软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2012,23(12):3059−3073 [doi: 10.3724/SP.J.1001.2012.04209]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

最大间隔对数向量机*

胡文军[1,2+], 王士同[1], 王 娟[2], 颜七笙[1,3]

[1](江南大学 数字媒体学院,江苏 无锡 214122)

[2](湖州师范学院 信息与工程学院,浙江 湖州 313000)

[3](东华理工大学 数学与信息科学学院,江西 抚州 344000)

Maximum Margin Logistic Vector Machine

HU Wen-Jun[1,2+], WANG Shi-Tong[1], WANG Juan[2], YAN Qi-Sheng[1,3]

[1](School of Digital Media, Jiangnan University, Wuxi 214122, China)

[2](School of Information and Engineering, Huzhou Teachers College, Huzhou 313000, China)

[3](School of Mathematics and Information Science, East China Institute of Technology, Fuzhou 344000, China)

+ Corresponding author: E-mail: hoowenjun@yahoo.com.cn

Hu WJ, Wang ST, Wang J, Yan QS. Maximum margin logistic vector machine. *Journal of Software*, 2012, 23(12):3059−3073 (in Chinese). http://www.jos.org.cn/1000-9825/4209.htm

Abstract: The L2-kernel classifier does not consider explicitly its classification margin when approximating the difference of densities (DoD) with the integrated squared error (ISE) criterion of probability densities, which is disadvantageous for improving the performance of classifiers to a certain extent. Its weights can simply be obtained by solving the corresponding QP problem which results in the comparatively slow training speed and is impractical especially for large datasets. With the aim of overcoming the above drawbacks, a new classification method is proposed in this paper, called the maximum margin logistic vector machine (MMLVM), which maximizes the DoD-based classification margin and finds the corresponding weight vector by solving a logistic optimization problem in gradient descent way. The theoretical analysis is provided in the globally optimal weights, the generalization error bound, and in the computational complexity of MMLVM. Experimental results on the artificial, UCI, PIE and USPS data sets demonstrate the effectiveness of the proposed approach in overcoming the drawbacks as above.

Key words: classification; maximum margin; logistic vector machine; kernel classifier; difference of densities

摘 要: 通过 ISE 准则逼近真实密度差的 L2-核分类器没有显式地考虑到分类间隔,在一定程度上不利于提高分类器精度;同时,权向量的求解最终转化为一个二次规划问题,导致 L2-核分类器训练速度较慢,特别是对于较大样本.基于这两个问题,利用样本间的密度差构造了分类间隔并最大化此间隔,而此问题最终转化为一个对数优化问题,故称其为最大间隔对数向量机(maximum margin logistic vector machine,简称 MMLVM),进而利用梯度下降法求解最优权.同时,分别从权的全局最优性、一般化误差界及算法复杂度这 3 方面进行了理论分析.最后,人工和 UCI,PIE 及 USPS 数据集的实验结果表明,算法理论正确,解决了上述两个问题并获得了较好的效果.

* 基金项目: 国家自然科学基金(61170122, 61272210); 江苏省自然科学基金(BK2011003, BK201141); 江苏省“333 专家”工程(BRA2011142); 江西省自然科学基金(20114BAB201022); 2011 年、2012 年江苏省普通高校研究生科研创新计划

收稿时间: 2011-03-27; 修改时间: 2012-02-15; 定稿时间: 2012-03-19

关键词: 分类;最大间隔;对数向量机;核分类器;密度差
中图法分类号: TP181 文献标识码: A

分类是模式识别和机器学习中的重要研究内容之一,广泛应用于现实生活中,如药物检测、门禁系统、医疗诊断等[1−3].常见的分类器有:解决二分问题的支持向量机(support vector machine,简称 SVM)[4]和 v-SVC[5];解决异常检测的支持向量数据描述(support vector data description,简称 SVDD)[2]和小球体大间隔(small sphere and large margin,简称 SSLM)算法[3];以及基于密度估计的 L2-核分类器[6]和核密度估计(kernel density estimate,简称 KDE)分类器[7]等.而评价分类器的优劣往往考虑 3 个方面:

(1) 测试精度.一般地,用于分类的平均间隔(如 SVM 的分类间隔 $2/\|\boldsymbol{w}\|$,这里,$\boldsymbol{w}$ 为超平面的法向量)越大,精度越高,但最大化平均间隔容易忽视样本的局部分布;

(2) 获取分类器(或决策函数)的时间,体现在训练速度上.因上述方法最终对应二次规划(quadratic programming,简称 QP)问题,因此在训练较大样本时速度较慢;

(3) 决策未知样本所属类别的速度.这常用稀疏性评价,如 SVM 和 SVDD 的支持向量数、L2-核分类器和 KDE 中非零权的个数,其越少稀疏性越好,决策速度越快.

实际上,只有基于某种测度才能给出相应的分类间隔.所以,采用不同测度将得到不同类型的分类器,如:SVM[4]和 SVDD[2]/SSLM[3]分别是以点到超平面的欧式距离和点到球心的欧式距离进行鉴别,它们以欧式距离为测度;而 L2-核分类器[6]、KDE[7]和贝叶斯分类器[8]是以概率密度大小进行鉴别,它们以概率密度为测度.给定相应测度的分类间隔后,一般通过最大化分类间隔来求解相应的优化模型.我们注意到,最大化分类间隔大多考虑欧式测度,如 SVM 通过 $\min(\boldsymbol{w}^T\boldsymbol{w}/2)$实现分类间隔 $2/\|\boldsymbol{w}\|$的最大化.文献[9]从概率密度测度出发定义了密度差异(density contrast),并通过最大化此差异来增强概率密度模型的鉴别能力.但是,基于概率密度测度的 L2-核分类器仅将两类间的密度差(difference of densities,简称 DoD)看成一种新的分布,并利用累积平方误差(integrated squared error,简称 ISE)准则最优逼近两者密度差分布进而实现分类.可见,在一定程度上,L2-核分类器并没有考虑到最大分类间隔问题,造成了如提出者所得到的分类精度与 SVM 相比相当或没有优势之结论[6].实际上,构造不同类型的分类间隔往往可以解决或改善机器学习中的一些问题或算法[10,11].

鉴于上述分析,本文利用密度差构建分类间隔,并最大化全体样本的总分类间隔,而此模型最终转化为一个对数优化问题进行求解,并在较大程度上提高了训练速度,称该方法为最大间隔对数向量机(maximum margin logistic vector machine,简称 MMLVM).

本文第 1 节介绍核分类器的一般形式.第 2 节介绍分类间隔的构造和 MMLVM 算法及其实现.第 3 节从理论上分析 MMLVM 算法的一些特性.第 4 节给出实验结果与分析.第 5 节总结全文.

1 L2-核分类器

定义 1. 设样本空间 $X\subset\Re^d$,输入样本$\boldsymbol{\chi}=\boldsymbol{\chi}^+\cup\boldsymbol{\chi}^-$,其中$\boldsymbol{\chi}^+$和$\boldsymbol{\chi}^-$分别是从样本空间 X 中采样获得的正负类样本集,不妨设$\boldsymbol{\chi}^+=\{\boldsymbol{x}_i|\boldsymbol{x}_i\in X,1\leqslant i\leqslant N^+\}$,$\boldsymbol{\chi}^-=\{\boldsymbol{x}_i|\boldsymbol{x}_i\in X,N^++1\leqslant i\leqslant N\}$,其中,$\boldsymbol{x}_i$ 是列向量,类标签 $\boldsymbol{y}=(y_1,\ldots,y_N)^T$.当 $\boldsymbol{x}_i\in\boldsymbol{\chi}^+$时,$y_i=+1$;当 $\boldsymbol{x}_i\in\boldsymbol{\chi}^-$时,$y_i=-1$.

一般地,样本空间 X 很难做到线性可分,为了提高线性分类器的灵活性,常使用所谓的核技巧:假设存在一个映射 ϕ将样本空间 X 映射到一个尽可能高维的特征空间 Φ中,即 $\phi:\boldsymbol{x}\in X\mapsto\phi(\boldsymbol{x})\in\Phi$,并且通过一个正定核函数 $k:\Re^d\times\Re^d\in\Re$诱导 Φ空间中的内积形式,即 $\phi(\boldsymbol{x}_i)^T\phi(\boldsymbol{x}_j)=k(\boldsymbol{x}_i,\boldsymbol{x}_j)$.通过核技巧,许多核分类器,如不带偏移项的 SVM 等,一般都具有如下形式:

$$f(\boldsymbol{x})=\sum_{i=1}^{N}\alpha_i y_i k(\boldsymbol{x}_i,\boldsymbol{x})=\sum_{i=1}^{N^+}\alpha_i k(\boldsymbol{x}_i,\boldsymbol{x})-\sum_{i=N^++1}^{N}\alpha_i k(\boldsymbol{x}_i,\boldsymbol{x}) \tag{1}$$

其中,$\boldsymbol{\alpha}=(\alpha_1,\ldots,\alpha_N)^T\geqslant 0$ 是参数向量.显然,公式(1)与 KDE 具有相同的形式.

假定$\boldsymbol{\chi}^+$和$\boldsymbol{\chi}^-$样本所在空间的真实概率密度为 $p_+(\boldsymbol{x})$和 $p_-(\boldsymbol{x})$,根据决策理论[6]可知,以概率密度为测度的分类

器具有如下形式:

$$f(\boldsymbol{x})=p_+(\boldsymbol{x})-\gamma p_-(\boldsymbol{x}) \tag{2}$$

若 $f(\boldsymbol{x})\geqslant 0$,则 $\boldsymbol{x}$ 属于+1 类;否则属于–1 类.其中,γ是通过先验概率获得的某一固定常数[6].根据文献[9]可知,$p_+(\boldsymbol{x})-p_-(\boldsymbol{x})$是两类间的密度差(difference of densities,简称 DoD),并利用密度差定义了两类间基于密度的差异(density-based contrast),这种差异恰恰反映了基于密度测度分类器的鉴别能力.

一般地,根据贝叶斯分类器和 Parzen Windows 理论[7,8]可知,正负类样本的类条件概率密度可简单地估计为 $p(\boldsymbol{x}\mid\boldsymbol{\chi}^+)=\frac{1}{N^+}\sum_{i=1}^{N^+}\varphi_h(\boldsymbol{x}_i,\boldsymbol{x})$ 和 $p(\boldsymbol{x}\mid\boldsymbol{\chi}^-)=\frac{1}{N-N^+}\sum_{i=N^++1}^{N}\varphi_h(\boldsymbol{x}_i,\boldsymbol{x})$,其中,$\varphi_h(\bullet,\bullet)$为某个窗函数,$h$ 对应窗宽,所以有

$$\begin{aligned}p(\boldsymbol{x}\mid\boldsymbol{\chi}^+)-p(\boldsymbol{x}\mid\boldsymbol{\chi}^-)&=\frac{1}{N^+}\sum_{i=1}^{N^+}\varphi_h(\boldsymbol{x}_i,\boldsymbol{x})-\frac{1}{N-N^+}\sum_{i=N^++1}^{N}\varphi_h(\boldsymbol{x}_i,\boldsymbol{x})\\&=\frac{1}{N^+}\left(\sum_{i=1}^{N^+}\varphi_h(\boldsymbol{x}_i,\boldsymbol{x})-\frac{N^+}{N-N^+}\sum_{i=N^++1}^{N}\varphi_h(\boldsymbol{x}_i,\boldsymbol{x})\right).\end{aligned}$$

上式 $1/N^+$可以不考虑,故本文令 $d_\gamma(\boldsymbol{x})=p_+(\boldsymbol{x})-\gamma p_-(\boldsymbol{x})$.显然,它反映的也是一种基于密度测度的差异.而$\gamma$可简单地通过 $N^+/(N-N^+)$确定,同时为了简单说明,本文直接称 $d_\gamma(\boldsymbol{x})$为密度差.

根据 KDE(如 Parzen Windows)理论[7,8]可知,利用有限的采样样本(根据定义 1,可直接使用$\boldsymbol{\chi}^+$和$\boldsymbol{\chi}^-$)可以估计 $p_+(\boldsymbol{x})$和 $p_-(\boldsymbol{x})$[6,7]权化形式:

$$\hat{p}_+(\boldsymbol{x};\boldsymbol{\alpha}_+)=\sum_{1\leqslant i\leqslant N^+}\alpha_i k_\sigma(\boldsymbol{x}_i,\boldsymbol{x}) \tag{3}$$

$$\hat{p}_-(\boldsymbol{x};\boldsymbol{\alpha}_-)=\sum_{N^++1\leqslant j\leqslant N}\alpha_j k_\sigma(\boldsymbol{x}_j,\boldsymbol{x}) \tag{4}$$

其中,$k_\sigma(\bullet,\bullet)$是带宽为$\sigma$的高斯核;$\boldsymbol{\alpha}_+=(\alpha_1,...,\alpha_{N^+})^T\geqslant 0$ 且 $\boldsymbol{\alpha}_+^T\mathbf{1}=1$;$\boldsymbol{\alpha}_-=(\alpha_{N^++1},...,\alpha_N)^T\geqslant 0$ 且 $\boldsymbol{\alpha}_-^T\mathbf{1}=1$.这里,$\mathbf{1}$ 是单位列向量.此时,密度差 $d_\gamma(\boldsymbol{x})$的估计为

$$\hat{d}_\gamma(\boldsymbol{x};\boldsymbol{\alpha})=\hat{p}_+(\boldsymbol{x};\boldsymbol{\alpha}_+)-\gamma\hat{p}_-(\boldsymbol{x};\boldsymbol{\alpha}_-) \tag{5}$$

其中,$\boldsymbol{\alpha}=\begin{pmatrix}\boldsymbol{\alpha}_+\\ \boldsymbol{\alpha}_-\end{pmatrix}$.L2-核分类器通过优化问题 $\boldsymbol{\alpha}=\underset{\boldsymbol{\alpha}\in\mathcal{R}^N;\boldsymbol{\alpha}_+^T\mathbf{1}=1;\boldsymbol{\alpha}_-^T\mathbf{1}=1}{\arg\min}\ ISE(\boldsymbol{\alpha})$ 求解得到,$ISE(\boldsymbol{\alpha})=\|\hat{d}_\gamma(\boldsymbol{x};\boldsymbol{\alpha})-d_\gamma(\boldsymbol{x})\|_2^2$.实际上是将问题转化为一个 QP 问题,见文献[6].此时,L2-核分类器可表示为

$$f^*(\boldsymbol{x})=\hat{p}_+(\boldsymbol{x};\boldsymbol{\alpha}_+)-\gamma\hat{p}_-(\boldsymbol{x};\boldsymbol{\alpha}_-)=\sum_{1\leqslant i\leqslant N^+}\alpha_i k_\sigma(\boldsymbol{x}_i,\boldsymbol{x})-\gamma\sum_{N^++1\leqslant j\leqslant N}\alpha_j k_\sigma(\boldsymbol{x}_j,\boldsymbol{x}) \tag{6}$$

显然,通过简单变形,公式(1)和公式(6)具有相同的形式.

2 MMLVM 算法

2.1 Margin构造

L2-核分类器通过 ISE 准则逼近真实密度差,没有显式地考虑到有助于提高分类精度的分类间隔问题,这对提高分类精度是不利的.为此,本文先构造一种分类间隔.根据公式(6)可知,决策未知样本是通过该样本在两类样本上的密度差估计值实现,因此可以定义训练样本 $\boldsymbol{x}_n$ 在训练时的分类间隔为

$$\rho_n=y_n(\hat{p}_+(\boldsymbol{x}_n;\boldsymbol{\alpha}_+)-\gamma\hat{p}_-(\boldsymbol{x}_n;\boldsymbol{\alpha}_-)) \tag{7}$$

显然:

- 当 $\boldsymbol{x}_n$ 正确训练时:
  - 若 $y_n=1$,则 $\hat{p}_+(\boldsymbol{x}_n;\boldsymbol{\alpha}_+)-\gamma\hat{p}_-(\boldsymbol{x}_n;\boldsymbol{\alpha}_-)\geqslant 0$;
  - 若 $y_n=-1$,则 $\hat{p}_+(\boldsymbol{x}_n;\boldsymbol{\alpha}_+)-\gamma\hat{p}_-(\boldsymbol{x}_n;\boldsymbol{\alpha}_-)<0$.

  对于这两种情况$\rho_n\geqslant 0$;
- 当 $\boldsymbol{x}_n$ 错误训练时,$\rho_n<0$.

将公式(7)展开成向量乘积形式,即

$$\rho_n = \begin{pmatrix} \boldsymbol{\alpha}_+ \\ \boldsymbol{\alpha}_- \end{pmatrix}^T z_n = \boldsymbol{\alpha}^T z_n \tag{8}$$

其中,

$$z_n = (y_n k_\sigma(\boldsymbol{x}_1, \boldsymbol{x}_n), ..., y_n k_\sigma(\boldsymbol{x}_{N^+}, \boldsymbol{x}_n), -\gamma y_n k_\sigma(\boldsymbol{x}_{N^++1}, \boldsymbol{x}_n), ..., -\gamma y_n k_\sigma(\boldsymbol{x}_N, \boldsymbol{x}_n))^T \tag{9}$$

注意,公式(9)中的 $z_n \in \mathcal{H}^N$,因此有如下结论:

通过公式(9)训练样本 $\boldsymbol{x}_n$ 被映射到 $\mathcal{H}^N$ 空间中的点 z_n,并且经 $\boldsymbol{\alpha}$ 权化的 z_n 真实地反映该训练样本的分类间隔.

因此,根据上述结论,公式(6)的核分类器学习可以转变为在最大化分类间隔框架下求解 $\boldsymbol{\alpha}$ 权实现.当获得最优权 $\boldsymbol{\alpha}$ 后,给定未知样本 $\boldsymbol{x}$,需要将其映射到 $\mathcal{H}^N$ 中,即

$$z(\boldsymbol{x}) = (k_\sigma(\boldsymbol{x}_1, \boldsymbol{x}), ..., k_\sigma(\boldsymbol{x}_{N^+}, \boldsymbol{x}), -\gamma k_\sigma(\boldsymbol{x}_{N^++1}, \boldsymbol{x}), ..., -\gamma k_\sigma(\boldsymbol{x}_N, \boldsymbol{x}))^T \tag{10}$$

因此,决策函数为

$$f^*(\boldsymbol{x}) = \boldsymbol{\alpha}^T z(\boldsymbol{x}) \tag{11}$$

若 $f^*(\boldsymbol{x}) \geqslant 0$,则 $\boldsymbol{x}$ 属于+1 类;否则属于–1 类.

2.2 Margin构造

累计所有训练样本的分类间隔,并考虑到线性累计可能会出现无边界问题,本文采用对数形式累计样本的分类间隔来估计权 $\boldsymbol{\alpha}$,得到下列优化模型:

$$\left.\begin{aligned} &\min_{\boldsymbol{\alpha}} \sum_{n=1}^{N} \ln\{1+\exp(-\boldsymbol{\alpha}^T z_n)\} \\ &\text{s.t. } \boldsymbol{\alpha}_+^T \mathbf{1} = 1, \boldsymbol{\alpha}_-^T \mathbf{1} = 1 \\ &\quad \boldsymbol{\alpha} \geqslant \mathbf{0} \end{aligned}\right\} \tag{12}$$

显然,公式(12)是一个带约束的凸优化问题.由于上式有两个等式约束和一个非负约束,故不能直接使用梯度下降法进行求解.为此,下面进行等价变换消除上述约束.首先,将公式(12)改写成

$$\left.\begin{aligned} &\min_{\boldsymbol{\alpha}} \sum_{n=1}^{N} \ln\left\{1+\exp\left(-\begin{pmatrix} \boldsymbol{\alpha}_+ \\ \boldsymbol{\alpha}_- \end{pmatrix}^T z_n\right)\right\} \\ &\text{s.t. } \boldsymbol{\alpha}_+^T \mathbf{1} = 1, \boldsymbol{\alpha}_-^T \mathbf{1} = 1 \\ &\quad \boldsymbol{\alpha} \geqslant \mathbf{0} \end{aligned}\right\} \tag{13}$$

令 $\boldsymbol{\alpha}_+ = \boldsymbol{\beta}_+ / \|\boldsymbol{\beta}_+\|_1, \boldsymbol{\alpha}_- = \boldsymbol{\beta}_- / \|\boldsymbol{\beta}_-\|_1$,且 $\boldsymbol{\beta} = \begin{pmatrix} \boldsymbol{\beta}_+ \\ \boldsymbol{\beta}_- \end{pmatrix} \geqslant 0$,显然, $\boldsymbol{\alpha}_+^T \mathbf{1} = 1$ 和 $\boldsymbol{\alpha}_-^T \mathbf{1} = 1$ 必定成立,故公式(13)等价于

$$\left.\begin{aligned} &\min_{\boldsymbol{\beta}} \sum_{n=1}^{N} \ln\left\{1+\exp\left(-\begin{pmatrix} \boldsymbol{\beta}_+ / \|\boldsymbol{\beta}_+\|_1 \\ \boldsymbol{\beta}_- / \|\boldsymbol{\beta}_-\|_1 \end{pmatrix}^T z_n\right)\right\} \\ &\text{s.t. } \boldsymbol{\beta} \geqslant \mathbf{0} \end{aligned}\right\} \tag{14}$$

为了消除上式中的非负不等式约束,令 $\boldsymbol{v} = \begin{pmatrix} \boldsymbol{v}_+ \\ \boldsymbol{v}_- \end{pmatrix}$,且 $\boldsymbol{v}_+ = (v_1, ..., v_{N^+})^T, \boldsymbol{v}_- = (v_{N^++1}, ..., v_N)^T$ 和

$$\beta_j = v_j^2 \tag{15}$$

显然, $\boldsymbol{\beta} \geqslant \mathbf{0}$ 必定成立,且 $\boldsymbol{\beta}_+ = \boldsymbol{v}_+ \otimes \boldsymbol{v}_+, \boldsymbol{\beta}_- = \boldsymbol{v}_- \otimes \boldsymbol{v}_-$, $\|\boldsymbol{\beta}_+\|_1 = \|\boldsymbol{v}_+\|_2^2, \|\boldsymbol{\beta}_-\|_1 = \|\boldsymbol{v}_-\|_2^2$.其中,⊗是 Hadamard 乘积算子.为了简单,记 $\overbrace{\boldsymbol{v} \otimes ... \otimes \boldsymbol{v}}^{p} = \boldsymbol{v}^p$,则公式(14)等价于下列无约束优化问题:

$$\min_{\boldsymbol{v}}\sum_{n=1}^{N}\ln\left\{1+\exp\left(-\begin{pmatrix}\boldsymbol{v}_+^2/\|\boldsymbol{v}_+\|_2^2\\ \boldsymbol{v}_-^2/\|\boldsymbol{v}_-\|_2^2\end{pmatrix}^T\boldsymbol{z}_n\right)\right\} \tag{16}$$

下面将给出梯度下降法求解公式(16)的过程.为此,将公式(16)的目标函数定义为 $F(\boldsymbol{v})$,即

$$F(\boldsymbol{v})=\sum_{n=1}^{N}\ln\left\{1+\exp\left(-\begin{pmatrix}\boldsymbol{v}_+^2/\|\boldsymbol{v}_+\|_2^2\\ \boldsymbol{v}_-^2/\|\boldsymbol{v}_-\|_2^2\end{pmatrix}^T\boldsymbol{z}_n\right)\right\} \tag{17}$$

则 $F(\boldsymbol{v})$的梯度为

$$\nabla F(\boldsymbol{v})=-2\sum_{n=1}^{N}\frac{\exp\left(-\begin{pmatrix}\boldsymbol{v}_+^2/\|\boldsymbol{v}_+\|_2^2\\ \boldsymbol{v}_-^2/\|\boldsymbol{v}_-\|_2^2\end{pmatrix}^T\boldsymbol{z}_n\right)}{1+\exp\left(-\begin{pmatrix}\boldsymbol{v}_+^2/\|\boldsymbol{v}_+\|_2^2\\ \boldsymbol{v}_-^2/\|\boldsymbol{v}_-\|_2^2\end{pmatrix}^T\boldsymbol{z}_n\right)}\boldsymbol{z}_n\otimes\begin{pmatrix}(\|\boldsymbol{v}_+\|_2^2\boldsymbol{v}_+-\boldsymbol{v}_+^3)/\|\boldsymbol{v}_+\|_2^4\\ (\|\boldsymbol{v}_-\|_2^2\boldsymbol{v}_--\boldsymbol{v}_-^3)/\|\boldsymbol{v}_-\|_2^4\end{pmatrix} \tag{18}$$

因此,$\boldsymbol{v}$ 可以通过下列迭代公式进行求解:

$$\boldsymbol{v}^{(t)}\leftarrow\boldsymbol{v}^{(t-1)}-\eta'\nabla F(\boldsymbol{v})=\boldsymbol{v}^{(t-1)}+\eta\sum_{n=1}^{N}\frac{\exp\left(-\begin{pmatrix}\boldsymbol{v}_+^2/\|\boldsymbol{v}_+\|_2^2\\ \boldsymbol{v}_-^2/\|\boldsymbol{v}_-\|_2^2\end{pmatrix}^T\boldsymbol{z}_n\right)}{1+\exp\left(-\begin{pmatrix}\boldsymbol{v}_+^2/\|\boldsymbol{v}_+\|_2^2\\ \boldsymbol{v}_-^2/\|\boldsymbol{v}_-\|_2^2\end{pmatrix}^T\boldsymbol{z}_n\right)}\boldsymbol{z}_n\otimes\begin{pmatrix}(\|\boldsymbol{v}_+\|_2^2\boldsymbol{v}_+-\boldsymbol{v}_+^3)/\|\boldsymbol{v}_+\|_2^4\\ (\|\boldsymbol{v}_-\|_2^2\boldsymbol{v}_--\boldsymbol{v}_-^3)/\|\boldsymbol{v}_-\|_2^4\end{pmatrix} \tag{19}$$

其中,$\eta>0$ 是迭代算法的学习率.注意,公式(16)的目标函数并不是一个凸函数,那么通过公式(19)迭代得到的解能否保证是目标函数公式(17)的全局最小值点,这将在第 3.1 节中加以讨论.

2.3 MMLVM实现

综合第 2.1 节和第 2.2 节的内容,MMLVM 算法归纳如下,其中包含训练和测试两个过程.

MMLVM 实现算法

(1) 训练

输入:数据集$\boldsymbol{\chi}^+=\{\boldsymbol{x}_n|\boldsymbol{x}_n\in\Re^d,1\leqslant n\leqslant N^+\}$,$\boldsymbol{\chi}^-=\{\boldsymbol{x}_n|\boldsymbol{x}_n\in\Re^d,N^++1\leqslant n\leqslant N\}$、核带宽参数$\sigma$、规则参数$\gamma$及迭代停止参数 e;

输出:权向量 $\boldsymbol{\alpha}=\begin{pmatrix}\boldsymbol{\alpha}_+\\ \boldsymbol{\alpha}_-\end{pmatrix}$.

步骤 1:初始化 $\boldsymbol{\alpha}^{(0)}=\begin{pmatrix}\boldsymbol{\alpha}_+\\ \boldsymbol{\alpha}_-\end{pmatrix}=\begin{pmatrix}\mathbf{1}/|I_+|\\ \mathbf{1}/|I_-|\end{pmatrix}$(或$\boldsymbol{\alpha}^{(0)}$随机生成,并对$\boldsymbol{\alpha}_+$和$\boldsymbol{\alpha}_-$分别进行归一处理),$\boldsymbol{\beta}^{(0)}=\boldsymbol{\alpha}^{(0)}$,$t=0$;

步骤 2:根据公式(9)计算 z_n,$1\leqslant n\leqslant N$;根据公式(15)计算 $\boldsymbol{v}^{(0)}$;

步骤 3:Do while

$t=t+1$;

根据迭代公式(19)计算 $\boldsymbol{v}^{(t)}$;

$\beta_i^{(t)}=v_i^{2(t)}(1\leqslant i\leqslant N);\boldsymbol{\alpha}^{(t)}=\begin{pmatrix}\boldsymbol{\beta}_+^t/\|\boldsymbol{\beta}_+^t\|_1\\ \boldsymbol{\beta}_-^t/\|\boldsymbol{\beta}_1^t\|_1\end{pmatrix}$;

Until ($\|\boldsymbol{\alpha}^{(t)}-\boldsymbol{\alpha}^{(t-1)}\|_1<e$)(或采用其他范数);

步骤 4:$\boldsymbol{\alpha}=\boldsymbol{\alpha}^{(t)}$.

(2) 测试

给定未知样本 $\boldsymbol{x}$,通过公式(10)计算 $\boldsymbol{z}(\boldsymbol{x})$,并由公式(11)的非负性输出所属类标签:若其为负,则为–1 类;否则为+1 类.

3 MMLVM 算法分析

3.1 权全局最优性分析

公式(12)和公式(14)虽然是一个凸优化问题,但通过公式(15)得到的无约束优化问题公式(16)的目标函数并不是一个凸函数.那么,通过梯度下降法(即通过公式(19)迭代)得到的解 $\boldsymbol{v}$,可能不是目标函数公式(16)的全局最小值点,而是它的局部最小值点或是鞍点.为了描述清楚,根据公式(14)~公式(16)定义如下函数:

$$F(\boldsymbol{\beta})=\sum_{n=1}^{N}\ln\left\{1+\exp\left(-\begin{pmatrix}\boldsymbol{\beta}_+/\|\boldsymbol{\beta}_+\|_1\\ \boldsymbol{\beta}_-/\|\boldsymbol{\beta}_-\|_1\end{pmatrix}^T\boldsymbol{z}_n\right)\right\}\text{(即公式(14)的目标函数)} \tag{20}$$

$$\boldsymbol{\beta}=\boldsymbol{\beta}(\boldsymbol{v})=\boldsymbol{v}^2\text{(即公式(15)的向量形式)} \tag{21}$$

$$F(\boldsymbol{v})=F(\boldsymbol{\beta}(\boldsymbol{v}))=\sum_{n=1}^{N}\ln\left\{1+\exp\left(-\begin{pmatrix}\boldsymbol{v}_+^2/\|\boldsymbol{v}_+\|_2^2\\ \boldsymbol{v}_-^2/\|\boldsymbol{v}_-\|_2^2\end{pmatrix}^T\boldsymbol{z}_n\right)\right\}\text{(即公式(16)的目标函数)} \tag{22}$$

显然,公式(22)是由公式(20)和公式(21)复合而成.

定理 1. 若 $\boldsymbol{v}^*$是 $F(\boldsymbol{v})$的驻点,则 $\boldsymbol{v}^*$是 $F(\boldsymbol{v})$的鞍点或全局最小值点,不可能是局部最小值点.

证明:因 $\boldsymbol{v}^*$是 $F(\boldsymbol{v})$的驻点,故

$$\frac{\partial F}{\partial \boldsymbol{v}^*}=\left(2v_1^*\frac{\partial F}{\partial \beta_1},...,2v_N^*\frac{\partial F}{\partial \beta_N}\right)^T=\boldsymbol{0} \tag{23}$$

这里,$\dfrac{\partial F}{\partial \boldsymbol{v}^*}$是$\left.\dfrac{\partial F}{\partial \boldsymbol{v}}\right|_{\boldsymbol{v}=\boldsymbol{v}^*}$的简写.不失一般性,假设 $\boldsymbol{v}^*$的前 M 元为 0,其他非 0,则有

$$\left.\frac{\partial F}{\partial \beta_i}\right|_{\beta_i=v_i^{*2}}=0(M+1\leqslant i\leqslant N)\ (\because \text{公式(23)}) \tag{24}$$

并记 $F(\boldsymbol{v})$的 Hessian 矩阵为 $H(\boldsymbol{v})=[h_{ij}]_{N\times N}$,其中,

$$h_{ij}=4v_iv_j\frac{\partial^2 F}{\partial\beta_i\partial\beta_j}+2\frac{\partial F}{\partial\beta_i}\delta_{ij} \tag{25}$$

则

$$H(\boldsymbol{v}^*)=\begin{pmatrix}\boldsymbol{A} & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{B}\otimes\boldsymbol{C}\end{pmatrix} \tag{26}$$

其中,

$$\boldsymbol{A}=\begin{pmatrix}2\dfrac{\partial F}{\partial\beta_1^*} & \cdots & \boldsymbol{0}\\ \vdots & \ddots & \vdots\\ \boldsymbol{0} & \cdots & 2\dfrac{\partial F}{\partial\beta_M^*}\end{pmatrix} \tag{27}$$

$$\boldsymbol{B}=\begin{pmatrix}\dfrac{\partial^2 F}{\partial^2\beta_{M+1}^*} & \cdots & \dfrac{\partial^2 F}{\partial\beta_{M+1}^*\partial\beta_N^*}\\ \vdots & \ddots & \vdots\\ \dfrac{\partial^2 F}{\partial\beta_N^*\partial\beta_{M+1}^*} & \cdots & \dfrac{\partial^2 F}{\partial^2\beta_N^*}\end{pmatrix} \tag{28}$$

$$\boldsymbol{C}=\begin{pmatrix}4v_{M+1}^{*2} & \cdots & 4v_{M+1}^*v_N^*\\ \vdots & \ddots & \vdots\\ 4v_N^*v_{M+1}^* & \cdots & 4v_N^{*2}\end{pmatrix} \tag{29}$$

注意,上式中$\boldsymbol{\beta}^*=\boldsymbol{v}^{*2}$.可知:

$$\boldsymbol{B}>0(\text{正定},\because F(\boldsymbol{\beta})\text{是严格凸函数}) \tag{30}$$

$$\boldsymbol{C}\geqslant 0(\text{半正定}) \tag{31}$$

根据 Schur 乘积理论[12],有

$$\boldsymbol{B}\otimes\boldsymbol{C}\geqslant 0 \tag{32}$$

因此,若有且只有 $\boldsymbol{A}\geqslant 0$ 时,$H(\boldsymbol{v}^*)\geqslant 0$.显然,当 $\boldsymbol{A}<0$ 时,$H(\boldsymbol{v}^*)<0$(负定),则 $\boldsymbol{v}^*$是 $F(\boldsymbol{v})$的鞍点;当 $\boldsymbol{A}\geqslant 0$,$H(\boldsymbol{v}^*)\geqslant 0$,下面证明此时的 $\boldsymbol{v}^*$是 $F(\boldsymbol{v})$的全局而不是局部最小值点.

(反证法)若 $\boldsymbol{v}^*$不是全局最小值点,可假设 $\bar{\boldsymbol{v}}^*$ 是 $F(\boldsymbol{v})$的全局最小值点,即 $F(\bar{\boldsymbol{v}}^*)<F(\boldsymbol{v}^*)$,那么有 $F(\bar{\boldsymbol{\beta}}^*)<F(\boldsymbol{\beta}^*)$ 且 $\bar{\boldsymbol{\beta}}^*\neq\boldsymbol{\beta}^*$ (因为 $\bar{\boldsymbol{\beta}}^*$ 和 $\boldsymbol{\beta}^*$ 是根据 $\bar{\boldsymbol{v}}^*$ 和 $\boldsymbol{v}^*$ 并通过公式(21)计算获得,若 $\bar{\boldsymbol{\beta}}^*=\boldsymbol{\beta}^*$,则 $F(\bar{\beta}^*)=F(\beta^*)\Rightarrow F(\bar{\boldsymbol{v}}^*)=F(\boldsymbol{v}^*)$).同时,根据泰勒理论,存在$\xi\in(0,1)$,使得

$$\begin{aligned}
F(\bar{\boldsymbol{\beta}}^*)&=F(\boldsymbol{\beta}^*)+\left(\frac{\partial F}{\partial\boldsymbol{\beta}^*}\right)^T(\bar{\boldsymbol{\beta}}^*-\boldsymbol{\beta}^*)+\frac{1}{2}(\bar{\boldsymbol{\beta}}^*-\boldsymbol{\beta}^*)^T\left(\frac{\partial^2F}{\partial^2\boldsymbol{\beta}}\right)\Bigg|_{\boldsymbol{\beta}=\boldsymbol{\beta}^*+\xi(\bar{\boldsymbol{\beta}}^*-\boldsymbol{\beta}^*)}(\bar{\boldsymbol{\beta}}^*-\boldsymbol{\beta}^*)\\
&=F(\boldsymbol{\beta}^*)+\left(\frac{\partial F}{\partial\boldsymbol{\beta}^*}\right)^T(\bar{\boldsymbol{\beta}}^*-\boldsymbol{\beta}^*)+\Delta\\
&=F(\boldsymbol{\beta}^*)+\left(\frac{\partial F}{\partial\beta_1^*},...,\frac{\partial F}{\partial\beta_M^*},0,...,0\right)(\bar{\boldsymbol{\beta}}^*-\boldsymbol{\beta}^*)+\Delta\quad(\text{因为公式(24)})\\
&=F(\boldsymbol{\beta}^*)+\left(\frac{\partial F}{\partial\beta_1^*},...,\frac{\partial F}{\partial\beta_M^*},0,...,0\right)\bar{\boldsymbol{\beta}}^*-\left(\frac{\partial F}{\partial\beta_1^*},...,\frac{\partial F}{\partial\beta_M^*},0,...,0\right)\boldsymbol{\beta}^*+\Delta\\
&=F(\boldsymbol{\beta}^*)+\left(\frac{\partial F}{\partial\beta_1^*},...,\frac{\partial F}{\partial\beta_M^*},0,...,0\right)\bar{\boldsymbol{\beta}}^*+\Delta
\end{aligned} \tag{33}$$

最后一步简化是因为 $\boldsymbol{v}^*$的前 M 元为 0,故$\boldsymbol{\beta}^*$的前 M 元也为 0.因 $F(\boldsymbol{\beta})$是一个严格凸函数,故 $\frac{\partial^2F}{\partial^2\boldsymbol{\beta}}>0$,故$\Delta\geqslant 0$;又因为证明条件是 $H(\boldsymbol{v}^*)\geqslant 0$(正定),则 $\boldsymbol{A}\geqslant 0\Rightarrow\frac{\partial F}{\partial\beta_i^*}\geqslant 0(1\leqslant i\leqslant M)$.因此,由公式(33)可得

$$F(\bar{\boldsymbol{\beta}}^*)-F(\boldsymbol{\beta}^*)=\left(\frac{\partial F}{\partial\beta_1^*},...,\frac{\partial F}{\partial\beta_M^*},0,...,0\right)\bar{\boldsymbol{\beta}}^*+\Delta\geqslant 0 \tag{34}$$

显然,这与假设($F(\bar{\boldsymbol{\beta}}^*)<F(\boldsymbol{\beta}^*)$)相矛盾.定理 1 成立. □

定理 2. 若 $\boldsymbol{v}^*$是通过初值 $\boldsymbol{v}^{(0)}\neq\boldsymbol{0}$ 和梯度下降法迭代获得,那么 $\boldsymbol{v}^*$是 $F(\boldsymbol{v})$的全局最小值点.

证明:(反证法)当梯度下降法迭代终止时,有 $\frac{\partial F}{\partial\boldsymbol{v}^*}=\boldsymbol{0}$,根据定理 1 可知,$\boldsymbol{v}^*$是 $F(\boldsymbol{v})$的鞍点或全局最小值点.假设 $\boldsymbol{v}^*$是鞍点,则 $H(\boldsymbol{v}^*)<0\Rightarrow\boldsymbol{A}<0$.由公式(27)可知,至少存在一个序号 $c(1\leqslant c\leqslant M)$,使得 $\frac{\partial F}{\partial\beta_c^*}<0$,那么根据函数连续性理论,在 β_c^* 的某个θ邻域($\theta>0$),记为 $\Theta=\{\beta\,|\,|\beta-\beta_c^*|<\theta\}$,当$\beta_c\in\Theta$时,$\frac{\partial F}{\partial\beta_c}<0$.所以,

- 当 $v_c=\sqrt{\beta_c}$ 时,

$$\frac{\partial F}{\partial v_c}=2v_c\frac{\partial F}{\partial\beta_c}<0 \tag{35}$$

- 当 $v_c=-\sqrt{\beta_c}$ 时,

$$\frac{\partial F}{\partial v_c}=2v_c\frac{\partial F}{\partial\beta_c}>0 \tag{36}$$

通过公式(19)迭代得到的 $v_c\left(\text{即}v_c^{(t)}\leftarrow v_c^{(t-1)}-\eta'\frac{\partial F}{\partial v_c}\right)$,由公式(35)、公式(36)可知:当 $\boldsymbol{v}^{(0)}\neq\mathbf{0}$ 时,迭代到的 v_c^* 不可能达到 0;同理,当 $\boldsymbol{v}^{(0)}\neq\mathbf{0}$ 时,迭代到 $\boldsymbol{v}^*$ 也不可能是其鞍点.定理 2 成立. □

3.2 一般化误差界分析

根据第 2.1 节可知,MMLVM 算法实际处理的样本空间是 $Z\subset\Re^N$,假设 Z 的真实概率分布为 $p(z)$,并且 $\boldsymbol{Z}^N=\{z_1,\ldots,z_N\}$是从此分布中独立采样获得的实验样本.而最优参数$\boldsymbol{\alpha}$从统计学习角度往往可以通过期望风险(损失)最小化实现[13,14].因此,对于给定的损失函数 $\mathcal{L}(\boldsymbol{\alpha},z)$,利用最小化公式(37)来求解$\boldsymbol{\alpha}$.

$$R(\boldsymbol{\alpha})=E[\mathcal{L}(\boldsymbol{\alpha},z)]=\int\mathcal{L}(\boldsymbol{\alpha},z)\,p(z)\mathrm{d}z \tag{37}$$

但不知道 $p(z)$的具体分布,通常做法是最小化实验样本 $\boldsymbol{Z}^N$ 的实验风险,即最小化公式(38)来求解$\boldsymbol{\alpha}$.

$$R(\boldsymbol{\alpha},\boldsymbol{Z}^N)=\frac{1}{N}\sum_{n=1}^{N}\mathcal{L}(\boldsymbol{\alpha},z_n) \tag{38}$$

定义 2(ε-覆盖数). 设 $f(\boldsymbol{\alpha},z)$是实值函数,$\boldsymbol{Z}^N=\{z_1,\ldots,z_N\}$是采样的实验样本,令

$$f(\boldsymbol{\alpha},\boldsymbol{Z}^N)=(f(\boldsymbol{\alpha},z_1),\ldots,f(\boldsymbol{\alpha},z_N))^T\in\Re^N.$$

若存在 $\boldsymbol{u}\in\{\boldsymbol{u}_j|\boldsymbol{u}_j\in\Re^N,1\leqslant j\leqslant m\}$,使得对于任意的$\boldsymbol{\alpha}$,不等式$\|f(\boldsymbol{\alpha},\boldsymbol{Z}^N)-\boldsymbol{u}\|_p\leqslant N^{1/p}\varepsilon$成立.那么,$f$ 在 p 范数和 $\boldsymbol{Z}^N$ 样本下的ε-覆盖数为 m 的最小值,记为 $\mathcal{N}_p(f,\varepsilon,\boldsymbol{Z}^N)$,故 $\mathcal{N}_p(f,\varepsilon,N)=\sup_{\boldsymbol{Z}^N}N_p(f,\varepsilon,\boldsymbol{Z}^N)$.

定理 3. 给定$\forall\varepsilon>0$ 和概率分布 $p(z)$,有

$$P[\sup_{\boldsymbol{\alpha}}|R(\boldsymbol{\alpha},\boldsymbol{Z}^N)-R(\boldsymbol{\alpha})|>\varepsilon]\leqslant 8E[\mathcal{N}_1(\mathcal{L},\varepsilon/8,\boldsymbol{Z}^N)]\exp\left(\frac{-N\varepsilon^2}{128M^2}\right) \tag{39}$$

其中, $M=\sup_{\boldsymbol{\alpha},z}\mathcal{L}(\boldsymbol{\alpha},z)-\inf_{\boldsymbol{\alpha},z}\mathcal{L}(\boldsymbol{\alpha},z)$.

证明:证明过程见文献[14]. □

定理 4. 若 $f(\boldsymbol{\alpha},z)=\boldsymbol{\alpha}^Tz$(线性函数),且$\|z\|_\infty\leqslant b$ 和$\|\boldsymbol{\alpha}\|_1\leqslant a$,那么有

$$\log_2\mathcal{N}_2(f,\varepsilon,N)\leqslant\left\lceil\frac{a^2b^2}{\varepsilon^2}\right\rceil\log_2(2N+1) \tag{40}$$

证明:证明过程见文献[15],将文献[15]定理 3 中的 p 和 q 分别取值∞和 1 即可证得. □

定理 5. 给定$\forall\varepsilon>0$ 和概率分布 $p(z)$,对于本文 MMLVM 算法,有

$$P[\sup_{\boldsymbol{\alpha}}|R(\boldsymbol{\alpha},\boldsymbol{Z}^N)-R(\boldsymbol{\alpha})|>\varepsilon]\leqslant 8(2N+1)^{\left\lceil\frac{256\gamma^2}{\varepsilon^2}\right\rceil}\exp\left(\frac{-N\varepsilon^2}{512(1+2\gamma)^2}\right) \tag{41}$$

其中,γ=max(1,γ).

证明:因为 $\boldsymbol{\alpha}_+^T\mathbf{1}=1,\boldsymbol{\alpha}_-^T\mathbf{1}=1,\boldsymbol{\alpha}\geqslant\mathbf{0}$,所以$\|\boldsymbol{\alpha}\|_1\leqslant 2$.当取 Gaussian 核及根据公式(10)有$\|z\|_\infty\leqslant\max(1,\gamma)$,并简写为$\|z\|_\infty\leqslant\gamma$.因为 MMLVM 算法中的 Margin 为$\rho(\boldsymbol{\alpha},z)=\boldsymbol{\alpha}^Tz$(线性函数),所以根据定理 4(即公式(40))有

$$\mathcal{N}_2(\rho,\varepsilon,N)\leqslant(2N+1)^{\left\lceil\frac{4\gamma^2}{\varepsilon^2}\right\rceil} \tag{42}$$

但 MMLVM 算法实现最终采用公式(12)的对数形式,因此构造对数损失函数 $\mathcal{L}(\boldsymbol{\alpha},z)=\ln(1+\exp(-\rho(\boldsymbol{\alpha},z)))$. 因为 $\mathcal{L}(\boldsymbol{\alpha},z)$ 是带 L=1 常数的 Lipschitz 函数[11,16],所以,

$$E[\mathcal{N}_1(\mathcal{L},\varepsilon,\boldsymbol{Z}^N)]\leqslant\mathcal{N}_1(\mathcal{L},\varepsilon,N)\leqslant\mathcal{N}_1(\rho,\varepsilon/L,N)=\mathcal{N}_1(\rho,\varepsilon,N) \tag{43}$$

同时,根据定义 2 和 Jensen 不等式有

$$\mathcal{N}_1(\rho,\varepsilon,N)\leqslant\mathcal{N}_2(\rho,\varepsilon,N) \tag{44}$$

因为

$$\begin{aligned}\mathcal{L}(\boldsymbol{\alpha},z)&=\ln(1+\exp(-\rho(\boldsymbol{\alpha},z)))\leqslant 1+\exp(-\rho(\boldsymbol{\alpha},z))\\&\leqslant 1+|\rho(\boldsymbol{\alpha},z)|=1+|\boldsymbol{\alpha}^T z|\leqslant 1+|\boldsymbol{\alpha}^T\|z\|_\infty\mathbf{1}|\\&=1+\|\boldsymbol{\alpha}\|_1\|z\|_\infty\leqslant 1+2\gamma\end{aligned}\tag{45}$$

所以,

$$M=\sup_{\boldsymbol{\alpha},z}\mathcal{L}(\boldsymbol{\alpha},z)-\inf_{\boldsymbol{\alpha},z}\mathcal{L}(\boldsymbol{\alpha},z)\leqslant 2(1+2\gamma)\tag{46}$$

根据公式(42)、公式(43)、公式(46)和定理 3(公式(39))可知定理 5 成立. □

定理 6. 给定 $0<\delta<1$,那么 MMLVM 算法中的一般化误差界满足下式且以 $1-\delta$概率成立.

$$\begin{aligned}|R(\boldsymbol{\alpha},\boldsymbol{Z}^N)-R(\boldsymbol{\alpha})|&<\sqrt{\frac{512(1+2\gamma)^2}{2}\left(\frac{\ln(8/\delta)}{N}+\sqrt{\left(\frac{\ln(8/\delta)}{N}\right)^2+\frac{2\gamma^2}{(1+2\gamma)^2}\frac{\ln(2N+1)}{N}}\right)}\\&=16(1+2\gamma)\sqrt{\frac{\ln(8/\delta)}{N}+\sqrt{\left(\frac{\ln(8/\delta)}{N}\right)^2+\frac{2\gamma^2}{(1+2\gamma)^2}\frac{\ln(2N+1)}{N}}}\end{aligned}\tag{47}$$

证明:令公式(41)右侧项等于δ,并求出ε即可证明定理 6 成立. □

可见,公式(47)中 $\ln(2N+1)/N$ 随着 N 的增大而逐渐变小.因此,MMLVM 算法处理较大样本时有更好的精度.第 4.4 节中的 USPS 实验结果验证了此结论.同时,这一结论也是熟知的常识,即当使用 KDE 来表达数据样本时,通常需要样本数足够多.

3.3 复杂度分析

根据公式(9)计算 $\boldsymbol{z}_n$,其计算复杂度为 $O(N^2)$,这与许多基于 QP 求解的分类器,如 SVM,SVDD 等,在计算核矩阵时的复杂度相同.根据公式(14)的凸优化问题可知,对于给定$\boldsymbol{\beta}$,其计算复杂度为 $O(N)$,即单次迭代的计算复杂度为 $O(N)$.假设求解过程进行了 T 次迭代,则 MMLVM 算法求解最优权的计算复杂度为 $O(TN)$,显然与样本成线性关系.而基于 QP 求解的方法(如 SVM,SVDD 等),其计算复杂度不小于 $O(N^2)$,甚至达到或超过 $O(N^3)$[17−19].因此,当 N 较大时,本文算法在训练速度上具有较大优势.

4 实验结果与分析

所有实验均在主频 2.6GHz,2GRAM,Intel Core(TM),XP 系统的计算机上完成,程序运行于 Matlab2009a 平台.核函数是带宽参数为σ的高斯核 $k(\boldsymbol{x}_i,\boldsymbol{x}_j)=\exp(-\|\boldsymbol{x}_i-\boldsymbol{x}_j\|^2/h)$.所有算法的性能分别从测试精度、训练时间 Training(单位:s)和分类测试时间 Testing(单位:s)等 3 个方面进行比较.同时,考虑到训练数据的不平衡,测试精度采用几何精度 g 进行评价,该方法常用于评价不平衡数据集[3,20].即,分别统计正负类的分类精度 a^+和 a^-,则 $g=\sqrt{a^+\cdot a^-}$,其中,a^+和 a^-分别用下式进行计算:

$$a^+=\frac{\#\text{ positive samples correctly classified}}{\#\text{total positive samples classified}}\times 100\%,$$

$$a^-=\frac{\#\text{ negative samples correctly classified}}{\#\text{total negative samples classified}}\times 100\%.$$

注意,如本文引言所述,尽管 MMLVM 是一种新的分类器,但 L2-核分类器在分类精度和训练时间上与 C-SVM(软间隔的 SVM 版本)[4,5]相比并没有优势[6],特别是在分类精度上.同时,本文算法是建立在概率密度测度上的,因此,本文在实验安排上将 MMLVM 与 C-SVM 和 Parzen 窗(parzen window,简称 PW)[8,21]密度估计进行比较,并从测试精度、训练速度以及测试速度等 3 个方面进行综合评价.因 PW 算法不需要训练而是直接进行未知样本的测试,故后面 PW 实验部分也只给出了测试精度和测试速度两个指标.

4.1 合成数据集实验

4.1.1 Gaussian 混合模型实验

本节使用一维高斯混合模型分析 MMLVM 的密度逼近以及分类 Margin 的特征.正负类样本概率密度分别

为 $p_+(\boldsymbol{x})=0.7\phi(\boldsymbol{x};\mu_1^+,\sigma_1^+)+0.3\phi(\boldsymbol{x};\mu_2^+,\sigma_2^+)$ 和 $p_-(\boldsymbol{x})=0.6\phi(\boldsymbol{x};\mu_3^-,\sigma_3^-)+0.4\phi(\boldsymbol{x};\mu_4^-,\sigma_4^-)$.其中,$\phi(\boldsymbol{x};\mu,\sigma)$是均值为$\mu$、方差为$\sigma^2$的高斯函数.正负类训练样本从区间[−5,15]上各随机采样 500 个点,图 1 给出了实验结果.

图 1 中,图 1(a)~图 1(c)对应 $\mu_1^+=-1,\mu_2^+=3,\sigma_1^+=1,\sigma_2^+=0.707,\mu_1^-=8,\mu_2^-=11,\sigma_1^-=0.707,\sigma_2^-=1$,此时,两类样本密度重叠很小.图 1(a)中的实线是正类样本的密度曲线 $p_+(\boldsymbol{x})$,虚线是 $\hat{p}_+(\boldsymbol{x};\boldsymbol{\alpha}_+)$;图 1(b)中的实线是正类样本的密度曲线 $p_-(\boldsymbol{x})$,虚线是 $\hat{p}_-(\boldsymbol{x};\boldsymbol{\alpha}_-)$;图 1(c)中的实线是两类样本的密度差曲线 $d_\gamma(\boldsymbol{x})$,虚线是 $\hat{d}_\gamma(\boldsymbol{x};\boldsymbol{\alpha})$.从图 1(a)、图 1(b)中可以看出,估计的密度曲线与实际密度曲线具有相同的趋势,但进行了一定的伸缩,主要是考虑到各个训练样本点在分类时的 Margin,并尽可能使得总体 Margin 最大化,确保大间隔分类.

而图 1(d)~图 1(f)对应 $\mu_1^+=-1,\mu_2^+=4,\sigma_1^+=1,\sigma_2^+=0.707,\mu_1^-=5,\mu_2^-=9,\sigma_1^-=0.707,\sigma_2^-=1$,此时,两类样本密度在区间[3,6]重叠较大.图 1(d)中的实线是正类样本的密度曲线 $p_+(\boldsymbol{x})$,虚线是 $\hat{p}_+(\boldsymbol{x};\boldsymbol{\alpha}_+)$;图 1(e)中的实线是正类样本的密度曲线 $p_-(\boldsymbol{x})$,虚线是 $\hat{p}_-(\boldsymbol{x};\boldsymbol{\alpha}_-)$;图 1(f)中的实线是两类样本的密度差曲线 $d_\gamma(\boldsymbol{x})$,虚线是 $\hat{d}_\gamma(\boldsymbol{x};\boldsymbol{\alpha})$.从图 1(d)、图 1(e)中可以看出,估计的密度曲线与实际密度曲线具有相同的趋势,但进行了一定的平移,通过平移尽可能地拉大两类样本间的重叠程度,使得分类间隔最大,如图 1(e)所示.

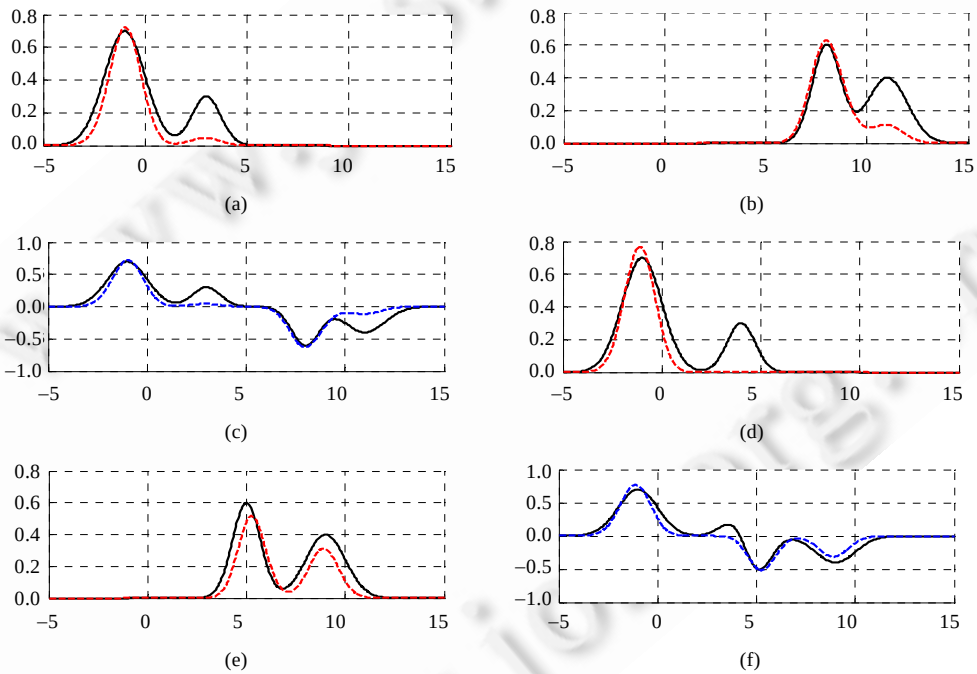


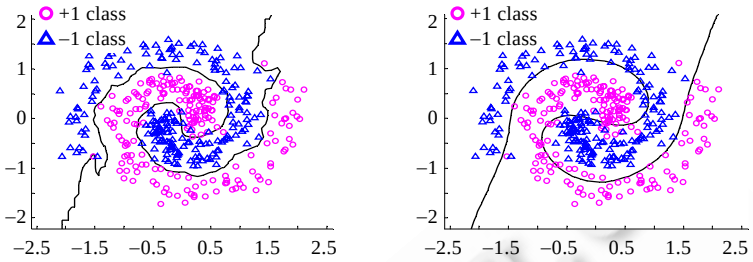
Fig.1 The effect of maximizing the DoD margin

图 1 最大化密度差 Margin 效果

4.1.2 Spiral 数据集实验

本节利用螺旋型数据集分析 MMLVM 算法的分类界面特征,并与 C-SVM 进行比较.螺旋型数据集是比较经典的人工数据集,分类器准确划分此类数据集一般相对较难[22−24],进行比较的实验结果如图 2 所示.

从图 2 可以看出,MMLVM 的分界面比 C-SVM 更优,更能反映出样本的分布形状,误分样本的可能性明显低于 C-SVM.但图 2 也表明,MMLVM 算法获得的分界面相对较硬,而 C-SVM 比较平滑(相对较软).这一点可能会降低 MMLVM 的泛化性能,该问题将作为我们近期研究的重点,这里不再进行深入探讨.



(a) MMLVM 分类界面 (b) C-SVM 分类界面

Fig.2 The classification boundary of MMLVM and C-SVM on the spiral-shaped data

图 2 MMLVM 和 C-SVM 在 Spiral 上的分类界面

4.2 UCI实验

本节利用表 1 所示的 7 种 UCI 数据集比较 3 种算法性能.注意,B.Cancer 的数据样本相对较大.

Table 1 The UCI data sets

表 1 UCI 数据集

数据集	维数	样本数	+1 类	−1 类
Wine	13	178	59	119
Iris	4	150	50	100
Biomed	5	194	67	127
Hepatitis	19	155	123	32
C. Bench	60	208	111	97
S. Heart	44	267	212	55
B. Cancer	9	699	241	458

实验参数的选择:以训练样本 2 范数的平均值 s 为基准,在 MMLVM 实验中,高斯核带宽参数σ从网格{$s^2/512$,$s^2/256$,$s^2/128$,$s^2/64$,$s^2/32$,$s^2/16$,$s^2/4$,s^2}中选择,迭代停止参数 e=0.001;在 C-SVM 实验中,带宽参数σ从网格{$s^2/128$,$s^2/64$,$s^2/32$,$s^2/16$,$s^2/8$,$s^2/4$,$s^2/2$,s^2,$2s^2$,$4s^2$,$8s^2$}中选择,而惩罚因子 $C^{[2,5]}$从网格{0.01,0.02,0.05,0.1,0.2,0.5,1,2,5,10,20,50}×N(N是训练样本数)中选择;在 PW 实验中,采用 Gaussian 窗进行估计,带宽从网格{0.01,0.02,0.05,0.1,0.2,0.5,1,2,5,10,20,50}中选择.

测试方法:从+1 和−1 类中各随机取 50%构成训练样本,各类剩余的 50%样本构成测试样本.选定最优参数并 10 次随机运行后,用均值和标准差统计分类几何精度 g、训练时间 Training 和分类测试时间 Testing.表 2 给出了实验结果.

Table 2 The experimental results on UCI

表 2 UCI 数据集性能比较结果

Dataset	MMLVM			C-SVM			PW	
	g (%)	训练时间(s)	测试时间(s)	g (%)	训练时间(s)	测试时间(s)	g (%)	测试时间(s)
Wine	**92.33±2.90**	**0.3153±0.0155**	0.0555±0.0007	90.19±2.52	0.3969±0.1351	0.0549±0.0006	91.28±2.74	**0.0305±0.0013**
Iris	**96.06±1.63**	0.4370±0.0212	0.0389±0.0007	94.91±2.18	**0.2340±0.1415**	0.0384±0.0010	95.64±1.32	**0.0227±0.0013**
Biomed	83.28±3.34	0.5448±0.0238	0.0647±0.0006	**84.24±3.86**	**0.4752±0.1587**	0.0638±0.0007	82.74±4.24	**0.0351±0.0012**
Hepatitis	**53.37±4.99**	0.4102±0.0377	0.0433±0.0006	42.57±7.59	**0.1259±0.1422**	0.0426±0.0007	49.11±5.62	**0.0245±0.0013**
C.Bench	**81.38±2.79**	**0.3653±0.1225**	0.0865±0.0006	80.68±2.85	0.3996±0.1576	0.0848±0.0015	79.82±2.29	**0.0471±0.0020**
S.Heart	**76.50±3.28**	**0.1772±0.0122**	0.1351±0.0015	56.69±7.12	0.2477±0.1426	0.1351±0.0010	59.06±4.07	**0.0734±0.0030**
B.Cancer	96.09±1.05	**2.8338±0.8020**	0.8507±0.0028	**97.07±0.77**	34.7411±1.5839	0.8419±0.0036	96.51±0.93	**0.4370±0.0029**
平均	**82.72±2.85**	**0.7262±0.1478**	**0.1821±0.0011**	**78.05±3.84**	**5.2315±0.3517**	**0.1802±0.0013**	**79.17±3.03**	**0.0958±0.0019**

表 2 中最后一行是算法在 UCI 数据集上的平均性能.从表 2 可以看出:

(1) 在几何精度上,本文的 MMLVM 算法在 Wine 等 5 个数据集上略优于 C-SVM,其他 2 个略低于 C-SVM 算法;并且除 B.Cancer 外的 6 个数据集上优于 PW 算法.因此,在精度上具有一定的优势;

(2) 在训练速度上,除 Iris,Biomed 和 Hepatitis 外,MMLVM 快于 C-SVM,特别是对于较大样本的 B.Cancer 数据集.故相比于 C-SVM 而言,本文方法具有绝对的优势;

(3) 在分类测试速度上,MMLVM 均不如 C-SVM 和 PW 快.显然,本文的 MMLVM 算法在分类测试速度上没有任何优势.这是因为 MMLVM 是基于概率密度为测度的,而 PW 不涉及权向量的优化;

(4) 从平均性能上看,PW 分类速度较快,而 MMLVM 略劣于 C-SVM;而在几何精度上,MMLVM 明显优于 C-SVM 和 PW.

4.3 PIE实验

本节利用 Pose-Illumination-Expression(PIE)人脸图像数据比较两种算法的性能,此数据集见文献[25],并能从网站 http://people.cs.uchicago.edu/~xiaofei/下载得到.实验数据集的构成:从 PIE 数据库中选择标号为 1,2, 35 和 38 的人脸图像构成 4 个实验类,其中 2 人为男性,2 人为女性.38 号人脸图像为 164 张,其余为 170 张.如图 3 所示,每张图像用一个 1024(32×32 个像素点)维向量表示.训练样本和测试样本的构成:每次实验从 4 个实验类中选择 1 类作为+1 类,其他类作为–1 类,并分别从+1 类和–1 类中随机抽取 50%构成训练样本,剩余 50%构成测试样本.参数选择同 UCI 实验,每个实验类随机运行 10 次,并统计比较算法的性能,表 3 给出了实验结果.

Fig.3 The PIE data sets

图 3 PIE 数据集

Table 3 The experimental results on PIE

表 3 PIE 数据集性能比较结果

+1 类	MMLVM			C-SVM			PW	
	g (%)	训练时间(s)	测试时间(s)	*g* (%)	训练时间(s)	测试时间(s)	*g* (%)	测试时间(s)
1	98.26±0.85	**5.5899±0.8156**	4.0276±0.0036	97.49±1.53	11.3894±0.7218	4.2225±0.4954	**98.48±0.95**	**2.7172±0.0621**
2	98.55±1.53	**5.6766±0.5394**	4.0758±0.0388	**100.00±0.00**	11.1133±0.7779	4.0726±0.0074	98.30±0.82	**2.7859±0.0928**
35	99.19±1.13	**5.3528±0.3469**	4.0650±0.0289	**99.53±0.31**	10.2226±0.5670	4.0745±0.0461	98.97±0.86	**2.7743±0.0967**
38	99.42±0.60	**5.1945±0.3190**	4.0532±0.0117	**99.82±0.28**	9.9878±0.9287	4.0420±0.0057	97.53±1.94	**2.8840±0.0853**
平均	**98.86±1.03**	**5.4535±0.5052**	**4.0554±0.0208**	**99.21±0.53**	**10.6783±0.7489**	**4.1029±0.1387**	**98.32±1.14**	**2.7904±0.0842**

从表 3 可以看出:

(1) 几何精度方面,MMLVM 算法在 1 号人脸数据集上优于 C-SVM,而在其他 3 个数据集上略劣于 C-SVM,但相差较小;相比于 PW,MMLVM 除 1 号人脸略低于 PW 外,在其他 3 副人脸上的精度均高于 PW;

(2) 训练速度方面,MMLVM 在 4 个人脸数据集上均快于 C-SVM 近一倍,表现出较大优势;

(3) 测试速度方面,在 1 号和 35 号人脸数据集上,MMLVM 略快于 C-SVM,但在其他 2 个数据集,其速度略慢于 C-SVM;但 PW 表现比较好;

(4) 平均性能方面,几何精度上 MMLVM 略低于 C-SVM 而高于 PW;训练速度上明显优于 C-SVM;测试速度上 MMLVM 略劣于 C-SVM 而较慢于 PW.

4.4 USPS实验

本节利用 USPS 手写数字图像数据集比较本文算法和 C-SVM 的性能.USPS 数据集包含 7 291 个训练样本和 2 007 个测试样本,并能从 http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/下载得到,每幅图像用 256(16×16 个像素点)维的向量表示[26],如图 4 所示.实验选择 USPS 的训练样本为实验数据集,其中,10 个数字 0~9 的手写图像分别有 1 194,1 005,731,658,652,556,664,645,542 和 644 幅.

训练样本和测试样本的构成:依次选择数字 0~9 为目标数字,并用目标数字样本构成+1 类,并从中随机抽取 50%构成+1 类训练样本,剩余 50%样本用于测试;而非目标数字样本构成−1 类,并从中随机抽取 10%构成−1 类训练样本,剩余 90%样本也用于测试.参数选择同 UCI 实验,每个目标数字随机运行 10 次,并统计比较算法的性能.注意,此实验中训练和测试样本数均相对较大(如当选择数字 0 为+1 类时,训练样本数为 1 206 个,测试样本数为 6 085 个).同时,考虑到测试样本存在不平衡性,如选 9 为目标数字时,+1 类测试样本只有 322 个,而−1 类测试样本大约有 6 000 个,故本节实验测试精度也采用几何精度 g 进行评价.表 4 给出了实验结果.

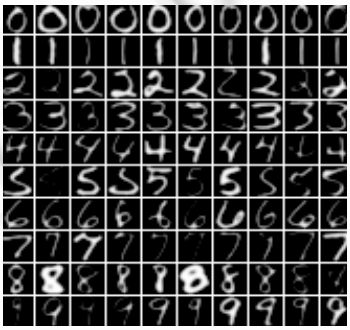


Fig.4 The USPS data sets

图 4 USPS 数据集

Table 4 The experimental results on USPS

表 4 USPS 数据集性能比较结果

+1 类	MMLVM			C-SVM			PW	
	g (%)	训练时间(s)	测试时间(s)	g (%)	训练时间(s)	测试时间(s)	g (%)	测试时间(s)
0	98.32±0.18	**28.7070±0.3384**	116.9733±0.1539	98.13±0.44	349.8524±6.8268	**115.8714±0.6999**	**99.42±0.17**	144.7281±2.7782
1	99.69±0.10	**26.5461±0.1535**	109.3729±0.1191	99.59±0.14	381.3839±78.7584	**108.2440±0.1762**	**99.73±0.07**	136.2827±1.8880
2	**98.64±0.16**	**20.0057±0.3065**	99.8076±0.1822	97.80±0.35	157.3837±11.9969	**98.6530±0.1888**	97.70±0.44	119.1150±2.2439
3	**97.70±0.42**	**20.0505±0.8397**	91.5437±0.0936	95.93±0.72	149.1242±4.1987	**90.5179±0.1461**	97.55±0.48	143.2355±1.2313
4	**98.14±0.29**	**20.6129±0.7868**	95.4995±0.1310	96.50±0.40	176.4192±17.8799	**94.5848±0.1705**	96.71±0.47	107.6324±0.9318
5	**97.73±0.41**	**19.2690±0.9521**	90.0319±0.0684	94.15±0.83	117.1604±5.4007	**89.2806±0.0923**	96.29±0.42	103.9008±0.6259
6	98.72±0.13	**20.7755±0.7202**	97.4801±0.1343	97.55±0.85	168.1536±3.3706	**96.1324±0.1733**	**98.97±0.43**	138.1549±4.3794
7	**98.26±0.36**	**20.3595±0.7049**	95.4408±0.1194	98.06±0.37	170.3127±1.8906	**94.2340±0.2029**	98.08±0.43	137.7472±0.6065
8	**97.61±0.34**	**18.2450±0.6401**	91.9850±0.1540	93.88±1.17	123.7756±5.1443	**91.4085±0.3170**	97.48±0.41	138.3551±0.2853
9	97.35±0.21	**20.6772±0.9905**	95.6477±0.2068	97.26±0.30	164.2569±3.8282	**94.2925±0.2077**	**98.48±0.39**	116.2357±1.0391
平均	**98.22±0.26**	**21.5248±0.6433**	**98.3783±0.1363**	**96.89±0.56**	**195.7823±13.9295**	**97.3219±0.2375**	**98.04±0.37**	**128.5387±1.6009**

从表 4 可以看出,在 USPS 数据集上:

(1) 几何精度上,MMLVM 均优于 C-SVM,而有 6 个手写数字上精度高于 PW,MMLVM 体现了较好优势;

(2) 训练速度方面,MMLVM 是 C-SVM 的近 7 倍以上,这说明 MMLVM 适合处理较大样本的数据集;

(3) 测试速度方面,对于每个目标数字 MMLVM 均略慢于 C-SVM 的测试速度,但相差不大;而此时 PW 的测试速度就相对较慢;

(4) 平均性能方面,MMLVM 在精度上略劣于 C-SVM 和 PW,在训练速度上也明显优于 C-SVM,而在测试速度上 MMLVM 与 C-SVM 相近,均优于 PW 算法.

综合 UCI,PIE 和 USPS 这 3 个不同类型数据集的实验结果表明,本文提出的 MMLVM 在几何精度和训练速度上具有较好的优势,特别是在处理较大样本时,优势更加明显.

5 结 论

一般地,核分类器具有相同形式,如 SVM、L2-核分类器、KDE 等,因此,通过 KDE,一个核分类器可认为是正负类样本概率密度的差.在此基础上,本文利用密度差构造了一种分类间隔,并通过最大化此间隔建立 MMLVM 算法的数学模型.该模型采用对数形式描述,进而转化为一个可以采用梯度下降法求解的最优化问题.更重要的是,在理论上保证了 MMLVM 权的全局最优性,并给出了 MMLVM 算法一般化的误差界.UCI,PIE 和 USPS 这 3 个数据集的实验结果表明,相比于 C-SVM 算法,本文 MMLVM 算法在测试精度和训练速度上有较大的优势,特别是对于较大样本.但是,正如实验结果分析一样,MMLVM 的分界面相对较硬,还需要进一步的完善和改进,这些将作为我们下一步的研究工作.

致谢 感谢审稿专家给本文提出的宝贵意见.

References:

[1] Sun JX. Modern Pattern Recognition. 2nd ed., Beijing: Higher Education Press, 2008 (in Chinese).

[2] Tax DMJ, Duin RPW. Support vector data description. Machine Learning, 2004,54(1):45−66. [doi: 10.1023/B:MACH.0000008084.60811.49]

[3] Wu MR, Ye JP. A small sphere and large margin approach for novelty detection using training data with outliers. IEEE Trans. on Pattern Analysis and Machine Intelligence, 2009,31(11):2088−2092. [doi: 10.1109/TPAMI.2009.24]

[4] Cortes C, Vapnik V. Support vector networks. Machine Learning, 1995,20(3):273−297. [doi: 10.1023/A:1022627411411]

[5] Schölkopf B, Smola A, Williamson RC, Bartlett PL. New support vector algorithms. Neural Computation, 2000,12(5):1207−1245. [doi: 10.1162/089976600300015565]

[6] Kim J, Scott CD. L2 kernel classification. IEEE Trans. on Pattern Analysis and Machine Intelligence, 2010,32(10):1822−1831. [doi: 10.1109/TPAMI.2009.188]

[7] Girolami M, He C. Probability density estimation from optimally condensed data samples. IEEE Trans. on Pattern Analysis and Machine Intelligence, 2003,25(10):1253−1264. [doi: 10.1109/TPAMI.2003.1233899]

[8] Duda RO, Hart PE, Stork DG. Pattern Classification. 2nd ed., New York: Wiley, 2000.

[9] Meinicke P, Twellmann T, Ritter H. Maximum contrast classifiers. In: Proc. of the 2002 Int'l Conf. on Artificial Neural Networks. London: Springer-Verlag, 2002. 745−750. [doi: 10.1007/3-540-46084-5_121]

[10] Deng ZH, Chung FL, Wang ST. Robust relief-feature weighting, margin maximization and fuzzy optimization. IEEE Trans. on Fuzzy Systems, 2010,18(4):726−744. [doi: 10.1109/TFUZZ.2010.2047947]

[11] Sun YJ, Todorovic S, Goodison S. Local-Learning-Based feature selection for high-dimensional data analysis. IEEE Trans. on Pattern Analysis and Machine Intelligence, 2010,32(9):1610−1626. [doi: 10.1109/TPAMI.2009.190]

[12] Horn RA, Johnson CR. Matrix Analysis. Cambridge: Cambridge University Press, 1985.

[13] Herbrich R. Learning Kernel Classifiers Theory and Algorithms. Cambridge: MIT Press, 2001.

[14] Pollard D. Convergence of Stochastic Processes. New York: Springer-Verlag, 1984.

[15] Zhang T. Covering number bounds of certain regularized linear function classes. Machine Learning Research, 2002,2:527−550. [doi: 10.1162/153244302760200713]

[16] Ng AY. Feature selection, L1 vs. L2 regularization, and rotational invariance. In: Proc. of the 21st Int'l Conf. on Machine Learning. 2004. 78−86. [doi: 10.1145/1015330.1015435]

[17] Tsang IW, Kwok JT, Cheung PM. Core vector machines: Fast SVM training on very large data sets. Journal of Machine Learning Research, 2005,6:363−392.

[18] Deng ZH, Chung FL, Wang ST. FRSDE: Fast reduced set density estimator using minimal enclosing ball approximation. Pattern Recognition, 2008,41:1363−1372. [doi: 10.1016/j.patcog.2007.09.013]

[19] Chung FL, Deng ZH, Wang ST. From minimum enclosing ball to fast fuzzy inference system training on large datasets. IEEE Trans. on Fuzzy Systems, 2009,17(1):173−184. [doi: 10.1109/TFUZZ.2008.2006620]

[20] Kubat M, Matwin S. Addressing the curse of imbalanced training sets: One-sided selection. In: Proc. of the 14th Int'l Conf. on Machine Learning. Nashville: Morgan Kaufmann Publishers, 1997. 179−186.

[21] Parzen E. On estimation of a probability density function and mode. Annals of Mathematical Statistics, 1962,33(3):1065−1076. [doi: 10.1214/aoms/1177704472]

[22] Zhang L, Zhang B, Yin HF. An alternative covering design algorithm of multi-layer neural networks. Journal of Software, 1999, 10(7):737−742 (in Chinese with English abstract).

[23] Zhou WD, Zhang L, Jiao LC. Linear programming support vector machines. ACTA ELECTRONICA SINICA, 2001,29(11): 1507−1511 (in Chinese with English abstract).

[24] Ren SQ, Yang DG, Li X, Zhuang ZW. Piecewise support vector machines. Chinese Journal of Computers, 2009,32(1):77−85 (in Chinese with English abstract). [doi: 10.3724/SP.J.1016.2009.00077]

[25] He XF, Cai D, Niyogi P. Laplacian score for feature selection. In: Weiss Y, Schölkopf B, Platt J, eds. Advances in Neural Information Processing Systems 18. Cambridge: MIT Press, 2006. 507−514.

[26] Hull JJ. A database for handwritten text recognition research. IEEE Trans. on Pattern Analysis and Machine Intelligence, 1994, 16(5):550−554. [doi: 10.1109/34.291440]

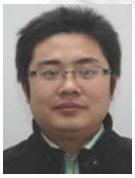
附中文参考文献:

[1] 孙即祥.现代模式识别.第 2 版,北京:高等教育出版社,2008.

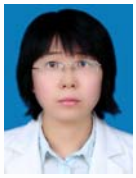
[22] 张铃,张钹,殷海风.多层前向网络的交叉覆盖设计算法.软件学报,1999,10(7):737−742.

[23] 周伟达,张莉,焦李成.线性规划支撑向量机.电子学报,2001,29(11):1507−1511.

[24] 任双桥,杨德贵,黎湘,庄钊文.分片支撑矢量机.计算机学报,2009,32(1):77−85. [doi: 10.3724/SP.J.1016.2009.00077]



胡文军(1977－),男,安徽绩溪人,博士,讲师,主要研究领域为模式识别,人工智能.



王娟(1981－),女,实验师,主要研究领域为故障检测,智能控制.

王士同(1964－),男,教授,博士生导师,主要研究领域为模式识别,人工智能,数据挖掘,模糊系统.

颜七笙(1975－),男,博士生,副教授,主要研究领域为智能计算及应用.