近邻法参考样本集的最优选择

张鸿宾,孙广煜
(北京工业大学计算机学院,北京 100044)

摘 要: 为克服近邻分类法需要大量计算和存储的缺点,本文利用 Tabu 搜索来求解满足一定错误率条件的最小参考样本集. 当错误率阈值设为 0 时,可以得到原训练集的一致子集. 当错误率阈值设为适当的非零值时,可以较好地克服近邻估计的偏置. 通过在 Tabu 搜索中引入适当的激活(aspiration)条件,避免了在可行和不可行解区间无意义的来回搜索,加快了收敛的速度. 实验结果表明,本文算法在压缩比和分类性能上都优于经典的算法. 本文还证明了 Dasarathy 的算法[6]得到的最小一致子集(Minimal Consistent Set: MCS)不是最小的,其 MCS 也不一定是单调减的.

关键词: 近邻分类; Tabu 搜索; 参考样本集
中图分类号: TP391 **文献标识码**: A **文章编号**: 0372-2112 (2000) 11-0016-06

Optimal Selection of Reference Subset for Nearest Neighbor Classification

ZHANG Hong-bin, SUN Guang-yu
(*Computer Institute of Beijing Polytechnic University, Beijing* 100044, *China*)

Abstract: A new approach is presented for tackling the problem of requiring a lot of memory and computational demand for nearest neighbor classification. Tabu search is used to find the optimal condensed reference set, which has minimal sample size and satisfies a certain error rate. 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 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 nonzero appropriate values. The experimental results also illustrate that the MCS (minimal consistent set) of Dasarathy's algorithm is not minimal, and his candidate consistent set is not always ensured to be reducing monotonically. A counter example is also given to confirm this claim.

Key words: nearest neighbor classification; tabu search; reference subset

1 引言

近邻法(Nearest Neighbor Classification: NN)是基于样本间距离的一种分类方法,至今仍是模式识别中最重要的非参数方法之一. 传统近邻法的主要问题是,当训练集的样本很多时,需要大量的计算和存储. 另一个问题是很难确定 k-近邻法中 k 的合适值,特别是在有噪声的情况下. 为解决近邻法的上述缺陷,多年来很多学者作了大量的工作. Dasarathy 在文献[1]中对以往的研究作了很好的总结.

为了减少近邻法的计算和存储量,可以采取两种途径. 一种途径是寻找有效的数据结构及查找最近邻样本的搜索算法. 另一种途径是减少训练集的样本数,这可以通过从原训练集中挑选一些有代表性的参考样本或者利用原样本集产生数目较少的新样本来实现. 本文主要讨论挑选有代表性参考样本的方法. 这类算法要求有较大的样本压缩比,同时又不降低近邻法的分类性能.

试图减少训练集样本数的研究开始于 Hart 的压缩近邻法(Condensed Nearest Neighbor Rule: CNN)[2]. 该算法得到的压缩集与原样本集是一致的,所谓一致是指原样本集中的样本用压缩集进行近邻法分类时可保证完全分类正确. CNN 算法的缺点是对原样本集样本的排列顺序敏感,而且压缩集中含有较多的冗余样本. 针对 CNN 算法中样本只能添入压缩集而不能删除的缺陷, Gates 提出了精简近邻法(ReducedNearest

收稿日期:1999-10-25;修回日期:2000-03-26
基金项目:国家自然科学基金(No. 69675007);北京自然科学基金(No. 4972008)

© 1994-2010 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

Neighbor Rule:RNN)[3],将得到的压缩集进行精简,得到 CNN 压缩集的一个子集,它与原样本集保持一致性. 与 CNN 和 RNN 方法不同,Chang 试图从原样本集中产生一些新的参考样本,而不是从原来的样本中进行挑选[4]. 这实际是统计学上的一种 bootstrap 方法. Devijver 和 Kittler 提出的多重剪辑算法 MULTIEDIT 的错误率是渐进贝叶斯最优的[5]. 但 MULTIEDIT 算法一般只适用于训练样本数相当多的情形. 另外,它剪辑掉的多是类边界附近的点,仍有大量冗余的样本.

最近,Dasarathy 在总结以往工作的基础上,提出了最小一致子集(Minimal Consistent Subset:MCS)的概念和算法[6]. 该算法是以最近异类子集(Nearest Unlike Neighbor Subset:NUNS)和最近异类样本(Nearest Unlike Neighbor:NUN)的概念为基础的. NUNS 实际上隐含规定了类间的边界. 原样本集的任一样本都可正确分类的充分条件是 MCS 中存在与此样本的距离小于其 NUN 距离的同类样本. Dasarathy 引入了一种投票机制,顺次把最有代表性的样本放入 MCS 中. 然后重新投票直到 MCS 中的样本不能再减少时为止. 从本文后面的实验中将会看到,Dasarathy 算法得到的压缩集比前面几种方法的要小,但不是如他和一些引用者所说的那样是最小的[6,7]. 另外,本文将给出一个反例说明,文献[6]的算法也不能保证候选一致子集的基数如文[6]所说的那样是单调减的. 下面将 Dasarathy 的算法称为 MCS 算法. MCS 是指算法还是指最小一致子集可以从上下文中看出.

综上所述,在近邻法参考样本集的选择问题上,仍存在一些问题值得进一步研究. 本文把近邻法参考样本集的选择作为一个最优化问题来处理,利用 Tabu 搜索求解满足一定错误率要求的最小参考集. 下面一节描述 Tabu 搜索及其在参考集选择上的应用. 第三、四节是试验数据及实验结果,从参考集的压缩比和分类性能上与 CNN 和 MCS 算法进行比较.

2　Tabu 搜索在近邻法参考样本集选择中的应用

2.1　近邻法参考样本集的最优选择

近邻法参考样本集的最优选择可以表述为如下的最优化问题:

对于一个 c 类的模式分类问题,记 $\Omega = \{\omega_1, \omega_2, \cdots, \omega_c\}$ 为所有类别标号的集合. 原样本集为 $X = \{x_1, x_2, \cdots, x_N\}$,其中每一样本均属于类标号集 Ω 中的一个类别. 令 $P(X)$ 为 X 的幂集,子集 $S \in P(X)$, $Card(S)$ 为 S 中包含的样本数. $e(S)$ 为用 S 中的样本对 X 进行分类时的错误率. 允许的分类错误率阈值为 t. 对训练集进行最优选择的目标是:

找一 S^*,使得

$$Card(S^*) = \min_{\substack{S \in P(X) \\ e(S) \le t}} Card(S) \qquad (1)$$

2.2　Tabu 搜索

Tabu 搜索是 Glover 提出的一种启发式寻优的方法[8]. 自提出以来,Tabu 搜索已在许多问题上取得了优于其它方法的结果,引起了人们越来越大的关注. Tabu 搜索的核心是允许次优的移动并记录搜索过程的履历,以此对搜索过程加以控制,增强搜索的广泛性和集中性. 由于 Tabu 搜索的灵活性,使它很容易和其他方法及特定问题的具体知识结合起来,组成面向具体对象的实用算法. Tabu 搜索的一般框架见[8]. 针对参考样本集的选择问题,Tabu 搜索的具体实施如下节所述.

2.3　Tabu 搜索在参考样本集选择中的应用

参考样本集可以用一个 0/1 的位串来表示. 0 或 1 表示 X 中的某个样本在或不在参考集中. 令 S_{curr}, S_{next}, S_{best} 分别为当前、下一步和最优的参考子集, $\mathbb{L}$ 为一有固定长度的先入先出的 Tabu 表. 求解参考样本集最优选择的 Tabu 搜索算法如下.

(1) 初始化:找一初始解 S_{curr},令 $S_{best} = X$(所有样本的集合), $\mathbb{L} = \Phi$.

(2) 修改 Tabu 表:将 S_{curr} 插入先入先出表 $\mathbb{L}$ 的尾部.

(3) 修改最优解:若 $e(S_{curr}) \le t$,且 $Card(S_{curr}) < Card(S_{best})$,或者 $Card(S_{curr}) = Card(S_{best})$ 且 $e(S_{curr}) < e(S_{best})$,则令 $S_{best} = S_{curr}$.

(4) 在 S_{curr} 的邻域中搜索最优的 S_{next}:

若 $e(S_{curr}) \le t$,即当前解满足错误率条件时,寻找同时满足条件(i) $S_{next} \subset S_{curr}$,(ii) $Card(S_{next}) = Card(S_{curr}) - 1$ 和(iii) $S_{next} \notin \mathbb{L}$ 的最优 S_{next},即寻找从 S_{curr} 中去掉一个样本所能得到的未被禁止的最优解.

若 $e(S_{curr}) > t$,即当前解不满足错误率条件时,寻找同时满足条件(i) $S_{curr} \subset S_{next}$, (ii) $Card(S_{next}) = Card(S_{curr}) + 1$ 和(iii) $S_{next} \notin \mathbb{L}$ 的最优 S_{next},即寻找向 S_{curr} 中添入一个样本所能得到的未被禁止的最优解.

添入和删除样本的最优条件将在下面详细说明.

(5) 令 $S_{curr} = S_{next}$,转 2.

算法的终止条件是执行规定的循环步数或(和)算法在一定步数内未能提高解的质量.

上述算法在添入或删除样本时考虑了候选参考样本集以下三方面的性质:

(1) 添入(或删除)前后错误率的变化情况.

(2) 添入前分类错误的样本在添入后的分类变化情况.

(3) 原训练集中的样本与添入(或删除)后得到的参考样本集之间的距离. 该距离可以用原样本集中每一个样本与参考集中的最近同类样本距离的和来度量.

添入样本时按下面的规则寻找最优的 S_{next}:

规则 1　查找可获得最小错误率的 S_{next},如果 $e(S_{next}) < e(S_{curr})$,则此 S_{next} 为 S_{curr} 邻域中的最优解. 如果多个 S_{next} 同时得到最小错误率,则选取与原样本集距离最小的(集合间距离的定义见上面的 3). 这样做的目的是选取与原样本聚类中心更近的样本,即类内部的样本.

规则 2　若获得的最小错误率与 S_{curr} 相比没有降低,则考虑添入后可使原来分类错误的样本中至少有一个变为分类正确的样本. 在这些样本中选择最小错误率或最小距离的. 如果得到的子集均被禁止(在 Tabu 表中),则激活它们中分类性能最好的(激活的含义是指当满足某种条件时取消对禁止解的禁止). 规则 2 的目的是为了防止过多冗余样本的添入. 如果仅按规则 1 进行,有可能会不断添入冗余的样本,它们的添入不会造成分类性能的下降,但对分类也没有多少帮助. 采用激活条件可以防止无意义的样本互换,提高算法的效率. 这一点将在后面的实验结果中进一步分析.

© 1994-2010 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

删除样本时的情况较为简单,可按与添入样本时的规则1相类似的方法来寻找最优的剔除样本.

算法的初始参考样本集可以是空集,或随机挑选的若干样本.如果用原来的全部样本作为初始集,则一般需要更多的迭代步数.

3 实验数据集

为了检验 Tabu 算法的压缩和分类性能以及和其他算法进行比较,用7个常用而又性质各异的数据集进行了两类实验.第一类实验将错误率阈值定为0,这样 Tabu 搜索将得到原样本集的一致子集,然后将 Tabu 搜索得到的一致子集与其它算法的结果进行比较.第二类实验将错误率设为0和其他非0值,使用独立的检验样本集对算法得到的参考子集的分类性能进行分析比较.实验中采用欧氏距离作为样本间的距离度量,使用最近邻法(1-NN)进行分类.所用的7个数据集如下:

(1) Iris 数据集(IRIS)

Iris 数据集含有三类各 50 个 4 维的样本,是研究人员广泛使用的一个经典数据集.

(2) I-I 数据集(I-I)

Fukunaga 和 Hamamoto 等在文献[9,10]中使用的数据集.它由两类相互独立的 n 维正态分布模型 $N(\mu_i, \sum_i)$ 产生,$i=1,2$.两类的均值和协方差矩阵分别为:$\mu_1=[0,\cdots,0]^T$,$\mu_2=[\mu,0,\cdots,0]^T$,$\sum_1=\sum_2=I_n$,I_n 是 $n\times n$ 的单位矩阵.这一数据集的贝叶斯错误率由 μ 值唯一确定,而且不随维数 n 的变化而改变.实验中取 $\mu=2.56$,这将生成贝叶斯错误率为 10% 的数据集.

(3) 环状分布数据集(RING)

一个两维的两类数据集.样本均匀分布在由三个圆组成的区域内(图1).最里面的圆和最外面的环形区域内分布的是一类样本(图中的灰色区域),中间的环形分布着另一类的样本(图中白色区域).三个圆的半径分别为 r_1,r_2 和 r_3.

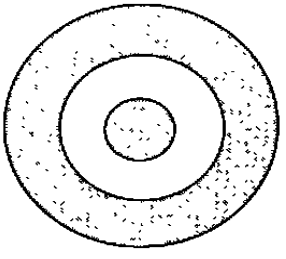


图1 环状分布数据集

(4) 对角分布数据集(DIAGONAL)

一个两维的两类数据集.每类样本均由两个正态分布模型产生,其概率密度函数分别为:

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

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

其中 $\mu_{11}=[0,0]^T$,$\mu_{12}=[\mu,\mu]^T$,$\mu_{21}=[\mu,0]^T$,$\mu_{22}=[0,\mu]^T$,分别位于正方形的不同对角上.这一样本集的贝叶斯错误率也由 μ 唯一确定.实验中取 $\mu=3.5$.

(5) 间隔分布数据集(INTERVAL)

Fukunaga 在文献[9]中使用的数据集.每一类都由两个正态分布组成,其概率密度函数分别为:

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

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

其中 $\mu_{11}=[0,\cdots,0]^T$,$\mu_{12}=[6.58,0,\cdots,0]^T$,$\mu_{21}=[3.29,0,\cdots,0]^T$,$\mu_{12}=[9.87,0,\cdots,0]^T$。这一数据集的贝叶期错误率为 7.5%。

(6) Ness 数据集(NESS)

Ness 在文献[11]中使用的数据集,由两个相互独立的 n 维正态分布模型 $N(\mu,\sum_i)$ 产生.其参数为:

$\mu_1=[0,\cdots,0]^T\mu_2=[\Delta/2,0,\cdots,0,\Delta/2]^T$,$\sum_1=I_n$,$\sum_2=\begin{bmatrix} I_{n/2} & O \\ O & \frac{1}{2}I_{n/2}\end{bmatrix}$.

其中 Δ 为两类间的马氏距离.这一样本集的贝叶斯错误率由 Δ 和 n 共同决定.

(7) 各维均值递减的正态分布数据集(VMD)

也是由两个相互独立的 n 维正态分布 $N(\mu_i,\sum_i)$ 产生,$i=1,2$.两类正态分布的参数分别为:

$\mu_1=[0,\cdots,0]^T$,$\mu_2=[\mu,\frac{\mu}{2},\frac{\mu}{3},\cdots,\frac{\mu}{n}]^T$,$\sum_1=\sum_2=I_n$.

表1列出了7个数据集的维数、类别数、样本数和参数值.其中 RING 训练集中两类的样本数分别为 120 和 60,检验集中两类样本的数量比也为 2∶1.其余数据集的训练和检验样本数相等.

表1 实验中所用的数据集

样本集	维数	类别数	样本数量(训练集/检验集)	所用参数值
IRIS	4	3	150	
I-I	6	2	300/3000	$n=6$, $=2.56$
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	$n=10$, $\mu=3.0$

4 实验结果与分析

4.1 用 Tabu 搜索求解原训练集的一致子集

将错误率阈值设为0,利用 Tabu 搜索可以寻找最优(即样本数最少)的一致子集.对于上述7个数据集,Tabu 搜索的结果见表2.为了对比,同时实现了 Hart 的 CNN 和 Dasarathy 的 MCS 算法,结果也一并列在了表中.对于 CNN 算法,表中列出了10次运行中的最好及平均结果.对于 Tabu 搜索的初始解,实验中分别采用了空集和随机选取的方法以便进行比较.当用随机挑选的若干样本作为 Tabu 搜索的初始解时,每个数据集都重复进行了10次实验,表中列出了10次中的最好、最差和平均结果,并在平均结果后给出了标准差.对于从空集开始的算法只做一次搜索,因为按本文的算法,在算法参数确定后,给定初始解将唯一确定一条搜索路径.实验中 Tabu 表的

© 1994-2010 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

长度设定为等于训练集的样本数 N,终止条件为迭代 2*N* 次或 *N* 次迭代中解的质量未得到改善.

从实验结果可以看出,在保证压缩集和原训练集一致的前提下,CNN 的压缩效果最差,而且如前面提到的那样,CNN 算法对样本的顺序敏感,不同的顺序会产生差别很大的结果. 与 CNN 算法相比,除 DIAGONAL 数据集外,MCS 的压缩性能均有较大的提高. 而 Tabu 搜索对所有的数据集都得到了最好的结果,所得一致子集的样本数比 CNN 和 MCS 均有相当程度的减少. 虽然 Tabu 搜索对初始解有一定的依赖性,但即使在最坏的情况下,Tabu 搜索的结果一般仍然优于 MCS(对 IRIS 数据集,二者持平). Tabu 搜索的运行时间虽然比 CNN 和 MCS 要长,但参考样本的大幅度减少,将有效地提高后继的实际分类阶段时的效率. 在准备参考集的阶段花费一些时间也是值得的.

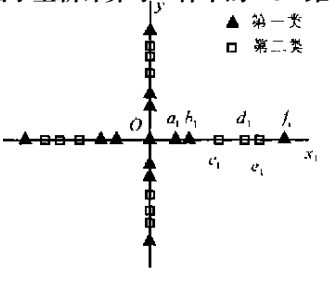
表 2　各算法得到的一致子集的大小比较

数据集	原始集合	CNN		MCS	Tabu(初始解为空集)	Tabu(随机挑选 *m* 个样本作初始解)			
		最好	平均			*m*	最好	最差	平均(±s.d.)
IRIS	150	18	19.8	15	15	15	12	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)

根据以上实验结果,下面进一步分析 MCS 算法以及 Tabu 搜索中的激活条件.

(1) Dasarathy 在文献[6]中曾讲,尽管没有严格的数学证明,但他猜测 MCS 算法得到的一致子集是最小的. Kuncheva 在引用文[6]时也认为 MCS 是最小的[7]. 本文的实验结果则表明,MCS 算法得到的一致子集不一定是最小的. 例如,在 Iris 数据集中,MCS 算法得到的一致子集含有 15 个参考样本(见文[6]和本文表 2),而本文的 Tabu 搜索算法得到了只含 12 个样本的一致子集.

(2) 在 MCS 算法中,Dasarathy 引入了候选一致集合,该集合包含了上一步的一致子集和添入后不产生新的不一致的样本,并且断言以后迭代时一致子集的样本数是单调减的. 在实现 MCS 算法时发现,Dasarathy 的这一措施并非完全有效,一致子集中的样本数有时可能会增加,并不能保证是单调减的. 由于重新计算每一样本的 NUN 距离和重新投票后,可能会使



样本坐标

样本标记	类别	*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

图 2　使 MCS 的一致子集样本数增加的数据集

样本的得票数排列顺序发生变化(同上一步比较),而这一结果对算法得到更小的一致子集是否有利是很难预料的. 下面给出一个反例,说明 Dasarathy 的断言不一定正确.

如图 2 所示,两类样本分布在 *x* 轴和 *y* 轴上. 部分样本点的坐标见图 2,其余样本点由这些点绕原点 *O* 分别旋转 90°、180 和 270 得到. 为叙述方便,将与原点对称的样本组成一个集合,分别称为 *O* 和 *A*, *B*, *C*, *D*, *E*, *F*. 根据 Dasarathy 的算法,第一步迭代时 *O*～*F* 集合中各点的得票数及 NUN 距离见表 3,同一集合中各点的得票数是相同的.

表 3

样本点	*o*	a_1	b_1	c_1	d_1	e_1	f_1
得票数	1	3	3	2	3	2	1
NUN 距离	1.9	0.9	0.6	0.6	0.8	0.5	0.5

于是,第一步时得到的候选一致子集为 $A \cup D \cup F$,其大小为 12. 第二步重新计算 NUN 距离及投票后,由于 *A* 中各点都会对 *O* 投票,因此 *O* 的票数将是 5. 其它各点的得票数及相应的 NUN 距离见表 4.

表 4

样本点	*o*	a_1	b_1	c_1	d_1	e_1	f_1
得票数	5	3	3	2	3	3	1
NUN 距离	2.4	1.4	1.1	0.9	0.8	0.5	0.8

按照得票数及加入后不引起新的不一致的条件,*O* 将被首先选入新的一致子集,这样第二步迭代得到的一致子集为 $\{O\} \cup A \cup D \cup F$,所含样本数目为 13,比前一步一致子集的样本数目增加了.

由于上述原因,表 2 中列出的 MCS 的大小不是 MCS 算法最后得到的子集大小,而是迭代中曾经达到的最小子集的大小.

(3) 在有约束的最优化问题中,当搜索进行到可行解区和不可行解区的边界附近时,搜索常在边界两侧进行无意义的来回游走,降低了算法收敛的速度. 本文在 Tabu 搜索中引入了激活条件来避免一些无意义的样本互换以提高算法的效率. 下面通过一个例子分析引入激活条件的必要性.

设 *a* 是训练集中的一个样本,只有当该样本本身在压缩集中时才能保证它的正确分类. 而 $B = \{b_1, b_2, \cdots, b_k\}$ 是某类样本的一个聚类,压缩集中含有 *B* 的任一样本都能使 *B* 中所有的样本正确分类. 假定 Tabu 搜索算法搜索到子集 *S*,*S* 满足错误率条件,且 $a, b_1 \in S$. 下一步 Tabu 搜索将试图将子集中的一个样本去掉. 经过比较,去掉 *a* 将获得最小错误率,因此 Tabu 搜索下一步将得到子集 $S - \{a\}$(符号 - 和 + 表示删除或添加一个样本). 假定这一子集的分类性能已不满足错误率的要求,Tabu 搜索将添加一个样本. 若没有激活条件,由于 *S* 已被禁止(*S* 在 Tabu 表中),可能会加入一个多余的样本 b_2. 此时错误率仍然不能满足阈值条件,需要继续加入样本,于是算法将加入 *a*,得到满足阈值条件的子集 $S + \{b_2\}$,之后会去掉 b_1,得到 $S + \{b_2\} - \{b_1\}$(此子集未被禁止),于是搜索过程将是:

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

© 1994-2010 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

这几步搜索所做的实际上是将 S 中的样本 $b1$ 替换为 $b2$,而这样的替换还将继续下去. 因为 $S+\{b_2\}-\{b_1\}$ 满足错误率条件,下一步又会去掉样本 a,算法将陷入如下无意义的样本互换:

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

显然,这些替换毫无意义. 尤其是在 Tabu 表长较短时,算法会陷入这样的替换中而无法跳出.

在上面的问题中,理想的搜索路径应该是在尝试去掉 a 失败后,重新回到 S,再尝试去掉其它的样本. 通过引入激活条件可以达到这样的目的. 例如,在搜索到子集 $S-\{a\}$ 后,由于添入 $b_i(i=1,\cdots,k)$ 不能使分类性能提高,也不能把原来错分的样本变为分类正确的样本,算法将激活 S(见第二节的算法描述),并从 S 开始进行另外途径的搜索,由于此时 $S-\{a\}$ 被禁止,算法将在 S 中去掉另外的样本,从而避免了无意义的样本互换.

4.2 Tabu 搜索获得的压缩子集的分类性能

上一小节把错误率阈值设为 0,利用 Tabu 搜索得到了原训练集的一致子集. 在实际工作中,当训练集的样本数不是足够多时,一致子集的分类性能未必是最好的. 文献[12,13]指出,当样本数有限时,一般的近邻估计有较大的偏差. 他们推荐在密度估计中引入一个阈值 t,即当 $\hat{p}(x|\omega_k)>\hat{p}(x|\omega_j)+t, j=1,\cdots,m, j\neq k$ 才将 x 分到 k 类. 但确定 t 的适当值是一件困难的工作. 这一小节通过为压缩集设置不同的错误率阈值,进一步分析所得到的压缩子集的分类性能.

在下面的实验中,式(1)的错误率阈值分别设置为 $t=0.00$、$t=0.05$ 和 $t=0.10$. 用独立的训练集和检验集来获得压缩集和检验压缩集的性能. 对不同的数据集和错误率阈值,重复 Tabu 搜索 5 次,其中 1 次以空集作为初始解,另外 4 次以随机选择的一定数量的样本作为初始解. 利用从训练集中得到的压缩集样本对检验集进行 1-NN 法分类,实验结果见表 5. 为了比较,利用整个训练集的 1-NN 法以及 CNN 和 MCS 算法的结果也列在了表 5. 表中的 IRIS1 和 IRIS2 数据集分别为对原 IRIS 集的不同划分,其训练集/检验集的样本数分别为 30/120 和 75/75. 表中的错误率用百分比表示.

表 5 各算法所得压缩子集的分类性能

算法 \ 数据集			IRIS1	IRIS2	FI	RING	DIAGONAL	INTERVAL	NESS	VMD
原训练集大小			30	75	300	180	100	300	300	200
1NN	错误率(%)		4.17	4.00	17.73	9.44	8.75	13.07	9.10	6.75
CNN	压缩子集大小(平均)		7.1	12.4	97.4	51.0	16.2	104.1	72.6	34.4
	错误率(平均)(%)		5.12	8.43	20.54	12.17	11.94	16.71	14.24	10.48
MCS	压缩子集大小		6	9	74	43	13	89	46	23
	错误率(%)		6.67	12.00	21.27	11.73	10.60	18.73	15.50	9.45
Tabu (t=0.0)	压缩子集大小(平均)		4.0	8.2	63.0	30.4	7.4	67.8	33.4	7.3
	错误率(%)	最好	3.33	6.67	19.23	10.17	12.90	15.57	12.10	4.90
		最差	5.00	12.00	21.03	12.93	6.55	18.30	14.97	7.55
		平均	3.67	8.27	20.23	11.19	10.28	17.04	13.80	6.36
Tabu (t=0.05)	压缩子集大小(平均)		4.0	3.0	11.4	13.6	4.0	12.2	2.0	2.0
	错误率(%)	最好	3.33	4.00	12.47	10.07	6.50	9.73	7.70	5.45
		最差	5.00	4.00	16.60	14.20	8.70	13.60	7.70	5.45
		平均	4.00	4.00	14.63	12.75	7.55	11.23	7.70	5.45
Tabu (t=0.10)	压缩子集大小(平均)		3.0	3.0	2.6	9.8	4.0	4.0	2.0	2.0
	错误率(%)	最好	10.00	4.00	11.33	13.40	5.50	9.33	7.70	5.45
		最差	12.50	4.00	12.67	19.97	7.50	12.40	7.70	5.45
		平均	11.33	4.00	11.95	16.36	6.57	10.54	7.70	5.45

从实验结果可以看出,当错误率阈值为 0 时,Tabu 搜索得到的一致子集比 CNN、MCS 的要小,而分类性能则比 CNN、MCS 有相当程度的提高. 当错误率阈值条件设置为 0.05 时,对于大多数数据集(除 IRIS1 外),Tabu 搜索得到的子集样本数与 $t=0.0$ 时相比有大幅度的减少. 在此阈值条件下,对 NESS 和 VMD 数据集找到了理想的子集,其分类性能接近于贝叶斯错误率. 而 $t=0.10$ 时,DIAGONAL,FI 和 INTERVAL 数据集也得到了理想子集,其样本数和前面相比有显著减少,而分类性能则有进一步的提高.

对 IRIS1 数据集,$t=0.10$ 时得出的压缩子集与 $t=0.05$ 时相比分类性能则有较大的退化. 数据集 RING 也是一个例外,其参考集需要包含较多的样本. 通过实验可以看出,数据集的类型不同,其适合的错误率阈值也不同. 对于一般的数据集,引入一个小的错误率阈值,一般都能减少参考集的样本数并提高分类的正确率. 本文数据集的合适阈值大致为(见表 6):

表 6

数据集	IRIS	FI	RING	DIAGONAL	INTERVAL	NESS	VMD
合适阈值	0.05	0.10	0.00	0.10	0.10	0.05	0.05

© 1994-2010 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

通过分析 Tabu 算法得到的压缩集样本点的分布可以发现,Tabu 搜索所选择的参考样本点的分布较为合理.图 3 给出了 DIAGONAL 训练集样本点以及各算法所得参考样本点的分布.其中(*a*)为原训练集的样本点分布,(*b*) - (*f*)分别标出了 CNN、MCS 和 Tabu(t = 0.00,0.05,0.10)挑选的压缩集样本点的分布情况.

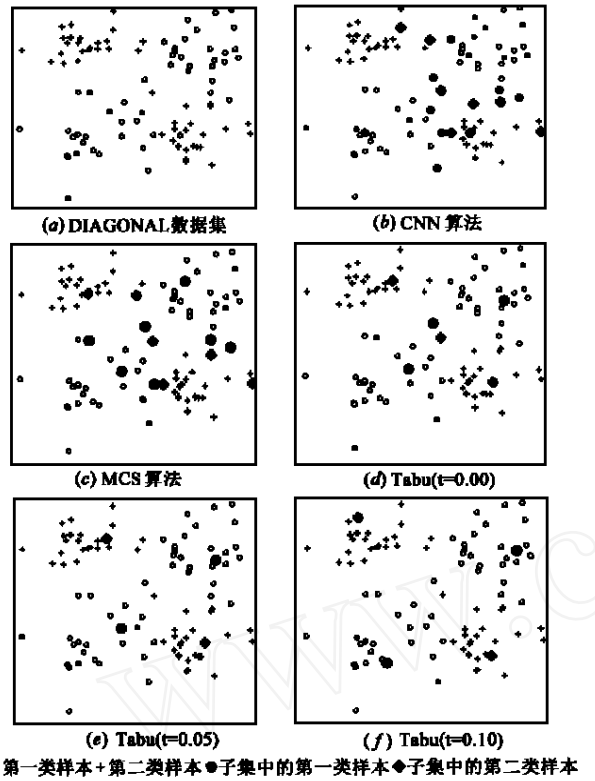


图 3　DIAGONAL 数据集及各算法所得压缩子集的样本分布

5　小结

本文利用 Tabu 搜索求解近邻法参考集的最优选择问题.与 CNN 和 MCS 算法相比,本文算法获得了更小的一致子集,并且提高了分类性能.在合适的错误率阈值条件下,压缩子集的样本数和分类性能一般还会有进一步的改善.文中证明了 Dasarathy 的 MCS 算法得到的一致子集不是最小的,一致子集的样本数也不总是单调减的.本文工作表明,在求解近邻法的压缩子集时,Tabu 搜索是值得考虑的一种方法.

参考文献:

[1] B. V. Dasarathy. Nearest neighbor (NN) norms: NN pattern classification techniques [A]. Los Alamitos, CA: IEEE Computer Society Press, 1991.

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

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

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

[5] P. A. Devijver, and J. Kittler. On the edited nearest neighbor rule [A]. Proc. 5th ICPR, Miami, Florida, 1980: 72 - 80.

[6] B. V. Dasarathy. Minimal consistent set (MCS) identification for optimal nearest neighbor decision systems design [J]. IEEE. Trans Syst. man Cybern., March, 1994, 24(3): 511 - 517.

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

[8] F. Glover, M. Laguna. Tabu Search, in Modern Heuristic Techniques for Combinatorial Problems [M]. R. C. Reeves, ed., McGraw - Hill, Berkshire: 70 - 150.

[9] K. Fukunaga. Introduction to Statistical Pattern Recognition [M]. Second ed. Academic Press, 1990.

[10] Y. Hamamoto, S. Uchimura, and S. Tomita. A bootstrap technique for nearest neighbor classifier design [J]. IEEE Trans. on Pattern Analysis and Machine Intelligence, Jan, 1997, 19(1): 73 - 79.

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

[12] K. Fukunaga, D. M. Hummel. Bias on nearest neighbor error estimates [J]. IEEE Trans. Pattern Analysis and Machine Intelligence, 1987, 9(1): 103 - 112.

[13] K. Fukunaga, D. M. Hummel. Bayes error estimation using Parzen and k-NN procedures [J]. IEEE Trans. PAMI, 1987, 9(5): 634 - 643.

作者简介:



张鸿宾　教授,博士生导师.1968 年清华大学自动控制系毕业.1981 年清华大学模式识别与智能控制专业研究生毕业.1986～1989 年日本京都大学、1993～1994 年美国 RPI 高级访问学者.目前主要从事模式识别、图象处理和计算机视觉、人工神经网络等方面的工作.

孙广煜　1992 年毕业于北京大学地质系,1999 年北京工业大学研究生毕业,主要从事计算机应用方面的工作.

© 1994-2010 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net