一种改进的结合 K 近邻法的 SVM 分类算法

殷小舟
(北京林业大学信息学院,北京 100083)

摘 要 在对支持向量机在超平面附近容易对测试样本造成错分进行研究的基础上,改进了将支持向量机分类和 k 近邻分类相结合的方法,形成了一种新的分类器。在分类阶段计算待识别样本和最优分类超平面的距离,如果距离差大于给定阈值可直接应用支持向量机分类,否则用最佳距离 k 近邻分类。数值实验表明,使用支持向量机结合最近邻分类的分类器分类比单独使用支持向量机分类具有更高的分类准确率。

关键词 支持向量机 k 近邻法 泛化错误 最佳距离度量
中图法分类号: TP301.6 **文献标识码**: A **文章编号**: 1006-8961(2009)11-2299-05

An Ameliorated SVM Classifying Algorithm Combined with kNN

YIN Xiao-zhou
(*School of Information Science & Technology*, *Beijing Forestry University*, *Beijing* 100083)

Abstract An ameliorated algorithm that combined support vector machine (SVM) with k nearest neighbour (kNN) is presented and it comes into being as a new classifier, based on the research that SVM classifies some tested samples in error nearby the optimal super-plane. In the class phase, the algorithm computes the distance from the tested sample to the optimal super-plane of SVM in the feature space. If the distance is greater than the given threshold, the tested sample will be classified on SVM, otherwise, the kNN algorithm will be used based on the best distance measurement. The numerical experiments show that the mixed algorithm improve the accuracy compared to the sole SVM.

Keywords SVM(support vector machine), kNN(k nearest neighbour), generalization error, best distance measurement

1 引 言

支持向量机(SVM)是 Vapnik 等人在统计学习理论的基础上提出的一种通用机器学习方法[1-2]。对于二类线性可分问题,SVM 通过在已知样本正确分类的约束条件下,通过使分类间隔最大化,从而确定最优分类面。因而可以被看做是结构风险最小化(SRM)原则的近似实施[3]。通过将参数空间中的输入数据映射到高维特征空间的方法,SVM 很容易被推广到非线性分类问题,并通过构造核函数的方法,在原空间中计算特征空间中的内积,而不进行真正的变换,甚至无需知道确切的变换形式。与其他分类器相比,SVM 有着如下优势:它能够在样本有限时获得较好的分类能力;对输入空间的维数、训练集大小、样本的概率分布不太敏感。目前 SVM 已经成为机器学习领域新的研究热点,并被应用于文本识别、手写体识别、图像识别、入侵检测等方面[4]。

然而,SVM 也存在一些问题:由于 SVM 的理论基础是不适当问题的正则化理论和非线性规划计算方法,核函数及其参数的确定依赖于给定的问题;对大规模问题训练时间较长;对于复杂问题分类精度不是很高。目前,对 SVM 性能的研究大都集中在平均错分率上界的定性分析[5-6]。文献[7]提出了一种通过改造核函数来提高 SVM 性能的方法。文献[8]在做了一些假设后通过简化的方法定量分析了

收稿日期:2009-05-04;**改回日期**:2009-08-10
第一作者简介:殷小舟(1988~),男。北京林业大学计算机科学与技术专业本科生。主要研究方向为艺术设计。E-mail:yxzdtc@163.com

1 维特征空间中,样本均匀分布下的 SVM 的错分率,并提出了一种增量训练方法减少 SVM 的错分率。另外,对一些不能纳入 SLT 框架的机器学习中的直推方法[9]的研究,也被用来改善 SVM 的性能。文献[10]、文献[11]提出了一种结合 kNN(k nearest neighbour)提高 SVM 分类精度的方法,并给出了 KSVM(kNN & support vector machine)、MPKSVM(multi-representative point kNN & support vector machine)两个算法。kNN 并不遵从 SRM 原则,它的 VC 维(VC dimension)为无穷大,但它的错分率在一倍到两倍的贝叶斯错分率之间,且算法思想简单。kNN 与 SVM 这两个不同类型的分类器各有优、缺点,将它们结合起来改善 SVM 性能的思路颇具启发性。

本文首先简要地分析并比较了 SVM,kNN 的特点,在分析了 SVM 对某些测试样本错分的真正原因后,指出了 KSVM,MPKSVM 算法的不足。在这两个算法的基础上,提出了结合最佳距离度量 k 近邻法提高 SVM 分类精度的方法,并给出了改进的算法,命名为 BDKSVM(Best distance kNN & support vector machine)。

2 SVM,k 近邻法简介

2.1 SVM

给定一组样本$(\boldsymbol{x}_i, y_i)$,$i=1,\cdots,n$, $\boldsymbol{x}\in\mathbf{R}^d$,$y\in\{+1,-1\}$,假设 ϕ 是输入空间 $\boldsymbol{S}$ 到特征空间 $\boldsymbol{F}$ 的一个映射,核函数 K 对应 $\boldsymbol{F}$ 中向量内积运算,即

$$K(\boldsymbol{x},\boldsymbol{x}')=\langle\phi(\boldsymbol{x}),\phi(\boldsymbol{x}')\rangle,$$

则非线性可分下的最优分类问题可转化为一个约束条件下的二次优化问题:

$$\max\{Q(\alpha)\}=\max\left\{\sum_{i=1}^{n}\alpha_i-\frac{1}{2}\sum_{i,j=1}^{n}\alpha_i\alpha_j y_i y_i K(x_i,x_j)\right\} \tag{1}$$

式中,$K(x_i,x_j)$为核函数,α_i 为与每个样本对应的 Lagrange 乘子。

约束条件为

$$\sum_{i=1}^{n}\alpha_i y_i=0 \qquad \alpha_i\geqslant 0, i=1,\cdots,n \tag{2}$$

设 α_i^* 为最优解,根据 Kuhn-Tucker 条件容易得到 α_i^* 中只有很小一部分不为零,对应的样本就是支持向量(SV)。于是得到的最佳分类函数是:

$$f(\boldsymbol{x})=\operatorname{sgn}\left\{\sum_{i\in SV}\alpha_i^* y_i K(\boldsymbol{x}_i,\boldsymbol{x})+b^*\right\} \tag{3}$$

此即为支持向量机,sgn 为符号函数。

关于 SVM 的泛化能力,有多种表示形式。文献[12]给出了一种简洁的形式:

$$E(P_e)\leqslant E[N]/(N'-1) \tag{4}$$

式中,N 为支持向量数量,N'为训练样本总数,P_e 为错分概率,$E()$为数学期望。

即对于测试样本分类错误率的期望的上界是训练样本中平均得支持向量占总训练样本数的比率。问题是无法找到普适的方法求出这个比率的精确值。

SVM 的性能受到核函数形式及其参数、问题本身的复杂程度、分类面附近的噪声点、输入向量参数选择、样本数量、样本分布等因素的影响,其中前 3 个因素起着关键性作用。

2.2 kNN

kNN 是 1NN(1 nearest neighbour)的推广,其算法思想比较简单:将所有 N 个训练样本都作为代表点,计算测试样本 $\boldsymbol{x}$ 到所有训练样本点的距离,找出与 $\boldsymbol{x}$ 最近的训练样本中 k 个最近邻,看这 k 个近邻中的多数属于哪一类,就把 $\boldsymbol{x}$ 分到哪一类。当 $N\to\infty$ 时,kNN 的错分率 P_e 满足下面不等式[12]:

$$\hat{P}\leqslant P_e\leqslant\hat{P}[2-\hat{P}c/(c-1)] \tag{5}$$

式中,$\hat{P}$ 是贝叶斯分类器的错分率,c 是类别数。

正是 kNN 的这种优良性质,使它成为分类问题中的重要方法。

当样本数 N 有限时,用下列距离定义[13]可以保证与无限样本 kNN 错分率的均方误差最小:

$$D(\boldsymbol{x},\boldsymbol{x}^L)=|\nabla P(\omega_1|\boldsymbol{x})^{\mathrm{T}}(\boldsymbol{x}-\boldsymbol{x}^L)| \tag{6}$$

式中,$\nabla P(\omega_1|\boldsymbol{x})^{\mathrm{T}}$ 表示 $P(\omega_1|\boldsymbol{x})$的梯度转置,$\boldsymbol{x}^L$ 表示 $\boldsymbol{x}$ 的局部近邻区域中的样本。

kNN 适用的前提是样本数目较大、局部近邻区域类条件概率相同[13]。对于 k 值的取法,一方面较大的 k 值可以减少错分率,另一方面要求 k 个近邻都靠近测试样本。kNN 缺点是计算量和存储量都较大。

3 KSVM,MPKSVM 算法及其不足

3.1 KSVM 算法

文献[10]、文献[11]首先提出了下面的定理,并给出了证明:

定理 1:SVM 可看做每类只有一个代表点的 1NN 分类器。”

在这个定理的基础上提出了 KSVM 算法,其主要算法思想为:由于 SVM 对每类只取一个代表点,有时该点不能很好地代表该类,这时将其与 kNN 相结合,因为 kNN 将每类所有支持向量作为代表点,从而使分类器具有更高的分类准确率。

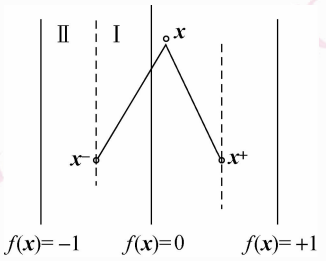

图 1　超平面与样本代表点

Fig. 1　Supper-plane and representative point

图 1 中,$f(\boldsymbol{x})=0$ 为 SVM 最优超平面,$\phi(\boldsymbol{x})^+ = \frac{1}{C}\sum_{i\in SV^+}\alpha_i^*\phi(\boldsymbol{x}_i)$ 与 $\phi(\boldsymbol{x})^- = \frac{1}{C}\sum_{i\in SV^-}\alpha_i^*\phi(\boldsymbol{x}_i)$ 分别是两类支持向量($\boldsymbol{SV}^+,\boldsymbol{SV}^-$)的代表点,其中,$C = \sum_{y_i=1}\alpha_i^* = \sum_{y_i=-1}\alpha_i^*$。具体如下:

(1)计算测试样本 $\boldsymbol{x}$ 到分类面的距离 $|f(\boldsymbol{x})|$。

(2)若 $|f(\boldsymbol{x})|\geqslant\varepsilon$,则用 $f(\boldsymbol{x})$ 判断 $\boldsymbol{x}$ 的类别,ε 为设定的 1 到 0 之间的常数,一般为 0.8。

(3)若 $|f(\boldsymbol{x})|<\varepsilon$,则用所有的支持向量作为代表点,用 kNN 法对 $\boldsymbol{x}$ 进行分类。采用下列距离公式:

$$d(\boldsymbol{x},\boldsymbol{x}_i) = \|\phi(\boldsymbol{x})-\phi(\boldsymbol{x}_i)\|^2 = K(\boldsymbol{x},\boldsymbol{x}) - 2K(\boldsymbol{x},\boldsymbol{x}_i) + K(\boldsymbol{x}_i,\boldsymbol{x}_i) \quad (7)$$

3.2　MPKSVM 算法

MPKSVM 算法与 KSVM 算法基本类似,都是基于上面的定理,其主要算法思想为:

(1)将训练样本集划分为 c 对正反样本子集,在每对子集上训练 SVM,由训练结果分别形成一对代表点。

(2)划分训练样本集的方法是:用聚类算法将正、反样本分成 c 对聚类。聚类准则是使得每对正、反样本聚类中心距离最短。各对聚类中心可表示为:

$$\phi(x_k)^+ = \frac{1}{C_k}\sum_{i\in SV_K^+}\alpha_{ki}^*\phi(x_{ki}) \quad (8)$$

$$\phi(x_k)^- = \frac{1}{C_k}\sum_{i\in SV_K^-}\alpha_{ki}^*\phi(x_{ki}) \qquad k=1,2,\cdots,c \quad (9)$$

式中,$\boldsymbol{SV}_k^+,\boldsymbol{SV}_k^-$ 分别表示第 k 对聚类的正副支持向量,$C_k = \sum_{y_{ki}=1}\alpha_i^* = \sum_{y_{ki}=-1}\alpha_i^*$。

(3)用 $2c$ 个聚类中心作为训练样本集的代表点,用 kNN 对测试样本 $\boldsymbol{x}$ 进行分类。

3.3　KSVM,MPKSVM 算法的不足

MPKSVM 算法与 KSVM 算法非常相似,都是在 SVM 最优分类面附近找多个代表点,用 kNN 法对 SVM 容易错分的测试样本进行分类,其出发点都是结合 SVM,kNN 各自的优势进一步提高分类器的分类精度。

在对 SVM,kNN 各自特点分析的基础上,可发现 KSVM,MPKSVM 算法的几点不足:

首先,kNN 较好的分类能力是在样本趋于无穷多时得到的,样本较少时 kNN 错分的风险很大[13]。无论是 KSVM 算法还是 MPKSVM 算法,所用的代表点都只占有限训练样本集中的较少一部分。这是这两个算法最主要的不足。

其次,kNN 用到的一个基本假设是类条件概率在局部近邻区域相同。当这一条件不满足时,kNN 错分的风险较大。而 SVM 分类器本身对样本的密度分布是不敏感的。

基于最优分类面的分类方法的核心思想是:对分类起主导作用的是分类面附近的样本[3],通过使类间隔最大化,找到支持向量,作为类的代表点,"支撑"着最优分类面,从而减少指示函数集的 VC 维。文献[10]、文献[11]在证明其定理时,用支持向量构造了一对代表点 $\phi(\boldsymbol{x})^+,\phi(\boldsymbol{x})^-$,在这基础上得到结论"SVM 可看做每类只有一个代表点的 1NN 分类器",这与支持向量"支持"的本意不太一致。这是 KSVM,MPKSVM 算法不能提高分类精度的根本原因所在。

4　证明 KSVM 算法不能提高分类精度

对于二类线性可分问题,设训练样本和测试样本各自独立产生,且服从同一个未知形式的固定的密度分布 Π。图中,L 为最优分类面,L_1 和 L_2 上的点是支持向量,L_0 为针对总体的期望分类面,即 L_0 能够将总体正确地分为两类,它是存在的,只是求不出相应的超平面方程。⊙、◎都为测试样本。

当测试样本落在 L 和 L_0 之间的区域时,SVM 都错分;当测试样本落在其他区域时,SVM 正确分类。

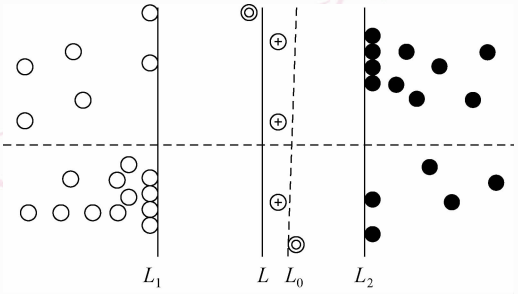

图 2 KSVM 算法产生错分情况

Fig. 2 The case when KSVM algorithm makes mistakes

若分布 Π 的形式能够知道的话,便能够定量地计算出该 SVM 的期望错分率。当测试样本落在 L 和 L_0 之间的区域时,用近邻法(1NN,kNN)来判定时也同样都是错判。另外,当测试样本落在上、下部,如◎,用 kNN 也是错判(12 个支持向量作为代表点,找 6 个近邻)。但 SVM 能够对◎正确判定。

因此,KSVM 算法的错分率不低于 SVM 的错分率。

5 BDKSVM 算法

BDKSVM 算法的出发点也是结合 SVM,kNN 各自的优势进一步提高分类器的分类精度。图 2 中可以看出,SVM 错分的测试样本都落在最佳分类面 L 附近。最坏情况当 L_0 与 L_1 或 L_2 重合时,落在 L_1 与 L_2 之间的测试样本有一半被错分。其原因是只用少数样本(支持向量)作为类的代表点,而没用到训练样本集中的多数样本的信息。因此,可对 KSVM 算法作如下改进:

BDKSVM 算法:

Input:给定训练样本集 $\boldsymbol{T}_r = \{x_i \mid i = 1,2,\cdots,m\}$、测试样本集 $\boldsymbol{T}_e = \{x_{m+j} \mid j = 1,2,\cdots,n\}$

Output:测试样本 x_{m+j} 的类别,+1 表示正例类,-1 表示反例类

Procedure:

$\boldsymbol{T}_r \xrightarrow{\text{训练 SVM}} \boldsymbol{SV} = \{x_i' \mid x_i' \in \boldsymbol{T}_r, i = 1,2,\cdots,L\}$

/* $\boldsymbol{SV}$ 为训练后得到的 L 个支持向量的集合 */

for($j = 1$; n; $j++$)

$\{g(x_{m+j}) = \sum_{i \in SV} \alpha_i^* y_i K(x_i * x_{m+j}) + b^*$;

if ($|g(x_{m+j})| \geqslant 1$)

/* 若 x_{m+j} 到最佳分类面的距离 $\geqslant 1$ */

printf(sgn($g(x_{m+j})$)); /* 输出 x_{m+j} 的类别 */

else

{for($i = 1$; m; $i++$)计算 $\|x_{m+j} - x_i\|$;

/* 用近邻区域距离定义计算 x_{m+j} 到每个训练样本 x_i 的距离 */

找出与 x_{m+j} 距离最短的(βk)个近邻,其中 μ 个为正例:

if ($\mu > (\beta k - \mu)$ printf(+1); else printf(-1);

}

}

当测试样本落在 L_1 左侧或 L_2 右侧时用 SVM 分类;当测试样本落在 L_1 与 L_2 之间时用 kNN 分类,并将全体测试样本都看做类代表点。考虑到训练样本集本身是有限的,采用最佳距离度量 k 近邻法找出(βk)个近邻(β 为大于 1 的常数),近邻区域距离采用下面的定义[13]:

$$D(\boldsymbol{x},\boldsymbol{x}^*) = |\nabla_1^{\mathrm{T}}(\boldsymbol{x} - \boldsymbol{x}^*)| \tag{10}$$

式中,$\boldsymbol{x}^*$ 为 $\boldsymbol{x}$ 近邻区域的样本;

$\nabla_1 = M_+(\boldsymbol{x}) - M(\boldsymbol{x})$; $M_+(\boldsymbol{x}) = \frac{1}{N_+}\sum_{N_+}(\boldsymbol{x}^* - \boldsymbol{x})$;

$M(\boldsymbol{x}) = \frac{1}{N}\sum_{N}(\boldsymbol{x}^* - \boldsymbol{x})$;$N_+$ 和 N 分别为 $\boldsymbol{x}$ 近邻区域中的正样本数、样本数。

由于落在 L_1 与 L_2 之间的测试样本只占测试样本的少数,因此,所增加的计算时间是不大的。

6 实验及数据分析

实验的目的有两方面:比较 SVM,KSVM,MPKSVM,BDKSVM 的分类精度和确定 BDKSVM 算法中 β 的取值与分类精度的关系。

实验数据为双螺旋曲线。双螺旋曲线问题是模式识别中经典的二类分类问题。如图 3 所示,螺旋线中的粗点作为正例,细点作为反例。螺旋线圈数的多少、点之间的间隔代表了分类问题的复杂程度,圈数越多、点之间的间隔越大,问题越复杂。实验中采用 Bootstrap 方法,对于 n 个样本的数据集随机抽

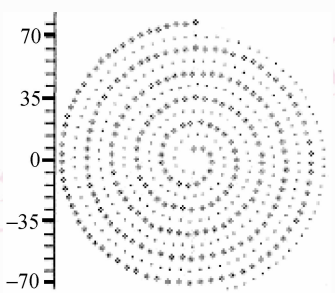

图 3 双螺旋曲线

Fig. 3 A two-spirals

取 n 次,将得到的样本去掉重复作为训练样本,余下的样本作为测试样本。重复上述操作 10 次,取平均结果,核函数采用高斯核:

$$K(\boldsymbol{x},\boldsymbol{y}) = \exp\{-g\|\boldsymbol{x}-\boldsymbol{y}\|^2\} \qquad (11)$$

实验中,$g = 0.05$,实验环境为 Petium Ⅳ 2.6 GH, 512 MB RAM, Windows XP。

6.1　测试分类精度

样本为双螺旋曲线上的 2 000 个点,螺旋曲线圈数取了 4 种,BDKSVM 算法中 β 的取值为 2。

表 1 中数据表明,KSVM,MPKSVM 算法不能提高分类精度;BDKSVM 算法能够明显提高分类精度,圈数越多、问题越复杂,提高的幅度越大。其原因在于落在最佳分类面附近的测试样本比率增加了,对于这些很难分的测试样本,以全部训练样本作为代表点,用最佳距离度量 K 近邻法提高了分类精度。表中还可以看出,所增加的计算时间在可接受的范围之内。

表 1　分类精度对比

Tab. 1　The comparison of classification accuracies

算法	3 圈		4 圈		10 圈		11 圈	
	CPU 时间(s)	分类精度(%)	CPU 时间(s)	分类精度(%)	CPU 时间(s)	分类精度(%)	CPU 时间(s)	分类精度(%)
SVM	1.83	92.2	1.90	90.3	1.97	81.7	1.92	76.2
KSVM	1.95	91.9	1.98	89.9	2.29	80.2	2.30	75.7
MPKSVM	1.97	92.0	2.04	90.4	2.23	80.8	2.32	75.8
BDKSVM	2.05	94.8	2.12	92.1	2.49	86.7	2.58	84.4

6.2　测试 β 值与分类精度的关系

样本为双螺旋曲线上的 5 000 个点,螺旋曲线圈数为 5。

表 2 中数据表明 CPU 时间随着 β 值增大而上升,$\beta=5$ 时 CPU 时间是 $\beta=1$ 时的两倍多;分类精度 $\beta=2$ 左右分类精度较高。分析原因当 $\beta=1$ 相当于用普通距离定义求了 k 个近邻;由于最佳距离量度是在测试样本的局部近邻区域作线性近似得到的,当 β 值较大时近似误差较大,造成分类有很大的精度下降。综上所述,β 值取 2 比较合适。

表 2　BDKSVM 算法中 β 值与分类精度的关系

Tab. 2　The relationship between the value of β and the classification accuracy on the BDKSVM

β	1.0	1.5	2.0	2.5	3.0	3.5	4.0
CPU 时间(s)	14.7	16.0	16.2	19.0	19.4	21.4	24.6
分类精度(%)	85.2	87.7	90.8	90.9	87.3	85.4	83.8

7　结　论

本文以结合 SVM,kNN 各自的优势提高分类精度为目的,在分析、对比了 SVM,kNN 的特点后,对 KSVM,MPKSVM 两个算法的不足进行了研究,并提出了改进的 BDKSVM 算法。实验结果表明,BDKSVM 算法能够真正地提高分类精度,并且所增加的计算时间在可接受的范围之内。今后进一步的研究是在特征空间里利用核函数计算近邻区域最佳距离量度的可行性,并设法减少额外增加的计算时间。

参考文献(References)

1　Boser B, Guyon I, Vapnik V. A training algorithm for optimal margin classifiers[A]. In: Proceedings of the Fifth Annual Workshop on Computational Learning Theory [C], Pittsburgh, Pennsylvania, USA, 1992:144-152.

2　Cortes C, Vapnik V. Support vector networks[J]. Machine Learning, 1995, **20**(3):273-297.

3　Cristianini N, Shawe-Taylor J. An Introduction to Support Vector Machines[M]. Cambridge: Cambridge University Press, 2000.

4　Srinivas Mukkamala, Guadalupe Janoski, Andrew Sung. Intrusion detection using neural networks and support vector machines[A]. In: Proceedings of IEEE International Joint Conference on Neural Networks[C], Honolulu, Hawaii, USA, 2002:1702-1707.

5　Vapnik V, Chapelle O. Bounds on error expectation for support vector machines[J]. Neural Computation, 2000, **12**(9): 2013-2036.

6　Dietrich R, Opper M, Sompolinsky H. Statistical mechanics of support vector networks [J]. Physical Review Letters, 1999, **82**(14): 2975-2978.

7　Amari S, Wu S. Improving support vector machine classifiers by modifying kernel functions[J]. Neural Networks, 1999, **12**(6): 783-789.

8　Feng Jian-feng. Peter Williams. The generalization error of the symmetric and scaled support vector machines[J]. Neural Networks, 1999, **12**(5):1255 - 1260.

9　Vladimir N. Vapnik. The Nature of Statistical Learning Theory[M]. New York: Springer-Verlag, 2000.

10　Li Rong, Ye Shi-wei, Shi Zhong-zhi. SVM-kNN classifier—a new method of improving the accuracy of SVM classifier [J]. Acta Electronica Sinica, 2002, **30** (5): 745-748. [李蓉, 叶世伟, 史忠植. SVM-kNN 分类器:一种提高 SVM 分类精度的新方法[J]. 电子学报, 2002, **30**(5): 745-748.]

11　Li Rong, Ye Shi-wei, Shi Zhong-zhi. A effective classified algorithm of support vector machine with multi-Representative points based on nearest neighbor principle [A]. In: Proceedings of International Conference on Info-tech and Info-net[C], Beijing, China, 2001: 113-119.

12　Bian Zhao-qi, Zhang Xue-gong. Pattern Recognition[M]. Beijing: Tsinghua University Press. [边肇祺, 张学工. 模式识别[M]. 北京:清华大学出版社, 2000.]

13　Chin K K. Support Vector Machines applied to Speech Pattern Classification[D]. Cambridge, UK: Cambridge University, 1998.