运筹学与控制论

DOI:10.11721/cqnuj20230613

基于 DC 规划方法的稀疏最小二乘支持向量机*

唐青青, 李国权
(重庆师范大学 数学科学学院, 重庆 401331)

摘要:最小二乘支持向量机(LS-SVM)是一种基于超平面的分类器,由于 LS-SVM 缺乏特征选择能力,因此在高维小样本数据集上的泛化表现不佳,所以有必要提高 LS-SVM 的特征选择能力。在 LS-SVM 的目标函数中引入 ℓ_0-范数正则项,提升模型的特征选择能力。然而由于 ℓ_0-范数的引入,导致新的模型不仅是非凸非光滑的,而且是 NP-难问题。为了克服这些困难,首先用一个非凸非光滑连续函数近似 ℓ_0-范数,再对该近似函数进行 DC(difference of convex functions)分解,将问题转化为 DC 规划问题,从而利用 DCA(difference of convex functions algorithm)求解该问题。该新方法的主要优点在于 DCA 的子问题具有解析解,从而使得训练速度得到很大地提升。数值实验表明,本文所提出的新方法不仅具有较好的泛化性能和特征选择能力,而且计算速度快。

关键词:最小二乘支持向量机(LS-SVM);特征选择;DC 规划;稀疏
中图分类号:O221 **文献标志码**:A **文章编号**:1672-6693(2023)06-0007-08

支持向量机(Support vector machines, SVM)是一种经典的二分类方法,不仅具有良好的泛化能力而且模型参数少,因此一直以来 SVM 备受研究者们的关注[1-4],并且被广泛应用于生物信息学[5]、医疗诊断[6]、人脸识别[7-8]、信号处理[9]等领域。但经典支持向量机的计算复杂度较高,针对这一缺点,1999 年 Suykens 等人[10]提出了对于二分类问题的最小二乘支持向量机(Least squares suppoort vector machines, LS-SVM),使用等式约束代替传统支持向量机中的不等式约束,降低了求解难度,提高了求解效率,使得计算复杂度相对减少。随后,许多关于 LS-SVM 的改进工作相继被提出,例如,2002 年 Suykens 等人[11]提出了加权最小二乘支持向量机;Wang 等人[12]提出了一种新的基于矩阵模式的 LS-SVM,该 LS-SVM 可以直接在矩阵模式上进行操作,只需要存储两个维数和偏差较低的权重向量,大大减少了所需的存储空间;在文献[13]中,作者基于截断最小二乘损失函数,提出了一种有效的鲁棒最小二乘支持向量机(RLS-SVM),可以有效地克服数据中的噪声对模型泛化性能的影响。有实验结果表明 LS-SVM 在许多问题上都得到了广泛的应用并取得了很好的效果[14-16]。

然而 LS-SVM 不具备特征选择能力,因此它在高维小样本数据集上的数值效果不佳。特征选择的主要目的是选择给定数据集的特征子集,删除冗余特征,提升模型的泛化能力,降低过拟合的风险。目前已有许多稀疏 SVM 模型相继被提出,包括 ℓ_0-SVM[17]、ℓ_2-ℓ_1-SVM[18]、ℓ_q-SVM($0<q<1$)[19]等。事实上,使用 ℓ_0-范数是寻求稀疏解的最直接的方法。然而由于 ℓ_0-范数的不连续性,使得相应的优化问题往往是一个 NP-难问题。处理这类问题的常用方法是将 ℓ_0-范数用 ℓ_1-范数进行凸近似,但是从 ℓ_0-范数到 ℓ_1-范数的松弛误差往往过大[19]。因此文献[19]使用 ℓ_q-范数($0<q<1$)近似 ℓ_0-范数,这可以视为 ℓ_0-范数的一类非凸近似,所提出的稀疏 LS-SVM 在小样本集上的泛化能力得到明显提升。

本文为了提高 LS-SVM 的特征选择能力,在传统的 LS-SVM 模型中引入 ℓ_0-范数正则项,并使用一类非凸非光滑函数对 ℓ_0-范数进行近似,再对该近似函数进行合理的 DC 分解,最后使用 DC 规划算法(DCA)进行求解。数值实验表明,新模型不仅具有很好的泛化表现和特征选择能力,而且计算速度非常快,这主要是因为新算法中的子问题具有解析解。

* **收稿日期**:2023-01-25 **修回日期**:2023-05-01 **网络出版时间**:2023-10-07T15:55
资助项目:国家自然科学基金面上项目(No. 12171063);重庆市自然科学基金项目(No. cstc2022ycjh-bgzxm0114);重庆市教委科技项目(No. KJQN 202100521)
第一作者简介:唐青青,女,研究方向为最优化理论与算法,E-mail:tangqqingzb@163.com;**通信作者**:李国权,教授,博士,E-mail:ligq@cqnu.edu.cn
网络出版地址:https://link.cnki.net/urlid/50.1165.N.20231006.1610.008

论文的安排如下:第1节主要介绍与本文相关的预备知识,包括最小二乘支持向量机和基于ℓ_q-范数的稀疏最小二乘支持向量机。第2节将给出本文所提出的新模型以及基于DCA的迭代算法。第3节给出了数值实验结果。第4节简要地总结了本文的工作。

1 预备知识

本文考虑n维欧氏空间$\mathbf{R}^n$上的二分类问题,它包含n个指标(即$x_i\in\mathbf{R}^n$)和m个样本点。记这m个样本点组成的集合为训练集$N=\{(x_i,y_i)\mid i=1,\cdots,m\}$,其中$x_i\in\mathbf{R}^n$为第$i$个输入,$x_i$的分量称为特征,$y_i\in\{+1,-1\}$为样本$x_i$的输出($i=1,\cdots,m$),即$x_i$的所属类别。令$\boldsymbol{X}\in\mathbf{R}^{m\times n}$为$m$个输入构成的矩阵,$\boldsymbol{Y}\in\mathbf{R}^{m\times m}$为相应输出构成的对角矩阵,其中$\boldsymbol{Y}_{ii}=y_i(i=1,\cdots,m)$。

接下来对LS-SVM进行简要叙述。LS-SVM[19]通过超平面

$$\boldsymbol{\omega}^{\mathrm{T}}\boldsymbol{x}+b=0 \tag{1}$$

区分两类数据,其中:$\boldsymbol{\omega}$为权重向量,b为偏置。基于最小化结构风险原则,LS-SVM的优化问题可表述为:

$$\begin{aligned}&\min_{\omega,b,\xi}\quad \frac{1}{2}\|\boldsymbol{\omega}\|^2+\frac{\gamma}{2}\boldsymbol{\xi}^{\mathrm{T}}\boldsymbol{\xi},\\&\text{s.t.}\quad \boldsymbol{Y}(\boldsymbol{X}\boldsymbol{\omega}+\boldsymbol{e}b)+\boldsymbol{\xi}=\boldsymbol{e}。\end{aligned} \tag{2}$$

式中:$\|\cdot\|$表示ℓ_2-范数即$\|\boldsymbol{\omega}\|_2^2=\sum_{i=1}^{n}|\omega_i|^2$,$\gamma>0$为参数,$\boldsymbol{\xi}\in\mathbf{R}^m$为松弛变量,$\boldsymbol{e}=(1,\cdots,1)^{\mathrm{T}}\in\mathbf{R}^m$。

引入Lagrange乘子$\boldsymbol{a}=(a_1,\cdots,a_m)^{\mathrm{T}}\in\mathbf{R}^m$,则对应于问题(2)的Lagrange函数为:

$$L(\boldsymbol{\omega},b,\boldsymbol{\xi},\boldsymbol{a})=\frac{1}{2}\|\boldsymbol{\omega}\|^2+\frac{\gamma}{2}\boldsymbol{\xi}^{\mathrm{T}}\boldsymbol{\xi}-\boldsymbol{a}^{\mathrm{T}}(\boldsymbol{Y}(\boldsymbol{X}\boldsymbol{\omega}+\boldsymbol{e}b)+\boldsymbol{\xi}-\boldsymbol{e})。$$

相应的KKT(Karush-Kuhn-Tucker)条件为:

$$\begin{cases}\dfrac{\partial L}{\partial\boldsymbol{\omega}}=0\Rightarrow\boldsymbol{\omega}=\boldsymbol{X}^{\mathrm{T}}\boldsymbol{Y}^{\mathrm{T}}\boldsymbol{a},\\[2mm]\dfrac{\partial L}{\partial b}=0\Rightarrow\boldsymbol{a}^{\mathrm{T}}\boldsymbol{Y}\boldsymbol{e}=0,\\[2mm]\dfrac{\partial L}{\partial\boldsymbol{\xi}}=0\Rightarrow\boldsymbol{a}=\gamma\boldsymbol{\xi},\\[2mm]\dfrac{\partial L}{\partial\boldsymbol{a}}=0\Rightarrow\boldsymbol{Y}(\boldsymbol{X}\boldsymbol{\omega}+\boldsymbol{e}b)+\boldsymbol{\xi}-\boldsymbol{e}。\end{cases}$$

上述条件可以表示为下列方程组的形式:

$$\begin{bmatrix}\boldsymbol{e}^{\mathrm{T}}\boldsymbol{Y} & 0\\ \boldsymbol{Y}\boldsymbol{X}\boldsymbol{X}^{\mathrm{T}}\boldsymbol{Y}^{\mathrm{T}}+\dfrac{1}{\gamma}\boldsymbol{I} & \boldsymbol{Y}\boldsymbol{e}\end{bmatrix}\begin{bmatrix}\boldsymbol{a}\\ b\end{bmatrix}=\begin{bmatrix}0\\ \boldsymbol{e}\end{bmatrix}。 \tag{3}$$

其中:$\boldsymbol{I}$为单位矩阵。求解线性方程组(3)得到$\boldsymbol{a}^*$和b^*,从而$\boldsymbol{\omega}^*=\boldsymbol{X}^{\mathrm{T}}\boldsymbol{Y}^{\mathrm{T}}\boldsymbol{a}^*$,对于新的输入向量$\boldsymbol{x}$,类别可以根据决策函数

$$j=\operatorname{sign}(\boldsymbol{\omega}^{*\mathrm{T}}\boldsymbol{x}+b^*)=\operatorname{sign}(\boldsymbol{a}^{*\mathrm{T}}\boldsymbol{Y}\boldsymbol{X}\boldsymbol{x}+b^*)$$

进行判别。

然后简要介绍ℓ_q-范数稀疏最小二乘支持向量机的相关信息。为了提升LS-SVM的特征选择能力,2018年邵等人在LS-SVM模型中引入了ℓ_q-范数,提出了一类基于ℓ_q-范数的稀疏LS-SVM[19]。首先考虑问题(2)的等价形式:

$$\min_{\omega,b}\quad f(\boldsymbol{\omega},b)=\frac{1}{2}\|\boldsymbol{\omega}\|^2+\frac{\gamma}{2}\|\boldsymbol{e}-\boldsymbol{Y}(\boldsymbol{X}\boldsymbol{\omega}+\boldsymbol{e}b)\|^2。 \tag{4}$$

计算梯度可得:

$$\begin{cases}\dfrac{\partial f}{\partial\boldsymbol{\omega}}=0\Rightarrow\boldsymbol{\omega}+\gamma\boldsymbol{X}^{\mathrm{T}}\boldsymbol{Y}^{\mathrm{T}}(\boldsymbol{Y}(\boldsymbol{X}\boldsymbol{\omega}+\boldsymbol{e}b)-\boldsymbol{e})=0,\\[2mm]\dfrac{\partial f}{\partial b}=0\Rightarrow\gamma\boldsymbol{e}^{\mathrm{T}}\boldsymbol{Y}^{\mathrm{T}}(\boldsymbol{Y}(\boldsymbol{X}\boldsymbol{\omega}+\boldsymbol{e}b)-\boldsymbol{e})=0。\end{cases} \tag{5}$$

则式(5)可写成 $\boldsymbol{Du}=\boldsymbol{s}$,其中 $\boldsymbol{D}=\boldsymbol{D}(\gamma)=\begin{bmatrix}\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X}+\dfrac{1}{\gamma}\boldsymbol{I} & \boldsymbol{X}^{\mathrm{T}}\boldsymbol{e}\\ \boldsymbol{e}^{\mathrm{T}}\boldsymbol{X} & \boldsymbol{e}^{\mathrm{T}}\boldsymbol{e}\end{bmatrix}\in\mathbf{R}^{(n+1)\times(n+1)}$ 为对称矩阵,$\boldsymbol{u}=\begin{bmatrix}\boldsymbol{\omega}\\ b\end{bmatrix}\in\mathbf{R}^{n+1}$,$\boldsymbol{s}=\begin{bmatrix}\boldsymbol{X}^{\mathrm{T}}\\ \boldsymbol{e}^{\mathrm{T}}\end{bmatrix}\boldsymbol{Ye}\in\mathbf{R}^{n+1}$。于是求解问题(4)等价于求解线性方程组 $\boldsymbol{Du}=\boldsymbol{s}$。但是当样本个数远小于特征个数时,矩阵 $\boldsymbol{D}$ 可能是病态的,会影响模型的泛化能力。为了获得稀疏解,增加分类器的特征选择能力,在文献[19]中,作者在目标函数中引入 ℓ_0-范数,从而考虑如下问题:

$$\min_{\boldsymbol{u}}\quad \|\boldsymbol{u}\|_0+\frac{1}{2\rho}\|\boldsymbol{Du}-\boldsymbol{s}\|^2, \tag{6}$$

其中:ρ 为正则化参数,$\|\boldsymbol{u}\|_0$ 为向量 $u\in\mathbf{R}^{n+1}$ 的 ℓ_0-范数,表示向量 u 的非零分量个数。但是问题(6)是 NP-难问题[20-21]。于是作者使用 ℓ_q-范数($0<q<1$)作为近似函数代替 ℓ_0-范数,即:

$$\min_{\boldsymbol{u}}\quad \|\boldsymbol{u}\|_q^q+\frac{1}{2\rho}\|\boldsymbol{Du}-\boldsymbol{s}\|^2, \tag{7}$$

其中:$\|\boldsymbol{u}\|_q^q=\sum_{i=1}^{n+1}|u_i|^q$。然而问题(7)仍为 NP-难问题[22],于是作者又引入了一个正则化参数 $\varepsilon>0$,并考虑如下问题:

$$\min_{\boldsymbol{u}}\quad \|\boldsymbol{u}\|_{q,\varepsilon}^q+\frac{1}{2\rho}\|\boldsymbol{Du}-\boldsymbol{s}\|^2, \tag{8}$$

其中:$\|\boldsymbol{u}\|_{q,\varepsilon}^q=\sum_{j=1}^{n+1}(\varepsilon+u_j^2)^{\frac{q}{2}}$,再根据一阶最优必要性条件设计出求解式(8)的迭代算法,从而获得问题(8)的稀疏解。该方法提升了 LS-SVM 的特征选择能力,然而在计算过程中每步的迭代成本过高,会导致训练时间比较长。

2　ℓ_0-范数稀疏最小二乘支持向量机

尽管 ℓ_q-范数稀疏最小二乘支持向量机能够有效地提升 LS-SVM 的特征选择能力,但是计算成本较大,尤其是当样本维度比较高时,训练时间比较长。因此本文考虑提出一个新的稀疏 LS-SVM 模型,不仅具有很好的特征选择能力,而且训练效率高。

令 $\widetilde{\boldsymbol{X}}=[\boldsymbol{X},\boldsymbol{e}]$,则问题(4)可以等价地写成:

$$\min_{\boldsymbol{u}}\quad \frac{1}{2}\|\boldsymbol{u}\|^2+\frac{\gamma}{2}\|\widetilde{\boldsymbol{X}}\boldsymbol{u}\|^2-\gamma e^{\mathrm{T}}\boldsymbol{Y}\widetilde{\boldsymbol{X}}\boldsymbol{u}。 \tag{9}$$

为了获得稀疏解提升分类器的特征选择能力,引入 ℓ_0-范数,从而得到新的 ℓ_0-范数稀疏最小二乘支持向量机问题:

$$\min_{\boldsymbol{u}}\quad \frac{1}{2}\|\boldsymbol{u}\|^2+\frac{\gamma}{2}\|\widetilde{\boldsymbol{X}}\boldsymbol{u}\|^2-\gamma e^{\mathrm{T}}\boldsymbol{Y}\widetilde{\boldsymbol{X}}\boldsymbol{u}+\lambda\|\boldsymbol{u}\|_0。 \tag{10}$$

其中:λ 为非负参数。由于 $\|\boldsymbol{u}\|_0$ 的存在,问题(10)是 NP-难问题。本文采用文献[23]中所给出的非凸函数 $\eta_\alpha(x)=\min\{1,\alpha x^2\}$ 来近似目标函数中的 ℓ_0-范数,其中 $\alpha>0$,$x\in\mathbf{R}$。则 $\|\boldsymbol{u}\|_0\approx\sum_{i=1}^{n+1}\eta_\alpha(u_i)$,从而问题(10)可近似地写成:

$$\min_{\boldsymbol{u}}\quad \psi(u):=\frac{1}{2}\|\boldsymbol{u}\|^2+\frac{\gamma}{2}\|\widetilde{\boldsymbol{X}}\boldsymbol{u}\|^2-\gamma e^{\mathrm{T}}\boldsymbol{Y}\widetilde{\boldsymbol{X}}\boldsymbol{u}+\lambda\sum_{i=1}^{n+1}\eta_\alpha(u_i)。 \tag{11}$$

问题(11)是连续的,但仍然是非凸非光滑的。受文献[23]的启发,本文采用 DC 规划算法求解问题(11)。DCA 是一种有效的线性收敛的方法[24-26],已被广泛的应用于许多非凸优化问题。

对近似函数 $\eta_\alpha(x)$ 进行 DC 分解[23]则 $\eta_\alpha(x)=\hat{g}(x)-\hat{h}(x)$,其中 $\hat{g}(x)=\alpha x^2$,$\hat{h}(x)=-1+\max\{\alpha x^2,1\}$ 都是凸函数。于是问题(11)可写成如下 DC 规划问题:

$$\min_{\boldsymbol{u}}\quad g(\boldsymbol{u})-\lambda h(\boldsymbol{u}), \tag{12}$$

其中:$g(\boldsymbol{u})=\frac{1}{2}\|\boldsymbol{u}\|^2+\frac{\gamma}{2}\|\widetilde{\boldsymbol{X}}\boldsymbol{u}\|^2-\gamma e^{\mathrm{T}}\boldsymbol{Y}\widetilde{\boldsymbol{X}}\boldsymbol{u}+\lambda\alpha\|\boldsymbol{u}\|^2$ 和 $h(\boldsymbol{u})=\sum_{i=1}^{n+1}\hat{h}(u_i)$ 都是凸函数。因为问题(12)是一个标

准的DC规划问题,所以可以通过DCA迭代求解。在每一次迭代中,需要计算次梯度 $\boldsymbol{v}^t\in\partial h(\boldsymbol{u}^t)$ 并求解子问题:

$$\min_{\boldsymbol{u}} \quad g(\boldsymbol{u})-\lambda\langle \boldsymbol{v}^t,\boldsymbol{u}\rangle。 \tag{13}$$

上述子问题是无约束凸优化问题,由无约束优化最优性充要条件可知问题(13)的解 $\boldsymbol{u}^{t+1}$ 一定满足 $\left(\frac{1}{\gamma}(1+2\lambda\alpha)\boldsymbol{I}+\widetilde{\boldsymbol{X}}^{\mathrm{T}}\widetilde{\boldsymbol{X}}\right)\boldsymbol{u}^{t+1}=\widetilde{\boldsymbol{X}}^{\mathrm{T}}\boldsymbol{Y}^{\mathrm{T}}\boldsymbol{e}+\frac{\lambda}{\gamma}\boldsymbol{v}^t$,其中 $\boldsymbol{I}\in\mathbf{R}^{n+1}$ 为单位矩阵。令 $\boldsymbol{M}=\frac{1}{\gamma}(1+2\lambda\alpha)\boldsymbol{I}$ 和 $\boldsymbol{d}=\widetilde{\boldsymbol{X}}^{\mathrm{T}}\boldsymbol{Y}^{\mathrm{T}}\boldsymbol{e}+\frac{\lambda}{\gamma}\boldsymbol{v}^t$,从而 $\boldsymbol{u}^{t+1}=\boldsymbol{L}^{-1}\boldsymbol{d}$,其中 $\boldsymbol{L}=\boldsymbol{M}+\widetilde{\boldsymbol{X}}^{\mathrm{T}}\widetilde{\boldsymbol{X}}$ 为 $n+1$ 阶方阵,n 为样本点的特征维度,对于高维小样本问题,n 往往是远远大于 m。再由Sherman-Morrion-Woodbury公式计算 L^{-1},即:

$$\boldsymbol{L}^{-1}=(\boldsymbol{M}+\widetilde{\boldsymbol{X}}^{\mathrm{T}}\widetilde{\boldsymbol{X}})^{-1}=\boldsymbol{M}^{-1}-\boldsymbol{M}^{-1}\widetilde{\boldsymbol{X}}^{\mathrm{T}}(\boldsymbol{I}+\widetilde{\boldsymbol{X}}\boldsymbol{M}^{-1}\widetilde{\boldsymbol{X}}^{\mathrm{T}})^{-1}\widetilde{\boldsymbol{X}}\boldsymbol{M}^{-1}。$$

值得注意的是 $\boldsymbol{L}^{-1}$ 的计算与迭代次数 t 无关,因此只需要计算一次逆矩阵。而且不必直接计算 $n+1$ 阶矩阵的逆矩阵 $\boldsymbol{L}^{-1}$,只需计算 m 阶矩阵 $\boldsymbol{I}+\widetilde{\boldsymbol{X}}\boldsymbol{M}^{-1}\widetilde{\boldsymbol{X}}^{\mathrm{T}}$ 的逆,这里的 m 是远远小于 n。下面给出求解问题(11)的迭代算法:

算法1 问题(11)的DC算法

步0,初始化:令 $t=0$,选择 $\boldsymbol{u}^0\in\mathbf{R}^{n+1}$;

步1,计算 $\boldsymbol{v}^t\in\partial h(\boldsymbol{u}^t)$;

步2,计算 $\boldsymbol{u}^{t+1}$,通过 $\boldsymbol{u}^{t+1}=\boldsymbol{L}^{-1}\boldsymbol{d}$,其中 $\boldsymbol{L}^{-1}=(\boldsymbol{M}+\widetilde{\boldsymbol{X}}^{\mathrm{T}}\widetilde{\boldsymbol{X}})^{-1}=\boldsymbol{M}^{-1}-\boldsymbol{M}^{-1}\widetilde{\boldsymbol{X}}^{\mathrm{T}}(\boldsymbol{I}+\widetilde{\boldsymbol{X}}\boldsymbol{M}^{-1}\widetilde{\boldsymbol{X}}^{\mathrm{T}})^{-1}\widetilde{\boldsymbol{X}}\boldsymbol{M}^{-1}$;

步3,若满足终止条件,算法停止;否则,令 $t:=t+1$,转步1;

注1 步1中涉及的 $\partial h(\boldsymbol{u}^t)$ 为 $v_i^t=\begin{cases}2\alpha u_i^t,\text{如果 }\alpha(u_i^t)^2\geqslant 1,\\0,\text{否则。}\end{cases}$

注2 文献[19]也提出了稀疏最小二乘支持向量机模型,但该文献是从问题(4)的一阶必要条件出发建立模型。而本文是直接使用DCA对问题(4)的 ℓ_0-范数正则化模型进行求解,而且通过合理地DC分解后,DCA子问题(13)不仅光滑而且具有解析解,这也恰好解释了为什么本文所提出的模型具有较快的训练速度。

注3 虽然也可以使用DC规划方法求解模型(6),但是求解子问题时需要计算 $n+1$ 阶高维矩阵的逆,这对于高维小样本数据集来说计算成本会很高,甚至是不可行的。为了克服上述问题,将问题(4)转化成问题(9)的形式,好处在于求解DC规划子问题时只需要计算 m 阶方阵的逆矩阵,这里 m 是远远小于 n 的样本点的个数,而且只需要计算一次逆矩阵,因此使用模型(9)可以克服上述问题,并且降低算法的计算复杂度。

注4 关于 ℓ_0-范数的非凸近似有很多,例如Capped-ℓ_1[27]、exponential concave function[28]、smoothly clipped absolute deviation (SCAD)[29]等等,但是通过DC分解后,子问题往往是非光滑的、不具有解析解,需要调用相关算法求解子问题,增加了算法的计算复杂度,所以本文未采用上述近似函数。

由于算法1是标准的DC算法,文献[26]已经给出了DC算法的全局收敛性结果,所以这里不再赘述。

3 数值实验

为了说明本文所提出算法(简称 ℓ_0-LSSVM)的有效性,将 ℓ_0-LSSVM 与 LSSVM[10]、ℓ_1-SVM[30]、ℓ_0-SVM[23]、ℓ_q-LSSVM[19]4种方法进行比较。实验过程中,使用LS-SVMlab工具箱实现LSSVM,使用CPLEX求解器解决算法 ℓ_1-SVM 和算法 ℓ_0-SVM 中的线性规划问题。所有实验均以Matlab 2020b为软件平台,以Intel(R) Core(TM)i7-11700(2.50 GHz)处理器,16 GB内存的PC机为硬件平台。在算法1中,当 $\|\boldsymbol{u}^{t+1}-\boldsymbol{u}^t\|\leqslant$eps 作为算法终止准则。在所有算法中,当 $|\boldsymbol{\omega}_i|<10^{-4}$ 时,令 $\boldsymbol{\omega}_i=0$,即此时认为该特征为冗余特征。所有实验利用五折交叉验证的方法对超参数进行优化。LSSVM中的参数 γ 与 ℓ_q-LSSVM中的参数 γ 取值范围保持一致,均在 $\{10^{-8},10^{-7},\cdots,10^{7},10^{8}\}$ 中进行选取,ℓ_0-LSSVM中的参数 α,λ,γ 分别从以下3个集合 $\{1,2,\cdots,9,10\}$,$\{10^{-10},10^{-9},\cdots,10^{9},10^{10}\}$,$\{10^{-10},10^{-9},\cdots,10^{9},10^{10}\}$ 中进行选取。将准确率(x_{acc})作为评价算法性能的指标,准确率的定义为 $x_{\mathrm{acc}}=\frac{x_{\mathrm{TP}}+x_{\mathrm{TN}}}{x_{\mathrm{TP}}+x_{\mathrm{TN}}+x_{\mathrm{FP}}+x_{\mathrm{FN}}}$,其中:$x_{\mathrm{TP}}$、$x_{\mathrm{TN}}$、$x_{\mathrm{FP}}$ 以及 x_{FN} 分别是真正率、真负率、假正率以及假负率。选定超参数后,在每一个数据上进行30次实验,并取30次实验的平均准确率作为实验结果。

在UCI和UCR数据库中选取了14个标准数据集进行试验比较。数据集Soybean(small)和Fbifaces来自

UCI 数据库,其余 12 个数据集均来自 UCR 数据库。表 1 给出了所有数据集的详细信息。

表 1　实验中使用的标准数据集

Tab. 1　Benchmark datasets used in experiments

数据集	样本大小	特征个数	正类样本数	负类样本数
Soybean(small)	47	35	20	27
MoteStrain	20	84	10	10
ECGFiveDays	23	136	14	9
gun	50	150	24	26
wine	57	237	30	27
ToeSegmentation1	40	277	20	20
coffee	28	286	14	14
ToeSegmentation2	36	343	18	18
ShapeletSim	20	500	10	10
Herring	64	512	39	25
BeetleFly	20	512	10	10
TwoLeadECG	20	512	10	10
BirdChicken	40	512	20	20
Fbifaces	132	800	31	101

表 2 给出了 ℓ_0-LSSVM、LSSVM、ℓ_1-SVM、ℓ_0-SVM、ℓ_q-LSSVM5 种方法在上述 14 个标准数据集上的数值表现,包括平均分类准确率、所选特征的平均数和平均训练时间,最高平均准确率采用黑体标记。从表 2 中,可以得出以下结论:

在准确率方面,本文所给出的算法 ℓ_0-LSSVM 在 14 个数据集中的 13 个数据集上取得了最高准确率,而其他方法最多只在 3 个数据集上达到最高。在 gun 数据集上,ℓ_q-LSSVM 取得最高准确率,比 ℓ_0-LSSVM 的准确率略高 0.0047。但在 ToeSegmentation1、ToeSegmentation2、ShapeletSim、BeetleFly、TwoLeadECG 和 BirdChicken 数据集上 ℓ_0-LSSVM 的准确率明显高于其余 4 个分类器。例如,在 BeetleFly 数据集上,ℓ_0-LSSVM 的准确率为 86%,LSSVM 的准确率仅为 60%,ℓ_0-LSSVM 的准确率比 LSSVM 高了 26%。同时 ℓ_0-LSSVM 在 Soybean(small)、wine 和 coffee 3 个数据集上准确率达到了 100%。因此在这些数据集上 ℓ_0-LSSVM 具有较好的泛化性能。

在特征选择方面,LSSVM 几乎不具有特征选择能力,与其余 4 个具有一定特征选择能力的分类器相比,泛化表现不佳,这也说明了提高 LSSVM 的特征选择能力是很有必要的。从实验结果可以发现,在绝大多数数据集上,ℓ_0-LSSVM 的特征数比 LSSVM 的特征数少,而且准确率也远远高于 LSSVM 的准确率。与 ℓ_q-LSSVM 相比,ℓ_0-LSSVM 的特征数在 Soybean(small)等 6 个数据集上最少,同时取得最高准确率;在其余 8 个数据集上,ℓ_0-LSSVM 的特征数虽然比 ℓ_q-LSSVM 的略高,但是在其中 7 个数据集上 ℓ_0-LSSVM 都取得了最高准确率。因此,和 LSSVM、ℓ_q-LSSVM 相比,ℓ_0-LSSVM 不仅具有很好的特征选择能力,而且泛化能力也比较高。

在训练时间方面,ℓ_0-LSSVM 在 Soybean(small)数据集上的训练时间仅为 0.000 2 s,比其余 4 个分类器的速度快了 10 倍左右,并且保证了准确率为 100%和特征数最少。在 ECGFiveDays 数据集上,ℓ_0-LSSVM 的训练速度比其余 4 个分类器快了 20 倍以上,且准确率最高。同时,ℓ_0-LSSVM 的训练速度在 14 个数据集上比其它 3 个具有特征选择能力的分类器 ℓ_1-SVM、ℓ_0-SVM 和 ℓ_q-LSSVM 都快,且训练时间均不超过 0.024 s。例如,与 ℓ_1-SVM 和 ℓ_0-SVM 相比,ℓ_0-LSSVM 在 MoteStrain、ECGFiveDays、gun、wine、Herring 和 Fbiface6 个数据集上的训练速度比这两个分类器快了 10 倍以上。尤其是和 ℓ_q-LSSVM 相比,ℓ_0-LSSVM 在训练速度上的优势非常明显,除了 MoteStrain 和 ToeSegmentation1 数据集,在剩余的 12 个数据集上 ℓ_0-LSSVM 的训练速度比 ℓ_q-LSSVM 快了至少 40 倍以上,特别是在 wine 数据集上,速度更是快了百倍,且准确率达到了 100%。因此实验结果表明 ℓ_0-LSSVM 在训练时间方面具有很好的优势。

表2 标准数据集的实验结果

Tab.1 Experimental results for benchmark datasets %

数据集	参数	LSSVM	ℓ_1-SVM	ℓ_0-SVM	ℓ_q-LSSVM	ℓ_0-LSSVM
Soybean(small)	x_{acc}/%	**100±0**	**100±0**	**100±0**	**100±0**	**100±0**
	特征个数	35	3	3	21	2
	时间/s	0.003 4	0.002 0	0.001 4	0.006 2	0.000 2
MoteStrain	x_{acc}/%	75.33±16.26	76.00±18.77	81.67±29.11	87.14±21.67	**88.00±17.89**
	特征个数	84	7	3	55	79
	时间/s	0.003 2	0.005 6	0.004 2	0.018 0	0.003 0
ECGFiveDays	x_{acc}/%	80.95±27.66	81.14±24.63	77.62±21.93	84.29±22.81	**88.67±10.43**
	特征个数	136	3	3	20	106
	时间/s	0.004 6	0.007 8	0.006 2	0.052 8	0.000 2
gun	x_{acc}/%	91.21±5.91	94.75±4.90	95.81±6.34	**97.14±6.39**	96.67±4.71
	特征个数	150	7	8	19	150
	时间/s	0.003 2	0.017 8	0.015 2	0.075 6	0.001 0
wine	x_{acc}/%	93.50±9.29	82.52±12.10	76.63±9.35	**100±0**	**100±0**
	特征个数	237	10	4	221	237
	时间/s	0.003 8	0.028 4	0.038 6	0.277 0	0.002 2
ToeSegmentation1	x_{acc}/%	60.38±17.19	63.77±23.87	70.39±6.17	70.16±17.94	**77.50±25.82**
	特征个数	277	17	15	275	177
	时间/s	0.003 6	0.048 0	0.173 0	0.198 4	0.006 2
coffee	x_{acc}/%	92.67±10.11	97.50±5.59	**100±0**	96.00±8.94	**100±0**
	特征个数	286	6	2	286	285
	时间/s	0.003 2	0.018 2	0.020 4	0.209 6	0.002 2
ToeSegmentation2	x_{acc}/%	55.56±15.71	61.89±31.11	65.67±25.37	54.11±17.51	**76.17±21.15**
	特征个数	343	20	6	320	185
	时间/s	0.003 0	0.064 2	0.197 6	0.251 6	0.006 8
ShapeletSim	x_{acc}/%	64.67±2.98	70.00±20.92	73.62±18.36	78.33±21.73	**85.00±33.54**
	特征个数	500	13	3	442	311
	时间/s	0.001 6	0.074 2	0.086 4	0.628 8	0.013 4
Herring	x_{acc}/%	61.02±13.94	60.61±3.03	65.43±10.38	63.61±11.42	**65.78±6.75**
	特征个数	512	25	14	414	188
	时间/s	0.003 0	0.199 6	0.297 0	0.925 6	0.015 6
BeetleFly	x_{acc}/%	60.00±13.69	64.00±10.84	65.33±20.63	66.43±21.37	**86.00±14.22**
	特征个数	512	7	5	69	325
	时间/s	0.001 2	0.074 8	0.096 8	0.662 4	0.011 0
TwoLeadECG	x_{acc}/%	68.33±20.75	71.67±24.72	67.50±32.60	65.00±22.36	**80.00±27.39**
	特征个数	512	8	4	21	321
	时间/s	0.001 4	0.074 8	0.070 6	0.662 4	0.012 2
BirdChicken	x_{acc}/%	72.79±6.51	77.76±17.32	80.06±14.88	73.33±6.97	**88.35±12.03**
	特征个数	512	11	7	91	502
	时间/s	0.003 2	0.110 6	0.118 4	0.693 2	0.013 8
Fbiface	x_{acc}/%	76.68±4.42	77.57±12.08	74.71±3.39	77.70±4.51	**79.38±6.97**
	特征个数	800	34	16	790	797
	时间/s	0.004 6	0.695 0	1.145 2	1.794 4	0.023 2
平均准确率/%		73.08	77.08	78.17	79.52	**86.54**

4 结论

本文研究了具有 ℓ_0-范数正则项的稀疏最小二乘支持向量机。利用非凸非光滑的连续函数近似 ℓ_0-范数并进行合理的 DC 分解，将问题转化成 DC 规划问题，最后使用 DCA 进行求解。新算法中的子问题不仅光滑而且具有解析解。实验表明本文所提出的方法具有很好的特征选择能力，而且在泛化能力和训练时间方面具有比较明显的优势。然而，本文的研究都还局限于原特征空间，后续研究将考虑研究稀疏最小二乘支持向量机的非线性分类器，从而进一步提升模型的泛化性能。

参考文献：

[1] VAPNIK V. Statistical learning theory[M]. New York：Wiley，1998.

[2] DENG N，TIAN Y，ZHANG C H. Support vector machines：optimization based theory，algorithms，and extensions[M]. Boca Raton：CRC press，2012.

[3] DING S，ZHANG X，AN Y，et al. Weighted linear loss multiple birth support vector machine based on information granulation for multi-class classification[J]. Pattern Recognition，2017，67：32-46.

[4] HUANG J，YU Z L，GU Z，et al. Sparse and heuristic support vector machine for binary classifier and regressor fusion[J]. International Journal of Machine Learning and Cybernetics，2019，10(12)：3667-3686.

[5] BURGES C J C. A tutorial on support vector machines for pattern recognition[J]. Data mining and knowledge discovery，1998，2(2)：121-167.

[6] OSOWSKI S，HOAI L T，MARKIEWICZ T. Support vector machine-based expert system for reliable heartbeat recognition[J]. IEEE transactions on biomedical engineering，2004，51(4)：582-589.

[7] SAHBI H，GEMAN D，BOUJEMAA N. Face detection using coarse-to-fine support vector classifiers[C]//Proceedings. International Conference on Image Processing. New York：IEEE，2002.

[8] SHI H P，LIU C. Face detection using discriminating feature analysis and support vector machine[J]. Pattern Recognition，2006，39(2)：260-276.

[9] KUH A. Adaptive kernel methods for CDMA systems[C]//IJCNN'01. International Joint Conference on Neural Networks. Proceedings (Cat. No. 01CH37222). New York：IEEE，2001.

[10] SUYKENS J，VANDEWALLE J. Least Squares Support Vector Machine Classifiers[J]. Neural Processing Letters，1999，9：293-300.

[11] SUYKENS J A K，DEBRABANTER J，LUKAS L，et al. Weighted least squares support vector machines：robustness and sparse approximation[J]. Neurocomputing，2002，48(1/2/3/4)：85-105.

[12] WANG Z，CHEN S. New least squares support vector machines based on matrix patterns[J]. Neural processing letters，2007，26(1)：41-56.

[13] YANG X，TAN L，HE L. A robust least squares support vector machine for regression and classification with noise[J]. Neurocomputing，2014，140：41-52.

[14] JAYADEVA，KHEMCHANDANI R，CHANDRA S. Twin support vector machines for pattern classification[J]. IEEE Transactions on pattern analysis and machine intelligence，2007，29(5)：905-910.

[15] LI C N，SHAO Y H，DENG N Y. Robust ℓ_1-norm non-parallel proximal support vector machine[J]. Optimization，2016，65(1)：169-183.

[16] YU D J，XU Z S，WANG X Z. Bibliometric analysis of support vector machines research trend：a case study in China[J]. International Journal of Machine Learning and Cybernetics，2020，11：715-728.

[17] WESTON J，ELISSEEFF A，SCHOLOKPF B，et al. Use of the zero norm with linear models and kernel methods[J]. The Journal of Machine Learning Research，2003，3：1439-1461.

[18] NEUMANN J，SCHNORR C，STEIDL G. Combined SVM-based feature selection and classification[J]. Machine learning，2005，61(1)：129-150.

[19] SHAO Y H，LI C N，LIU M Z，et al. Sparse ℓ_q-norm least squares support vector machine with feature selection[J]. Pattern Recognition，2018，78：167-181.

[20] NATARAJAN B K. Sparse approximate solutions to linear systems[J]. SIAM journal on computing，1995，24(2)：227-234.

[21] DONOHO D L，For most large underdetermined systems of linear equations，the minimal ℓ_1-norm solution is also the sparsest

solution[J]. Communications on Pure and Applied Mathematics,2006,59(6):797-829.

[22] LIU Y F,MA S,DAI Y H,et al. A smoothing SQP framework for a class of composite ℓ_q minimization over polyhedron[J]. Mathematical Programming,2016,158(1):467-500.

[23] LI G Q,YANG L X,WU Z Y,et al. DC programming for sparse proximal support vector machines[J]. Information Sciences,2021,547:187-201.

[24] LETHI H A,PHAMDINH T. DC programming and DCA:thirty years of developments[J]. Mathematical Programming,2018,169(1):5-68.

[25] WU C Z,LI C J,LONG Q. A DC programming approach for sensor network localization with uncertainties in anchor positions [J]. Journal of Industrial and Management Optimization,2014,10(3):817-826.

[26] TAO P D,AN L T H. Convex analysis approach to DC programming:theory,algorithms and applications[J]. Actamathematica vietnamica,1997,22(1):289-355.

[27] PELEG D,MEIR R. A bilinear formulation for vector sparsity optimization[J]. Signal Processing,2008,88(2):375-389.

[28] BRADLEY P S, MANGASARIAN O L. Feature selection via concave minimization and support vector machines[C]// International Conference on Machine Learning. San Francisco,CA,USA:Morgan Kaufmann Publishers Inc,1998.

[29] FAN J,LI R. Variable selection via nonconcave penalized likelihood and its oracle properties[J]. Journal of the American statistical Association,2001,96(456):1348-1360.

[30] ONG C S,AN L T H. Learning sparse classifiers with difference of convex functions algorithms[J]. Optimization Methods and Software,2013,28(4):830-854.

Operations Research and Cybernetics

Sparse Least Squares Support Vector Machines via DC Programming

TANG Qingqing, LI Guoquan

(School of Mathematical Sciences, Chongqing Normal University, Chongqing 401331, China)

Abstract: Least squares support vector machines (LS-SVM) is a hyperplane-based classifier. Due to the lack of feature selection ability, LS-SVM does not perform well on high-dimensional small sample data sets. Thus it is necessary to improve the feature selection ability of LS-SVM. The ℓ_0-norm regular term is introduced into the objective function of LS-SVM to enhance the feature selection ability of the model. However, due to the presence of the ℓ_0-norm, the new model is not only non-convex and non-smooth, but also NP-hard. In order to overcome these difficulties, a non-convex non-smooth continuous function was first used to approximate the ℓ_0-norm and then the approximate function is decomposed into a DC (difference of convex functions) programming problem, and the DCA (difference of convex functions algorithm) is used to solve the problem. The main advantage of the new method is that the subproblems of DCA have closed form solutions, which greatly improves the training speed. Numerical experiments show that the proposed new method not only has better generalization performance and feature selection ability, but also has fast computation speed.

Keywords: least squares support vector machines(LS-SVM); feature selection; DC programming; sparsity

(责任编辑 陈 乔)