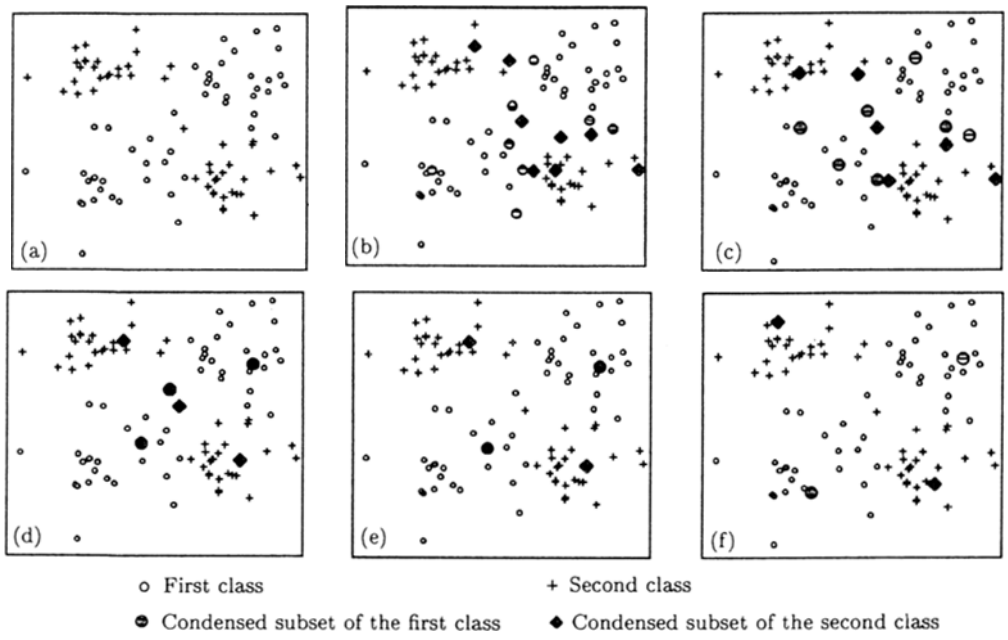


Fig.3. Distribution of DIAGONAL and its condensed subsets. (a) Diagnal data set. (b) Result of CNN. (c) Result of MCS. (d) Result of TS ($t = 0.0$). (e) Result of TS ($t = 0.05$). (f) Result of TS ($t = 0.01$).

5 Conclusion

We have used TS to select the optimal reference subset for the nearest neighbor classification. The performance of the proposed algorithm was demonstrated for several data sets. It is shown that the proposed algorithm outperforms the CNN and MCS in the reference sample reduction rate and classification performance. We have also demonstrated that the minimal consistent set of Dasarathy's algorithm is generally not truly minimal, and his claim of monotonically reducing of the consistent set size is not always true. We feel that the TS-based condensing method significantly reduces the size of the reference set without losing the classification accuracy. Therefore the TS-based method should be considered as a promising tool in the NN classifier design.

References

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

[2] Hart P E. The condensed nearest neighbor rule. *IEEE Trans. Information Theory*, May 1968, IT-14(3): 515–516.

[3] Gates G W. The reduced nearest neighbor rule. *IEEE Trans. Information Theory*, May 1972, IT-18(3): 431–433.

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

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

[6] Devijver P A, Kittler J. On the edited nearest neighbor rule. In *Proc. 5th Int. Conf. Pattern Recognition*, Miami, Florida, 1980, pp.72–80.

[7] Dasarathy B V. Minimal consistent set (MCS) identification for optimal nearest neighbor decision systems design. *IEEE Trans. Syst. Man Cybern.*, March 1994, 24(3): 511–517.
[8] Kuncheva L I. Fitness functions in editing k-NN reference set by genetic algorithms. *Pattern Recognition*, 1997, 30(6): 1041–1049.
[9] Glover F, Laguna M. Tabu Search, in Modern Heuristic Techniques for Combinatorial Problems. Reeves R C (ed.), Berkshire: McGraw-Hill, 1995, pp.70–150.
[10] Fukunaga K. Introduction to Statistical Pattern Recognition. Second Edition, New York: Academic Press, 1990.
[11] Hamamoto Y, Uchimura S, Tomita S. A bootstrap technique for nearest neighbor classifier design. *IEEE Trans. Pattern Analysis and Machine Intelligence*, Jan. 1997, 19(1): 73–79.
[12] Van Ness J. On the dominance of non-parametric Bayes rule discriminant algorithms in high dimensions. *Pattern Recognition*, 1980, 12(3): 355–368.
[13] Fukunaga K, Hummels D M. Bias on nearest neighbor error estimates. *IEEE Trans. Pattern Analysis and Machine Intelligence*, Jan. 1987, 9(1): 103–112.
[14] Fukunaga K, Hummels D M. Bayes error estimation using Parzen and k-NN procedures. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 1987, 9(5): 634–643.

ZHANG Hongbin received the B.S. degree in automation in 1968, and the 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 Department of Information Science of Kyoto University, Japan. From 1993 to 1994 he was a visiting scholar of Rensselaer Polytechnic Institute, USA. Since 1993, he has been a professor of Computer Institute, Beijing Polytechnic University, China. His current research interests include pattern recognition, computer vision, neural networks and image processing.

SUN Guangyu received the B.S. degree in geology from Peking University in 1992 and the M.S. degree from Computer Institute, Beijing Polytechnic University in 1999. His current research interests include pattern recognition and computer vision.