DOI:10.19651/j.cnki.emt.2416006

邻域信息加权的最小二乘投影双支持向量聚类*

王顺霞[1,2] 黄成泉[1,2] 罗森艳[1,2] 杨贵燕[1,2] 蔡江海[1,2]
(1.贵州民族大学数据科学与信息工程学院 贵阳 550025; 2.贵州民族大学工程技术人才实践训练中心 贵阳 550025)

摘 要: 针对最小二乘投影双支持向量聚类(LSPTSVC)算法未充分利用样本邻域之间的潜在信息、实用性不强等问题,本文提出了一种高效的邻域信息加权的最小二乘投影双支持向量聚类算法。首先引入相对密度概念充分提取数据集中同类数据点之间的局部相似性信息,然后计算该点的相对权重,最后利用该权重获得数据点的加权平均值,来更好的反映同类样本的几何结构。实验结果验证了所提算法的有效性,结果表明本文算法在相似的计算复杂度下,相比现有方法取得了更好的聚类准确性,且在真实世界的医学数据集的实际应用中表现出良好的聚类性能。

关键词: 邻域信息;相对权重;最小二乘;双支持聚类
中图分类号: TP181;TN014 **文献标识码**: A **国家标准学科分类代码**: 520.10

Weighted least square projection twin support vector clustering with neighborhood information

Wang Shunxia[1,2] Huang Chengquan[1,2] Luo Senyan[1,2] Yang Guiyan[1,2] Cai Jianghai[1,2]
(1. School of Data Science and Information Engineering, Guizhou Minzu University, Guiyang 550025, China;
2. Engineering Training Center, Guizhou Minzu University, Guiyang 550025, China)

Abstract: In order to solve the problem that the least square projection twin support vector clustering (LSPTSVC) algorithm fails to make full use of the potential information among sample neighborhoods and is not practical, this paper proposes an efficient weighted least square projection twin support vector clustering algorithm with neighborhood information. Firstly, the algorithm introduces the concept of relative density to fully extract local similarity information between data points of the same class. Then, the algorithm calculates the relative weight of the point. Finally, in order to better reflect the geometric structure of similar samples, the algorithm calculates the weighted average value of the data points by using the relative weight. These experimental results verify the effectiveness of the algorithm. The results show that the proposed algorithm achieves better clustering accuracy than the existing methods under the similar computational complexity and good clustering performance in the practical application of medical datasets in the real world.

Keywords: neighborhood information;relative weight;least square;twin support clustering

0 引 言

聚类分析作为一种无监督学习算法,被广泛地应用于机器学习和数据挖掘的各个领域,旨在将各种形式的数据的集合分为由相似的对象组成的数个不同类[1-2]。传统的聚类算法普遍存在处理效率不高、实用性不强的现象[3],为了解决这一问题,许多研究学者在孪生支持向量机的变体中不断引入新的概念来解决数据集的分类效率问题,例如,Rezvani 等[4] 提出的直觉模糊孪生支持向量机(intuitionistic fuzzy twin support vector machine, IFTWSVM)通过隶属函数和非隶属函数来降低噪声的影响,然而,IFTWSVM 忽略了训练数据样本的局部邻域结构,且计算复杂度大,为了克服这些问题,Tanveer 等[5]提出了直觉模糊加权最小二乘双支持向量机(intuitionistic fuzzy weighted least squares twin SVMs, IFW-LSTSVM),该方法使用基于 k 近邻的加权技术的类内局部权重来结合局部邻域结构,求解一组线性方程而不是传统的一系列二次规划问题(quadratic programming problems, QPP),提高

收稿日期:2024-05-10
* 基金项目:国家自然科学基金(62062024)、贵州省模式识别与智能系统重点实验室 2022 年度开放课(GZMUKL[2022]KF03)、贵州省省级科技计划项目(黔科合基础-ZK[2021]一般 342)、贵州省教育厅自然科学研究项目(黔教技[2022]015)资助

了算法的学习效率。Hua 等[6]提出的具有局部信息的加权最小二乘投影双支持向量机(weighted least squares projection twin SVM with local information, LIWLSPTSVM)充分利用了数据样本点之间的潜在相关信息,提高了模型的泛化性能。然而,这些算法都属于有监督二分类问题[7],需要有已知的标签信息作为训练数据。有研究学者将孪生支持向量机首次扩展到无监督学习领域,成为解决数据聚类问题的经典方法之一,由此开启了一些重要的研究和应用领域的大门。与传统的基于点的聚类方法(k 均值聚类等)不同,一些基于平面的聚类方法,例如,Wang 等[8]将孪生支持向量机(twin SVM,TWSVM)扩展到聚类问题,首次提出了孪生支持向量聚类(twin support vector clustering,TWSVC),而 Bai 等[9]提出的孪生有界支持向量聚类(twin bounded SVC,TBSVC)则是在 TWSVC 的目标问题中加入正则化项对其进行改进,避免矩阵奇异性,显著提高其性能,此外,Bai 等提出的最小二乘双有界支持向量聚类(LSTBSVC)在 TBSVC 中引入了最小二乘的形式,通过求解一组线性方程,大大加快了学习的速度。近年来,一种新兴的聚类算法,基于投影轴的方法,展现出了优秀的性能表现。例如,Richharya 等[10]提出的最小二乘投影双支持向量聚类(least squares projection twin SVC,LSPTSVC)是一种基于投影的聚类算法,该算法通过求解一组线性方程,使得计算时间明显降低。尽管该算法在理论上显示出了潜力,但是由于未充分利用具有相同标签的任何一对数据点之间的潜在相关性或相似性信息,其在实际应用中仍面临着一些知识空白需要深入探索和解决。有研究学者证实数据集的大多数样本都是高度相关的,几乎有一半的数据信息存在于点之间的 k 近邻关系中,充分利用这些信息对提升算法的聚类性能至关重要,同时使得算法对不同数据集具有更好的适应性。

因此,针对 LSPTSVC 算法存在的问题,本文在现有研究的基础上,提出了一种邻域信息加权的最小二乘投影双支持向量聚类(weighted least squares projection twin SVC with neighborhood information,NIWLSPTSVC)算法,旨在改进现有算法 LSPTSVC,探索基于不同理论的聚类方法,提升算法的聚类准确性和计算效率,获得更好的聚类效果和实用意义。通过提取同类中每个数据点的相对密度,将其作为该点的相对权重,并使用加权平均值表示方程的类中心,利用凹凸迭代过程(concave-convex procedure, CCCP)[11]求解其目标函数的优化问题,获得一组投影轴,而不是传统的超平面,对数据进行聚类。当每个点的相对密度为 1 时,所提算法继承了 LSPTSVC 算法的优点。首先基于 matlab 平台将所提算法与一些经典算法在人工合成和真实世界的基准数据集上进行实验评估,然后通过统计分析检验所提算法的显著性,最后将所提算法应用到现实生活的医学数据集中进行实验分析,实验结果证实了所提算法具有一定的有效性和实用性。

1　相关理论

本节简要讨论分类算法(即 LIWLSPTSVM)和聚类算法(即 LSPTSVC)。

1.1　LIWLSPTSVM 算法

线性 LIWLSPTSVM 的原始问题简化如下:

$$\min \frac{1}{2}(\boldsymbol{A}\boldsymbol{\omega}_1-\boldsymbol{e}_1\boldsymbol{e}_1^{\mathrm{T}}\boldsymbol{E}^{(1)}\boldsymbol{A}\boldsymbol{\omega}_1)^{\mathrm{T}}\boldsymbol{D}^{(1)}(\boldsymbol{A}\boldsymbol{\omega}_1-\boldsymbol{e}_1\boldsymbol{e}_1^{\mathrm{T}}\boldsymbol{E}^{(1)}\boldsymbol{A}\boldsymbol{\omega}_1)+\frac{c_1}{2}\xi^{\mathrm{T}}\xi+\frac{c_3}{2}\|\boldsymbol{\omega}_1\|^2,$$

$$s.t.\ -(B\boldsymbol{\omega}_1-\boldsymbol{e}_2\boldsymbol{e}_1^{\mathrm{T}}\boldsymbol{E}^{(1)}\boldsymbol{A}\boldsymbol{\omega}_1)+\xi=\boldsymbol{D}^{(2)}\boldsymbol{e}_{(2)}, \tag{1}$$

$$\min \frac{1}{2}(\boldsymbol{B}\boldsymbol{\omega}_1-\boldsymbol{e}_2\boldsymbol{e}_2^{\mathrm{T}}\boldsymbol{E}^{(2)}\boldsymbol{B}\boldsymbol{\omega}_2)^{\mathrm{T}}\boldsymbol{D}^{(2)}(\boldsymbol{B}\boldsymbol{\omega}_2-\boldsymbol{e}_2\boldsymbol{e}_2^{\mathrm{T}}\boldsymbol{E}^{(2)}\boldsymbol{B}\boldsymbol{\omega}_2)+\frac{c_2}{2}\eta^{\mathrm{T}}\eta+\frac{c_4}{2}\|\boldsymbol{\omega}_2\|^2,$$

$$s.t.\ -(\boldsymbol{A}\boldsymbol{\omega}_2-\boldsymbol{e}_1\boldsymbol{e}_2^{\mathrm{T}}\boldsymbol{E}^{(1)}\boldsymbol{B}\boldsymbol{\omega}_1)+\eta=\boldsymbol{D}^{(1)}\boldsymbol{e}_{(1)}, \tag{2}$$

其中,惩罚参数 c_1、c_2、c_3、c_4 均大于 0,$\boldsymbol{A}$ 和 $\boldsymbol{B}$ 分别为正负类样本矩阵,ξ,η 是非负松弛变量,$\|\cdot\|$ 为二范数,$\boldsymbol{e}$ 为各分量全为 1 的单位向量,$\boldsymbol{D}_{(c)}=\mathrm{diag}(\boldsymbol{\rho}_1^{(c)},\cdots,\boldsymbol{\rho}_{m_{(c)}}^{(c)})$,$\boldsymbol{E}_{(c)}=\mathrm{diag}(\boldsymbol{\lambda}_1^{(c)},\cdots,\boldsymbol{\lambda}_{m_{(c)}}^{(c)})$,$c=1,2$。

进行系列求解后得到 $\boldsymbol{\omega}_1$ 如下:

$$\boldsymbol{\omega}_1=-\left(\frac{1}{c_1}(\boldsymbol{A}-\boldsymbol{e}_1\boldsymbol{e}_1^{\mathrm{T}}\boldsymbol{E}^{(1)}\boldsymbol{A})\boldsymbol{D}^{(1)}(\boldsymbol{A}-\boldsymbol{e}_1\boldsymbol{e}_1^{\mathrm{T}}\boldsymbol{E}^{(1)}\boldsymbol{A})+(\boldsymbol{B}-\boldsymbol{e}_2\boldsymbol{e}_1^{\mathrm{T}}\boldsymbol{E}^{(1)}\boldsymbol{A})^{\mathrm{T}}(\boldsymbol{B}-\boldsymbol{e}_2\boldsymbol{e}_1^{\mathrm{T}}\boldsymbol{E}^{(1)}\boldsymbol{A})+\frac{c_3}{c_1}\boldsymbol{I}\right)^{-1}$$

$$(\boldsymbol{B}-\boldsymbol{e}_2\boldsymbol{e}_1^{\mathrm{T}}\boldsymbol{E}^{(1)}\boldsymbol{A})^{\mathrm{T}}\boldsymbol{D}^{(2)}\boldsymbol{e}_2 \tag{3}$$

其中,$\boldsymbol{I}$ 是适当维度的单位矩阵。类似地,$\boldsymbol{\omega}_2$ 计算如下:

$$\boldsymbol{\omega}_2=-\left(\frac{1}{c_2}(\boldsymbol{B}-\boldsymbol{e}_2\boldsymbol{e}_2^{\mathrm{T}}\boldsymbol{E}^{(2)}\boldsymbol{B})\boldsymbol{D}^{(2)}(\boldsymbol{B}-\boldsymbol{e}_2\boldsymbol{e}_2^{\mathrm{T}}\boldsymbol{E}^{(2)}\boldsymbol{B})+(\boldsymbol{A}-\boldsymbol{e}_1\boldsymbol{e}_2^{\mathrm{T}}\boldsymbol{E}^{(2)}B)^{\mathrm{T}}(\boldsymbol{A}-\boldsymbol{e}_1\boldsymbol{e}_2^{\mathrm{T}}\boldsymbol{E}^{(2)}\boldsymbol{B})+\frac{c_4}{c_2}\boldsymbol{I}\right)^{-1}$$

$$(\boldsymbol{A}-\boldsymbol{e}_1\boldsymbol{e}_2^{\mathrm{T}}\boldsymbol{E}^{(2)}\boldsymbol{B})^{\mathrm{T}}\boldsymbol{D}^{(1)}\boldsymbol{e}_1 \tag{4}$$

对于测试样本 $\boldsymbol{x}$,类别确定如下:

$$y(\boldsymbol{x})=\underset{i=1,2}{\operatorname{argmin}}\left(\left|\boldsymbol{\omega}_i^{\mathrm{T}}\boldsymbol{x}-\boldsymbol{\omega}_i^{\mathrm{T}}\sum_{j=1}^{m_i}\boldsymbol{\lambda}_j^{(i)}\boldsymbol{x}_j^{(i)}\right|\right) \tag{5}$$

1.2　LSPTSVC 算法

线性 LSPTSVC 算法的优化问题表示如下:

$$\min_{\boldsymbol{\omega}_i^{j+1}}\frac{1}{2}(\boldsymbol{\omega}_i^{j+1})^{\mathrm{T}}\boldsymbol{S}_i\boldsymbol{\omega}_i^{j+1}+\frac{c_1}{2}\sum_{q=1}^{\overline{m}_i}(\xi_{iq}^{j+1})^2+\frac{c_2}{2}\|\boldsymbol{\omega}_i^{j+1}\|^2$$

$$s.t.\ T(|\bar{\boldsymbol{X}}_i\boldsymbol{\omega}_i^{j+1}-\frac{1}{m_i}\bar{\boldsymbol{e}}_i\boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{X}_i\boldsymbol{\omega}_i^{j+1}|)+\xi_{iq}^{j+1}=\bar{\boldsymbol{e}}_i \tag{6}$$

其中,$c_1,c_2>0$ 是惩罚参数,$\boldsymbol{\omega}_i^{j+1}$ 表示第 $j+1$ 次迭代的权重向量。$\boldsymbol{e}_i$ 和 $\bar{\boldsymbol{e}}_i$ 分别表示大小为 p 和 q 的分量全为 1 的向量。矩阵 $\boldsymbol{S}_i$ 可以表示为:

$$\boldsymbol{S}_i=\sum_{p=1}^{m_i}\left(x_p^{(i)}-\frac{1}{m_i}\sum_{p=1}^{m_i}x_p^{(i)}\right)\left(x_p^{(i)}-\frac{1}{m_i}\sum_{p=1}^{m_i}x_p^{(i)}\right)^{\mathrm{T}} \tag{7}$$

使用 CCCP 求解目标函数的优化问题,利用 $\boldsymbol{\omega}_i^j$ 的 $|\bar{\boldsymbol{X}}_i\boldsymbol{\omega}_i^j-\frac{1}{m_i}\bar{\mathbf{e}}_i\mathbf{e}_i^{\mathrm{T}}\boldsymbol{X}_i\boldsymbol{\omega}_i^j|$ 次梯度和泰勒级数展开式,目标函数可以写为:

$$L=\frac{1}{2}(\boldsymbol{\omega}_i^{j+1})^{\mathrm{T}}\boldsymbol{S}_i\boldsymbol{\omega}_i^{j+1}+\frac{c_2}{2}\|\boldsymbol{\omega}_i^{j+1}\|^2+\frac{c_1}{2}\|-\mathrm{diag}\left(\mathrm{sign}(\bar{\boldsymbol{X}}_i\boldsymbol{\omega}_i^j-\frac{1}{m_i}\bar{\boldsymbol{e}}_i\boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{X}_i\boldsymbol{\omega}_i^j)\right)\left(\bar{\boldsymbol{X}}_i\boldsymbol{\omega}_i^{j+1}-\frac{1}{m_i}\bar{\boldsymbol{e}}_i\boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{X}_i\boldsymbol{\omega}_i^{j+1}\right)+\bar{\boldsymbol{e}}_i\|^2 \tag{8}$$

通过求解式(8)关于 $\boldsymbol{\omega}_i^{j+1}$ 的梯度并令其为零,最终得到 $\boldsymbol{\omega}_i^{j+1}$ 如下:

$$\boldsymbol{\omega}_i^{j+1}=(\boldsymbol{D}_i^{\mathrm{T}}\boldsymbol{D}_i+\frac{c_2}{c_1}\boldsymbol{I}_i+\frac{\boldsymbol{S}_i}{c_1})^{-1}\boldsymbol{D}_i^{\mathrm{T}}\bar{\boldsymbol{e}}_i \tag{9}$$

其中,$\boldsymbol{D}_i=\mathrm{diag}(\mathrm{sign}(\bar{\boldsymbol{X}}_i\boldsymbol{\omega}_i^j-\frac{1}{m_i}\bar{\boldsymbol{e}}_i\boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{X}_i\boldsymbol{\omega}_i^j))(\bar{\boldsymbol{X}}_i-\frac{1}{m_i}\bar{\boldsymbol{e}}_i\boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{X}_i)$。

对于测试样本 $\boldsymbol{x}_t$,标签 $\boldsymbol{y}$ 确定如下:

$$\boldsymbol{y}(\boldsymbol{x}_t)=\underset{i=1,2,\cdots,N}{\mathrm{argmin}}\left|\boldsymbol{\omega}_i^{\mathrm{T}}\boldsymbol{x}_t-\frac{1}{m_i}\boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{X}_i\boldsymbol{\omega}_i\right| \tag{10}$$

2 NIWLSPTSVC 算法

本节给出所提算法 NIWLSPTSVC 在线性情况和非线性情况下的算法推导过程及相关分析。

2.1 线性 NIWLSPTSVC 算法

给定同类中任意一对点,权重矩阵定义如下:

$$\boldsymbol{W}_{ij}=\begin{cases}\exp(\dfrac{-\|\boldsymbol{x}_i-\boldsymbol{x}_j\|^2}{\mathrm{t}}), & \text{如果 } \boldsymbol{x}_j \text{ 是 } \boldsymbol{x}_i \text{ 的 } k \text{ 最近邻,}\\ & \text{或者 } \boldsymbol{x}_i \text{ 是 } \boldsymbol{x}_j \text{ 的 } k \text{ 最近邻}\\ 0, & \text{其他}\end{cases} \tag{11}$$

其中,t 是热核参数。

当数据在特征空间中是线性可分时,即存在一组投影轴能够将不同类别的数据样本完全分开,则线性 NIWLSPTSVC 的优化问题可以表示如下:

$$\min_{\boldsymbol{\omega}_i^{j+1}}\frac{1}{2}\sum_{p=1}^{m_i}\boldsymbol{\rho}_p^{(i)}\left((\boldsymbol{\omega}_i^{j+1})^{\mathrm{T}}\boldsymbol{x}_p^{(i)}-(\boldsymbol{\omega}_i^{j+1})^{\mathrm{T}}\sum_{p=1}^{m_i}\boldsymbol{\lambda}_p^{(i)}\boldsymbol{x}_p^{(i)}\right)^2+\frac{c_1}{2}\sum_{q=1}^{m_i}(\xi_i^{j+1})^2+\frac{c_2}{2}\|\boldsymbol{\omega}_i^{j+1}\|^2$$

$$s.t.\ \left|(\boldsymbol{\omega}_i^{j+1})^{\mathrm{T}}\bar{\boldsymbol{x}}(i)_q-(\boldsymbol{\omega}_i^{j+1})^{\mathrm{T}}\sum_{p=1}^{m_i}\boldsymbol{\lambda}_p^{(i)}\boldsymbol{x}_p^{(i)}\right|+\xi_{iq}^{j+1}=\bar{\boldsymbol{\rho}}_q^{(i)}$$

$$p=1,\cdots,m_i,\ q=1,\cdots,\bar{m}_i,i=1,\cdots,N,\ j=0,1,\cdots \tag{12}$$

其中,$c_1,c_2>0$ 是惩罚参数,$\boldsymbol{\omega}_i^{j+1}$ 表示第 $j+1$ 次迭代的投影权重向量,ξ_{iq}^{j+1} 是非负松弛变量。N 是集群的总数,簇的数据点和剩余数据点分别由 $\boldsymbol{x}_p^{(i)}$ 和 $\bar{\boldsymbol{x}}_p^{(i)}$ 表示。本文在 LSPTSVC 算法的目标函数中引入样本领域信息,即数据点 $\boldsymbol{x}_p^{(i)}$ 之间的相对密度,用 $\boldsymbol{\rho}_p^{(i)}=\sum_{q=1}^{m_i}\boldsymbol{W}_{qp}$ 表示,并将等式约束中的标准均值替换为加权均值作为类均值,更好的反映同类样本的几何结构,其中第 i 类的加权平均值表示为 $\sum_{p=1}^{m_i}\boldsymbol{\lambda}_p^{(i)}\boldsymbol{x}_p^{(i)},\boldsymbol{\lambda}_p^{(i)}=\boldsymbol{\rho}_p^{(i)}/\sum_{q=1}^{m_i}\boldsymbol{\rho}_q^{(i)}$。

式(12)的等式约束要求类 $\boldsymbol{x}_p^{(i)}$ 的中心在 $\bar{\boldsymbol{\rho}}_q^{(i)}$ 距离内,这会使得一类的中心远离其余类,有效减少异常值的影响。使用 CCCP 求解式(12)的优化问题,得到如下:

$$\min_{\boldsymbol{\omega}_i^{j+1}}\frac{1}{2}\boldsymbol{\rho}_i(\boldsymbol{\omega}_i^{j+1})^{\mathrm{T}}\boldsymbol{Z}_i\boldsymbol{\omega}_i^{j+1}+\frac{c_1}{2}\sum_{q=1}^{m_i}(\xi_{iq}^{j+1})^2+\frac{c_2}{2}\|\boldsymbol{\omega}_i^{j+1}\|^2$$

$$s.t.\ T(|\bar{\boldsymbol{X}}_i\boldsymbol{\omega}_i^{j+1}-\bar{\boldsymbol{e}}_i\boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{\lambda}_i\boldsymbol{X}_i\boldsymbol{\omega}_i^{j+1}|)+\xi_{iq}^{j+1}=\bar{\boldsymbol{\rho}}_i \tag{13}$$

其中,$\boldsymbol{e}_i$ 和 $\bar{\boldsymbol{e}}_i$ 分别表示大小为 p 和 q 的向量。矩阵 $\boldsymbol{Z}_i$ 写为:

$$\boldsymbol{Z}_i=\sum_{p=1}^{m_i}(\boldsymbol{x}_p^{(i)}-\boldsymbol{z}_i)(\boldsymbol{x}_p^{(i)}-\boldsymbol{z}_i)^{\mathrm{T}} \tag{14}$$

其中,$\boldsymbol{z}_i=\sum_{p=1}^{m_i}\boldsymbol{\lambda}_p^{(i)}\boldsymbol{x}_p^{(i)}$ 是每个聚类的中心,式(14)可以写为:

$$\boldsymbol{Z}_i=(\boldsymbol{X}_i-\boldsymbol{e}_i\boldsymbol{z}_i^{\mathrm{T}})^{\mathrm{T}}(\boldsymbol{X}_i-\boldsymbol{e}_i\boldsymbol{z}_i^{\mathrm{T}}) \tag{15}$$

将式(13)中的约束代入目标函数后写为:

$$L=\frac{c_1}{2}\|-T(|\bar{\boldsymbol{X}}_i\boldsymbol{\omega}_i^{j+1}-\bar{\boldsymbol{e}}_i\boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{\lambda}_i\boldsymbol{X}_i\boldsymbol{\omega}_i^{j+1}|)+\bar{\boldsymbol{\rho}}_i\|^2+\frac{1}{2}\boldsymbol{\rho}_i(\boldsymbol{\omega}_i^{j+1})^{\mathrm{T}}\boldsymbol{Z}_i\boldsymbol{\omega}_i^{j+1}+\frac{c_2}{2}\|\boldsymbol{\omega}_i^{j+1}\|^2 \tag{16}$$

泰勒级数展开的值通过使用 $|\bar{\mathbf{X}}_i\boldsymbol{\omega}_i^j-\bar{\mathbf{e}}_i\mathbf{e}_i^{\mathrm{T}}\boldsymbol{\lambda}_i\mathbf{X}_i\boldsymbol{\omega}_i^j|$ 关于 $\boldsymbol{\omega}_i^j$ 的次梯度可写为:

$$T(|\bar{\boldsymbol{X}}_i\boldsymbol{\omega}_i^{j+1}-\bar{\boldsymbol{e}}_i\boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{\lambda}_i\boldsymbol{X}_i\boldsymbol{\omega}_i^{j+1}|)=\mathrm{diag}(\mathrm{sign}(\bar{\boldsymbol{X}}_i\boldsymbol{\omega}_i^j-\bar{\boldsymbol{e}}_i\boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{\lambda}_i\boldsymbol{X}_i\boldsymbol{\omega}_i^j))(\bar{\boldsymbol{X}}_i\boldsymbol{\omega}_i^{j+1}-\bar{\boldsymbol{e}}_i\boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{\lambda}_i\boldsymbol{X}_i\boldsymbol{\omega}_i^{j+1}) \tag{17}$$

将式(17)代入式(16)得到如下:

$$L=\frac{1}{2}\boldsymbol{\rho}_i(\boldsymbol{\omega}_i^{j+1})^{\mathrm{T}}\boldsymbol{Z}_i\boldsymbol{\omega}_i^{j+1}+\frac{c_2}{2}\|\boldsymbol{\omega}_i^{j+1}\|^2+\frac{c_1}{2}\|-\mathrm{diag}(\mathrm{sign}(\bar{\boldsymbol{X}}_i\boldsymbol{\omega}_i^j-\bar{\boldsymbol{e}}_i\boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{\lambda}_i\boldsymbol{X}_i\boldsymbol{\omega}_i^j))(\bar{\boldsymbol{X}}_i\boldsymbol{\omega}_i^{j+1}-\bar{\boldsymbol{e}}_i\boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{\lambda}_i\boldsymbol{X}_i\boldsymbol{\omega}_i^{j+1})+\bar{\boldsymbol{\rho}}_i\|^2 \tag{18}$$

求解式(18)关于 $\boldsymbol{\omega}_i^{j+1}$ 的梯度得到如下:

$$\boldsymbol{\rho}_i\boldsymbol{Z}_i\boldsymbol{\omega}_i^{j+1}+c_2\boldsymbol{\omega}_i^{j+1}+c_1\boldsymbol{D}_i^{\mathrm{T}}(\boldsymbol{D}_i\boldsymbol{\omega}_i^{j+1}-\bar{\boldsymbol{\rho}}_i)=0 \tag{19}$$

其中,$\boldsymbol{D}_i=\mathrm{diag}(\mathrm{sign}(\bar{\boldsymbol{X}}_i\boldsymbol{\omega}_i^j-\bar{\boldsymbol{e}}_i\boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{\lambda}_i\boldsymbol{X}_i\boldsymbol{\omega}_i^j))(\bar{\boldsymbol{X}}_i-\bar{\boldsymbol{e}}_i\boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{\lambda}_i\boldsymbol{X}_i)$

对式(19)求解 $\boldsymbol{\omega}_i^{j+1}$ 得到如下:

$$\boldsymbol{\omega}_i^{j+1}=(\boldsymbol{D}_i^{\mathrm{T}}\boldsymbol{D}_i+\frac{c_2}{c_1}\boldsymbol{I}_i+\frac{1}{c_1}\boldsymbol{\rho}_i\boldsymbol{Z}_i)^{-1}\boldsymbol{D}_i^{\mathrm{T}}\bar{\boldsymbol{\rho}}_i \tag{20}$$

对于测试样本 $\boldsymbol{x}_t$,聚类标签 $\boldsymbol{y}$ 的确定如下:

$$\boldsymbol{y}(\boldsymbol{x}_t)=\underset{i=1,2,\cdots,N}{\mathrm{argmin}}|\boldsymbol{\omega}_i^{\mathrm{T}}\boldsymbol{x}_t-\boldsymbol{e}^{\mathrm{T}}\boldsymbol{\lambda}_i\boldsymbol{X}_i\boldsymbol{\omega}_i| \tag{21}$$

线性 NIWLSPTSVC 算法步骤描述如下:

算法 1 线性 NIWLSPTSVC

输入：无标签数据 $\boldsymbol{X} \in \mathbf{R}^n$；

输出：聚类标签 $\boldsymbol{Y}$ 和投影权重向量 $\boldsymbol{\omega}_i$。

步骤 1) 初始化：根据 NNG 初始化，得到初始标签 $\boldsymbol{Y}_0$；

步骤 2) 核心变量计算：计算每一类样本的相对密度 $\boldsymbol{\rho}_i$，根据 $\boldsymbol{\rho}_i$ 计算每类样本的加权平均值，计算聚类 i 的初始权重向量 $\boldsymbol{\omega}_i^0, \boldsymbol{\omega}_i^0 = Eigenvector(\boldsymbol{Z}_i)$；

步骤 3) CCCP 过程：(1)对于每个聚类 i，当 $j=0$ 时使用初始投影权重向量 $\boldsymbol{\omega}_i^0$ 计算 $\boldsymbol{\omega}_i^{j+1}$。(2)使用式(20)计算 $\boldsymbol{\omega}_i^{j+1} = \text{NIWLSPTSVC}(\boldsymbol{X}, \boldsymbol{\omega}_i^j)$。(3)如果 $(\| \boldsymbol{\omega}_i^{j+1} - \boldsymbol{\omega}_i^j \| > tol)$，则 $j=j+1$，进行步骤 3).(2)，否则，进行步骤 4)；

步骤 4) 分配聚类标签：将新标签分配给每个聚类 i 的数据点。初始 $k=0$，使用式(21)计算 $\boldsymbol{Y}_{k+1} = Df(\boldsymbol{X}, \boldsymbol{\omega}_i)$，如果 $(\| \boldsymbol{Y}_{k+1} - \boldsymbol{Y}_k \| \neq 0)$，则 $k=k+1$，进行步骤 3)，否则，算法停止，输出聚类标签 $\boldsymbol{Y}$ 和投影权重向量 $\boldsymbol{\omega}_i$。

2.2　非线性 NIWLSPTSVC 算法

当数据在特征空间中线性不可分时，即不存在一组投影轴将不同类别的数据样本完全分开，则通过使用核函数将线性可分 NIWLSPTSVC 扩展到非线性情况，非线性 NIWLSPTSVC 的优化问题可以表示如下：

$$\min_{\omega_i^{j+1}} \frac{1}{2}\boldsymbol{\rho}_i(\boldsymbol{\omega}_i^{j+1})^{\mathrm{T}}\boldsymbol{S}_i\boldsymbol{\omega}_i^{j+1} + \frac{c_1}{2}\sum_{q=1}^{m_i}(\xi_{iq}^{j+1})^2 + \frac{c_2}{2}\| \boldsymbol{\omega}_i^{j+1} \|^2$$

$$s.t.\ T(|K(\bar{\boldsymbol{X}}_i, \boldsymbol{M}^{\mathrm{T}})\boldsymbol{\omega}_i^{j+1} - \bar{\boldsymbol{e}}_i\boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{\lambda}_i K(\boldsymbol{X}_i, \boldsymbol{M}^{\mathrm{T}})\boldsymbol{\omega}_i^{j+1}|) + \xi_{iq}^{j+1} = \bar{\boldsymbol{\rho}}_i \tag{22}$$

其中，$K(\cdot, \boldsymbol{M}^{\mathrm{T}})$ 是内核函数，并且 $\boldsymbol{M} = [\boldsymbol{X}_1, \boldsymbol{X}_2, \cdots, \boldsymbol{X}_N]$，矩阵 $\boldsymbol{S}_i$ 可以写为：

$$\boldsymbol{S}_i = \sum_{p=1}^{m_i}(K(\boldsymbol{x}_p^{(i)}, \boldsymbol{M}^{\mathrm{T}}) - \boldsymbol{s}_i)(K(\boldsymbol{x}_p^{(i)}, \boldsymbol{M}^{\mathrm{T}}) - \boldsymbol{s}_i)^{\mathrm{T}} \tag{23}$$

其中，$\boldsymbol{s}_i = \sum_{p=1}^{m_i}\boldsymbol{\lambda}_p^{(i)}K(\boldsymbol{x}_p^{(i)}, \boldsymbol{M}^{\mathrm{T}})$。式(23)可以写为：

$$\boldsymbol{S} = (K(\boldsymbol{x}_i, \boldsymbol{M}^{\mathrm{T}}) - \boldsymbol{e}_i\boldsymbol{s}_i^{\mathrm{T}})^{\mathrm{T}}(K(\boldsymbol{x}_i, \boldsymbol{M}^{\mathrm{T}}) - \boldsymbol{e}_i\boldsymbol{s}_i^{\mathrm{T}}) \tag{24}$$

将式(22)中的约束代入目标函数，使用 CCCP 迭代得到如下：

$$L = \frac{c_1}{2}\| -\mathrm{diag}(\mathrm{sign}(K(\bar{\boldsymbol{X}}_i, \boldsymbol{M}^{\mathrm{T}})\boldsymbol{\omega}_i^j)) - \bar{\boldsymbol{e}}_i\boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{\lambda}_i K(\boldsymbol{X}_i, \boldsymbol{M}^{\mathrm{T}})\boldsymbol{\omega}_i^j))(K(\bar{\boldsymbol{X}}_i, \boldsymbol{M}^{\mathrm{T}})\boldsymbol{\omega}_i^{j+1} - \bar{\boldsymbol{e}}_i\boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{\lambda}_i K(\boldsymbol{X}_i, \boldsymbol{M}^{\mathrm{T}})\boldsymbol{\omega}_i^{j+1}) + \bar{\boldsymbol{\rho}}_i\|^2 + \frac{1}{2}\boldsymbol{\rho}_i(\boldsymbol{\omega}_i^{j+1})^{\mathrm{T}}\boldsymbol{S}_i\boldsymbol{\omega}_i^{j+1} + \frac{c_2}{2}\| \boldsymbol{\omega}_i^{j+1} \|^2 \tag{25}$$

求解式(25)对于 $\boldsymbol{\omega}_i^{j+1}$ 的梯度得到如下：

$$\boldsymbol{\rho}_i\boldsymbol{S}_i\boldsymbol{\omega}_i^{j+1} + c_2\boldsymbol{\omega}_i^{j+1} + c_1\boldsymbol{G}_i^{\mathrm{T}}(\boldsymbol{G}_i\boldsymbol{\omega}_i^{j+1} - \bar{\boldsymbol{\rho}}_i) = 0 \tag{26}$$

其中，$\boldsymbol{G}_i = \mathrm{diag}(\mathrm{sign}(K(\bar{\boldsymbol{X}}_i, \boldsymbol{M}^{\mathrm{T}})\boldsymbol{\omega}_i^j - \bar{\boldsymbol{e}}_i\boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{\lambda}_i K(\boldsymbol{X}_i, \boldsymbol{M}^{\mathrm{T}})\boldsymbol{\omega}_i^j))(K(\bar{\boldsymbol{X}}_i, \boldsymbol{M}^{\mathrm{T}}) - \bar{\boldsymbol{e}}_i\boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{\lambda}_i K(\boldsymbol{X}_i, \boldsymbol{M}^{\mathrm{T}}))$

对式(26)求解 $\boldsymbol{\omega}_i^{j+1}$，得到如下：

$$\boldsymbol{\omega}_i^{j+1} = \left(\boldsymbol{G}_i^{\mathrm{T}}\boldsymbol{G}_i + \frac{c_2}{c_1}\boldsymbol{I}_i + \frac{1}{c_1}\boldsymbol{\rho}_i\boldsymbol{S}_i\right)\boldsymbol{G}_i^{\mathrm{T}}\bar{\boldsymbol{\rho}}_i \tag{27}$$

式(27)涉及阶数为 $m \times m$ 的矩阵求逆。为降低矩阵逆的计算时间，本文使用了(sherman-sorrison-woodbury，SMW)公式[12]。式(27)可以写为：

$$\boldsymbol{\omega}_i^{j+1} = (\boldsymbol{G}_i^{\mathrm{T}}\boldsymbol{G}_i + \frac{c_2}{c_1}\boldsymbol{I}_i + \frac{\boldsymbol{\rho}_i\boldsymbol{U}_i^{\mathrm{T}}\boldsymbol{U}_i}{c_1})^{-1}\boldsymbol{G}_i^{\mathrm{T}}\bar{\boldsymbol{\rho}}_i \tag{28}$$

其中，$\boldsymbol{U} = (K(\boldsymbol{X}_i, \boldsymbol{M}^{\mathrm{T}}) - \boldsymbol{e}_i\boldsymbol{s}_i^{\mathrm{T}})$。在式(28)中使用 SMW 公式，得到如下：

$$\boldsymbol{\omega}_i^{j+1} = (\boldsymbol{A}_i^{-1} - \boldsymbol{A}_i^{-1}\boldsymbol{G}_i^{\mathrm{T}}((\boldsymbol{I}_i + \boldsymbol{G}_i\boldsymbol{A}_i^{-1}\boldsymbol{G}_i^{\mathrm{T}})^{-1}\boldsymbol{G}_i\boldsymbol{A}_i^{-1}))\boldsymbol{G}_i^{\mathrm{T}}\bar{\boldsymbol{\rho}}_i \tag{29}$$

其中，$\boldsymbol{A}_i^{-1} = \frac{c_1}{c_2}(\boldsymbol{I}_i - \boldsymbol{\rho}_i\boldsymbol{U}_i^{\mathrm{T}}(c_2\boldsymbol{I}_i + \boldsymbol{U}_i\boldsymbol{\rho}_i\boldsymbol{U}_i^{\mathrm{T}})^{-1}\boldsymbol{U}_i)$。为计算 $\boldsymbol{\omega}_i^{j+1}$，需计算一个大小为 $(m_i \times m_i)$ 的逆和大小为 $(m - m_i) \times (m - m_i)$ 的其他逆，而不用计算大小为 $(m \times m)$ 的逆。

对于某个测试样本 $\boldsymbol{x}_t$，聚类标签 $\boldsymbol{y}$ 的确定如下：

$$\boldsymbol{y}(\boldsymbol{x}_t) = \underset{i=1,2,\cdots,N}{\mathrm{argmin}} |\boldsymbol{\omega}_i^{\mathrm{T}}K(\boldsymbol{x}_t, \boldsymbol{M}^{\mathrm{T}}) - \boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{\lambda}_i K(\boldsymbol{X}_i, \boldsymbol{M}^{\mathrm{T}})\boldsymbol{\omega}_i| \tag{30}$$

引理 2.1. 设 $\boldsymbol{X} \in \mathrm{R}^{m \times n}, \boldsymbol{Z} \in \mathrm{R}^{n \times n}, m > n$，那么 $\boldsymbol{Z} = (\boldsymbol{X} - \boldsymbol{e}\boldsymbol{z})^{\mathrm{T}}(\boldsymbol{X} - \boldsymbol{e}\boldsymbol{z})$ 是一个正半定矩阵，其中 $\boldsymbol{z} = \sum_{p=1}^{m}\boldsymbol{\lambda}_p\boldsymbol{x}_p$。

定理 2.1. 设 $\boldsymbol{X} \in \mathrm{R}^{p \times n}, m > n$，且 $\boldsymbol{Z} = (\boldsymbol{X} - \boldsymbol{e}\boldsymbol{z})^{\mathrm{T}}(\boldsymbol{X} - \boldsymbol{e}\boldsymbol{z})$，那么全局最小值：

$$\min_{\boldsymbol{\omega}} \boldsymbol{\omega}^{\mathrm{T}}\boldsymbol{Z}\boldsymbol{\omega} \quad s.t.\ \boldsymbol{\omega}^{\mathrm{T}}\boldsymbol{\omega} = 1, \tag{31}$$

由具有最小特征值 $\boldsymbol{Z}$ 的任意特征向量 $\boldsymbol{\omega}$ 获得。式(31)的最小值是正定的，如果 $\boldsymbol{S}$ 是正定的或者 $Rank(\boldsymbol{X} - \boldsymbol{e}\boldsymbol{z}) = n$。

注 1. 如果 $\boldsymbol{Z}$ 是正定的，那么 $\boldsymbol{Z}$ 是非奇异的。

定理 2.2. 设 $\boldsymbol{X} \in \mathrm{R}^{p \times n}, \boldsymbol{Z} \in \mathrm{R}^{q \times n}, \boldsymbol{Z} = (\boldsymbol{X} - \boldsymbol{e}\boldsymbol{z})^{\mathrm{T}}(\boldsymbol{X} - \boldsymbol{e}\boldsymbol{z}) \in \mathbf{R}^{n \times n}, \boldsymbol{I} \in \mathbf{R}^{n \times n}$ 是一个单位矩阵，那么对任意 $c_1, c_2 > 0$，矩阵 $\boldsymbol{D}^{\mathrm{T}}\boldsymbol{D} + \frac{c_2}{c_1}\boldsymbol{I} + \frac{\boldsymbol{\rho}\boldsymbol{Z}}{c_1}$ 是可逆的。

注：以上引理和定理的详细证明过程见文献[10]。

非线性 NIWLSPTSVC 算法步骤描述如下：

算法 2 非线性 NIWLSPTSVC

输入：无标签数据 $\boldsymbol{X} \in \mathbf{R}^n$，并使用内核函数计算内核矩阵 $\boldsymbol{K}$；

输出：聚类标签 $\boldsymbol{Y}$ 和投影权重向量 $\boldsymbol{\omega}_i$。

步骤 1) 初始化：根据 NNG 初始化，得到初始标签 $\boldsymbol{Y}_0$；

步骤 2) 核心变量计算，计算每一类样本的相对密度 $\boldsymbol{\rho}_i$。根据 $\boldsymbol{\rho}_i$ 计算每类样本的加权平均值。计算聚类 i 的初始投影权重向量 $\boldsymbol{\omega}_i^0, \boldsymbol{\omega}_i^0 = Eigenvector(\boldsymbol{S}_i)$；

步骤 3) CCCP 过程：(1)对于每个聚类 i，当 $j=0$ 时使用初始投影权重向量 $\boldsymbol{\omega}_i^0$ 计算 $\boldsymbol{\omega}_i^{j+1}$。(2)使用式(28)计算 $\boldsymbol{\omega}_i^{j+1} = \text{NIWLSPTSVC}(\boldsymbol{K}, \boldsymbol{\omega}_i^j)$。(3)如果 $(\| \boldsymbol{\omega}_i^{j+1} - \boldsymbol{\omega}_i^j \| > tol)$，则 $j=j+1$，进行步骤 3).(2)，否则，进行步骤 4)；

步骤 4) 分配聚类标签:将新标签分配给每个聚类 i 的数据点。初始 $k=0$,使用式(30)计算 $\boldsymbol{Y}_{k+1}=Df(K,\boldsymbol{\omega}_i)$。如果 $(\|\boldsymbol{Y}_{k+1}-\boldsymbol{Y}_k\|\neq 0)$,则 $k=k+1$,进行步骤 3),否则,算法停止,输出聚类标签 $\boldsymbol{Y}$ 和投影权重向量 $\boldsymbol{\omega}_i$。

2.3 时间复杂度

聚类算法的计算成本主要来自于解决优化问题,因此本节使用 big-O 表示法[13],着重分析本文算法求解优化问题的时间复杂度。TWSVC 算法通过使用 QPP 来求解聚类问题,其计算时间主要由求解 QPP 和矩阵的逆组成。在样本大小为 $m=m_i+\bar{m}_i$,类别数为 N 的情况下,线性 TWSVC 的时间复杂度为 $O(N(\bar{m}_i^3+n^3))$。非线性 TWSVC 的时间复杂度为 $O(\bar{m}3_i+m^3)$,TBSVC 的时间复杂度与 TWSVC 相同。线性 LSPTSVC 求解的是一组线性方程。对大小为 $n\times n$ 的 N 个矩阵求逆,其时间复杂度为 $O(Nn^3)$。非线性 LSPTSVC 需要计算大小为 m_i 和 $\bar{m}_i$ 的 N 个矩阵逆,其时间复杂度为 $O(m_i^3+\bar{m}_i^3)$。LSTBSVC 与 LSPTSVC 具有相似的时间复杂度。本文所提算法求解的也是一组线性方程,其时间复杂性讨论如下:

1)k 最近邻图构造:用于所有 m 个数据点的 k 最近邻图查找[14],以计算权重矩阵的时间复杂度约为 $O(2m_i^2\log(m_i))$。

2)线性 NIWLSPTSVC 需要对大小为 $n\times n$ 的 N 个矩阵求逆,在上文式(20)的时间复杂度表示为 $O(Nn^3)$。

3)非线性 NIWLSPTSVC 需要计算大小为 m_i 和 $\bar{m}_i$ 的 N 个矩阵求逆,因此,时间复杂度为 $O(m_i^3+\bar{m}_i^3)$。

3 实验结果

本节将所提算法 NIWLSPTSVC 与 TWSVC、TBSVC、LSTBSVC 和 LSPTSVC 算法的性能进行比较。共使用 16 个基准数据集进行实验分析,来评估所提算法在线性和非线性情况下的有效性。人工合成数据集来自①开源数据集,真实世界数据集来自 UCI 存储库②。实验使用数据集的详细信息如表 1 所示。

表 1 实验数据集的详细信息

Table 1 Details of the experimental datasets

数据集	样本数	属性数	类别数	类别分布
Ds2c2sc13	588	2	13	108/24/27/73/32/49/33/40/28/24/32/71/47
Pathbased	300	2	3	110/97/93
3MC	400	2	3	120/170/110
2d-4c-no4	863	2	4	421/86/294
Compound	399	2	6	50/92/38/45/158/16
2d-4c-no9	876	2	4	244/312/101/219
Zelnik3	266	2	3	73/75/118
Glass	214	9	6	70/13/76/29/17/9
Haberman	306	3	2	225/81
Zelnik1	299	2	3	139/99/61
Aggregation	788	2	7	45/170/102/273/34/130/34
Balance-scale	625	4	3	288/288/49
Hayes-roth	132	5	3	51/51/30
R15	600	2	15	40/40/40/40/40/40/40/40/40/40/40/40/40/40/40
Longsquare	900	2	6	155/150/148/150/150/147
Tae	150	5	3	49/50/52

3.1 实验设置

本文的实验过程在 64 位 Windows10 操作系统,Intel(R)Core(TM)i5-4590CPU@3.30 sGHZ CPU 8 GB RAM 的处理器,MATLAB R2020a 环境的 PC 上运行。在非线性情况下,本文所有算法都使用了高斯核。

在实验过程中,总数据样本的 50%用于训练,其余用于测试。惩罚参数 c_1,c_2 从集合$\{10^{-5},10^{-4},\cdots,10^5\}$中选择,内核参数 mu 从集合$\{2^{-5},2^{-4},\cdots,2^5\}$中选择,而热核参数 t 从集合$\{2^{-8},2^{-7},\cdots,2^8\}$中选择,邻域大小 k 在 1 到 9 的范围内选择。最后,对于所有算法,CCCP 过程的容差值都设置为 0.001。

使用如下相似性矩阵来度量具有聚类标签 $\boldsymbol{y}$ 的 m 个数据样本的聚类精度,

$$\boldsymbol{P}(i,j)=\begin{cases}1, & \text{if } y_i=y_j\\ 0, & \text{其他}\end{cases} \tag{32}$$

设 $\boldsymbol{P}_t$ 是预测聚类标签的相似性矩阵,$\boldsymbol{P}_a$ 是实际标签的相似矩阵。精度定义如下:

$$\text{accuracy}=\frac{n_0+n_1-m}{m^2-m}\times 100\% \tag{33}$$

其中,n_0 是 $\boldsymbol{P}_a$ 和 $\boldsymbol{P}_t$ 中的 0 的数目,n_1 是 $\boldsymbol{P}_a$ 和 $\boldsymbol{P}_t$ 中的 1 的数目。

3.2 实验结果分析

线性算法(即 TWSVC、TBSVC、LSTBSVC、LSPTSVC 和 NIWLSPTSVC)和非线性算法(即 TWSVC、TBSVC、LSTBSVC、LSPTSVC 和 NIWLSPTSVC)的聚类精度结果如表 2 和表 3 所示。其中 R15 数据集,Longsquare 数据集和 Tae 数据集是平衡的,其他数据集都是非平衡的。

从表 2 可以看到,在线性情况下,所提算法在 13 个数据集上获得最高精度,例如对于 Compound 数据集,所提算法的聚类准确率为 85.28%,而 TWSVC 为 79.41%、TBSVC

① https://github.com/deric/clustering-benchmark

② https://archive.ics.uci.edu

表 2　线性算法聚类精度比较

Table 2　Comparison of clustering accuracy of linear algorithms　%

数据集	TWSVC	TBSVC	LSTBSVC	LSPTSVC	所提算法
Ds2c2sc13	**95.32**	92.90	93.35	93.27	93.27
Pathbased	65.35	**72.24**	**72.24**	67.61	71.65
3MC	93.89	97.87	88.94	95.18	**98.57**
2d-4c-no4	77.53	86.42	85.38	97.68	**99.19**
Compound	79.41	77.52	75.08	83.46	**85.28**
2d-4c-no9	95.79	93.96	97.91	**98.63**	95.17
Zelnik3	76.87	78.89	78.89	77.80	**93.14**
Glass	65.56	65.69	67.06	67.89	**71.75**
Haberman	63.09	51.24	63.78	63.78	**64.48**
Zelnik1	54.85	55.91	54.86	53.33	**63.15**
Aggregation	80.40	80.33	80.78	84.68	**87.40**
Balance-scale	66.98	70.20	70.20	74.24	**77.33**
Hayes-roth	55.99	58.97	58.41	63.82	**66.99**
R15	96.64	93.61	94.17	97.36	**97.71**
Longsquare	81.89	88.78	88.59	90.13	**93.65**
Tae	50.13	54.63	47.28	54.70	**59.28**
平均精度	74.98	76.20	76.06	78.97	**82.38**

注：加粗字体为对应算法精度的最优值，以%为单位。

表 3　非线性算法聚类精度比较

Table 3　Comparison of clustering accuracy of nonlinear algorithms　%

数据集	TWSVC	TBSVC	LSTBSVC	LSPTSVC	所提算法
Ds2c2sc13	93.94	**95.54**	94.60	92.18	92.18
Pathbased	59.54	68.65	73.53	84.48	**91.34**
3MC	80.59	75.84	88.07	93.57	**98.57**
2d-4c-no4	58.14	63.86	62.88	89.91	**97.73**
Compound	91.74	91.73	**93.04**	92.70	90.22
2d-4c-no9	96.36	95.70	96.05	90.00	96.76
Zelnik3	98.95	97.90	98.94	**100.00**	**100.00**
Glass	64.72	67.62	68.91	68.51	**72.97**
Haberman	61.76	**64.48**	63.09	**64.48**	**64.48**
Zelnik1	64.30	67.88	68.46	**100.00**	**100.00**
Aggregation	79.88	77.04	76.60	71.39	**89.78**
Balance-scale	67.10	66.41	57.50	70.39	**72.13**
Hayes-roth	33.29	33.29	52.91	**58.23**	56.13
R15	**99.51**	99.49	99.00	96.44	98.35
Longsquare	77.14	85.04	81.02	95.36	**95.74**
Tae	46.99	49.33	52.36	55.35	**56.18**
平均精度	73.37	74.99	76.68	82.69	**85.78**

注：加粗字体为对应算法精度的最优值，以%为单位。

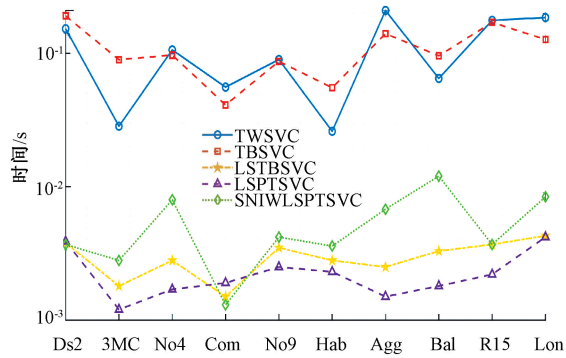
为 77.52%、LSTBSVC 为 75.08%、LSPTSVC 为 83.46%。所提算法的平均精度也达到最高，即 82.38%，分别比线性 TWSVC、TBSVC、LSTBSVC 和 LSPTSVC 算法高 7.40%、6.18%、6.32%和 3.41%。从表 3 可以看到，非线性情况下，所提算法在 12 个数据集中获得最高精度。例如对于 Pathbased 数据集，所提算法的聚类准确率为

91.34%,而 TWSVC 为 59.54%、TBSVC 为 68.65%、LSTBSVC 为 73.53%、LSPTSVC 为 84.48%。从平均聚类精度也可以看出这一点,所提算法的平均精度最高,即 85.78%,分别比非线性 TWSVC、TBSVC、LSTBSVC 和 LSPTSVC 算法高 12.41%、10.79%、9.10% 和 3.09%。由线性情况和非线性情况的实验结果可知,样本邻域信息的引入,使得所提算法 NIWLSPTSVC 的聚类精度得到有效地提升。线性 NIWLSPTSVC 和非线性 NIWLSPTSVC 在大多数数据集上都获得了较高的聚类精度,在平衡数据集和非平衡数据集上均表现出明显的优势,在区分具有不同类别之间的不平衡数据集和平衡数据集时都能够保持较高的准确率,这也进一步说明所提算法具有较好的泛化能力和鲁棒性能。

从表 1 选取十个时间变化较为明显的数据集进行比较。在十个数据集上的平均学习时间如图 1 所示。图 1 直观地显示所有算法在十个数据集上的平均学习时间。如图 1(a)所示,线性 NIWLSPTSVC 算法的平均学习时间明显小于线性 TWSVC 和 TBSVC 算法。这是因为这两种聚类算法都需要通过求解 QPP 来获得聚类超平面,而所提算法则通过求解线性方程组获得一组投影轴对样本进行聚类。与 LSTBSVC 和 LSPTSVC 相比,所提算法需要构造类内权重矩阵来计算同类中每一个数据点的相对密度 ρ,与 LSPTSVC 算法一样需要计算权重向量 ω,但不需要计算偏差 b,所以 NIWLSPTSVC 算法的计算比 LSTBSVC 和 LSPTSVC 算法略耗时。如图 1(b)所示,非线性 NIWLSPTSVC 算法的平均学习时间明显小于非线性 TWSVC 和 TBSVC 算法,与非线性 LSTBSVC 和 LSPTSVC 算法相差不大,甚至在有些数据集上,所提算法的平均学习时间达到最低,例如,3 MC 数据集和 Longsquare 数据集。以上分析都证明在处理多类别数据的聚类问题时,所提算法是有效的。

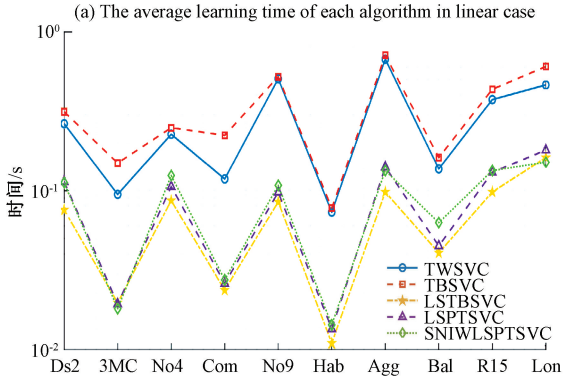
使用线性 NIWLSPTSVC 算法和其他算法对 Zelnik3 数据集进行聚类,结果如图 2 所示。该数据集的实际聚类如图 2(a)所示,线性 TWSVC 算法的聚类效果如图 2(b)所示,线性 TBSVC 算法的聚类效果如图 2(c)所示,线性 LSTBSVC 算法的聚类效果如图 2(d)所示,线性 LSPTSVC 算法的聚类效果如图 2(e)所示,如图 2(f)所示是所提算法标记的聚类效果图,其聚类精度达到最高,为 93.14%,与图 2(a)中的原始聚类很相似,只有少数数据点被误分,绝大多数数据点都可以被正确划分到自己的聚类类别,与其他算法的聚类结果相比,可以直观地看到所提算法具有较好的聚类质量。线性情况下的聚类效果图进一步验证本文所提算法具有优良的性能。

使用非线性 NIWLSPTSVC 算法和其他算法对 3 MC 数据集进行聚类,结果如图 3 所示。该数据集的实际聚类如图 3(a)所示。非线性 TWSVC 算法的聚类效果如图 3(b)所示,非线性 TBSVC 算法的聚类效果如图 3(c)所示,非线性 LSTBSVC 算法的聚类效果如图 3(d)所示,非线性 LSPTSVC 算法的聚类效果如图 3(e)所示,如图 3(f)所示是所提算法标记的聚类效果图,其聚类精度达到最高,为 98.57%,与图 3(a)中的原始聚类很相似,仅仅只有一个数据点被误分了,与其他算法的聚类结果相比,可以直观地看到所提算法具有清晰准确的聚类效果。非线性情况下的聚类效果图进一步验证本文所提算法具有优良的性能。



(a) 各算法在线性情况下的平均学习时间

(a) The average learning time of each algorithm in linear case



(b) 各算法在非线性情况下的平均学习时间

(b) The average learning time of each algorithm in nonlinear case

图 1 各算法在 10 个数据集下的平均学习时间

Fig. 1 Average learning time of each algorithm in 10 datasets

3.3 统计分析

为验证两种方法之间的显著差异,本文使用 Friedman 检验和 Nemenyi 检验[15]来验证 NIWLSPTSVC 算法与 TWSVC、TBSVC、LSTBSVC 和 LSPTSVC 算法之间的统计显著性。首先根据聚类精度计算每种算法在每一个数据集上的秩排名,秩排名及各算法的平均秩的计算结果分别如表 4 和表 5 所示。最初,假设各方法之间没有差异,并将其作为零假设。

1)线性情况

通过表 4 中的平均秩计算 Friedman 检验的 χ_F^2 值如下

$$\chi_F^2 = \frac{12N}{k(k+1)}\left[\sum_{i=1}^{k} R_i^2 - \frac{k(k+1)^2}{4}\right] \tag{34}$$

$$F_F = \frac{(N-1)\chi_F^2}{N(k-1)-\chi_F^2} \tag{35}$$

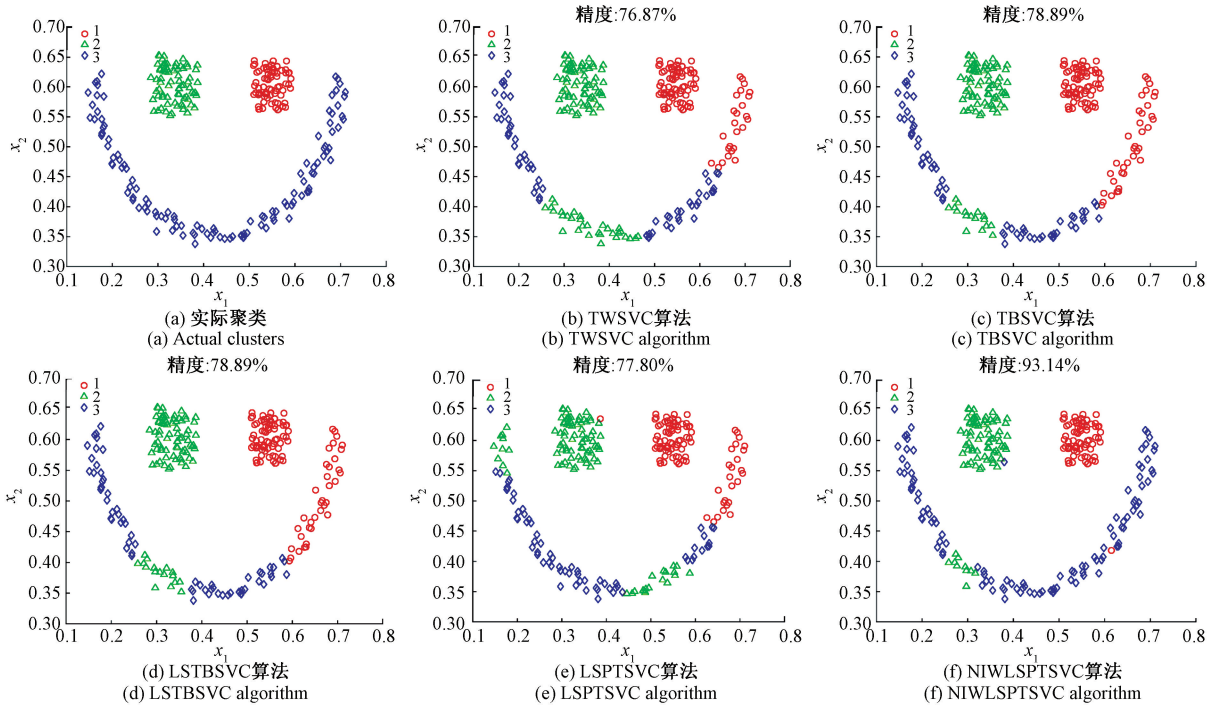


图 2　线性算法在 Zelnik3 数据集的聚类效果图

Fig. 2　Clustering effect of linear algorithm in Zelnik3 dataset

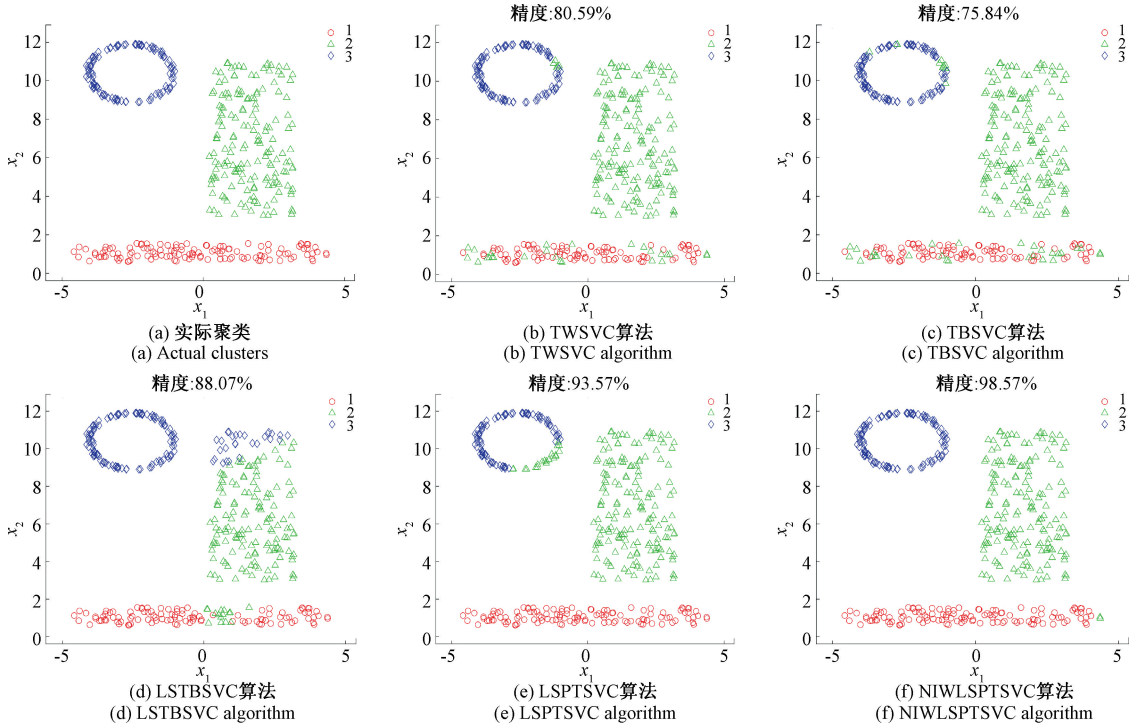


图 3　非线性算法在 3MC 数据集的聚类效果图

Fig. 3　Clustering effect of nonlinear algorithm in 3MC dataset

其中，N 是所使用的数据集的数量，k 表示算法的数量，R_i，是第 i 种算法在 N 个数据集上的平均秩。由表 4 可以看到线性 TWSVC、TBSVC、LSTBSVC、LSPTSVC 和 NIWLSPTSVC 算法的平均秩分别为 4.06、3.53、3.38、

表 4 线性算法在所有数据集上的秩排名

Table 4 Rank ranking of linear algorithms on all datasets

数据集	TWSVC	TBSVC	LSTBSVC	LSPTSVC	所提算法
Ds2c2sc13	1	5	2	3.5	3.5
Pathbased	5	1.5	1.5	4	3
3MC	4	2	5	3	1
2d-4c-no4	5	3	4	2	1
Compound	3	4	5	2	1
2d-4c-no9	3	5	2	1	4
Zelnik3	5	2.5	2.5	4	1
Glass	5	4	3	2	1
Haberman	4	5	2.5	2.5	1
Zelnik1	4	2	3	5	1
Aggregation	4	5	3	2	1
Balance-scale	5	3.5	3.5	2	1
Hayes-roth	5	3	4	2	1
R15	3	5	4	2	1
Longsquare	5	3	4	2	1
Tae	4	3	5	2	1
平均秩	4.06	3.53	3.38	2.56	1.47

表 5 非线性算法在所有数据集上的秩排名

Table 5 Rank ranking of nonlinear algorithms on all datasets

数据集	TWSVC	TBSVC	LSTBSVC	LSPTSVC	所提算法
Ds2c2sc13	3	1	2	4.5	4.5
Pathbased	5	4	3	2	1
3MC	4	5	3	2	1
2d-4c-no4	5	3	4	2	1
Compound	3	4	1	2	5
2d-4c-no9	2	4	3	5	1
Zelnik3	3	5	4	1.5	1.5
Glass	5	4	2	3	1
Haberman	5	2	4	2	2
Zelnik1	5	4	3	1.5	1.5
Aggregation	2	3	4	5	1
Balance-scale	3	4	5	2	1
Hayes-roth	4.5	4.5	3	1	2
R15	1	2	3	5	4
Longsquare	5	3	4	2	1
Tae	5	4	3	2	1
平均秩	3.78	3.53	3.19	2.66	1.84

2.56 和 1.47,线性情况下,所提算法获得了最低平均秩排名。将对应值代入方程(34)和(35)中,本文得到 $\chi_F^2 \approx 26.133\ 8$,$F_F \approx 10.352\ 4$。这里 F 分布自由度为(5−1,(5−1)(16−1))=(4,60)。在 $\alpha = 0.05$ 的显著性水平下,临界值 F(4,60)=2.525,$F_F = 10.352\ 4 > 2.525$,拒绝零假设,Friedman 检验通过。

为了检验所提算法和其他算法之间的成对统计差异,使用 Nemenyi 检验计算如下:

$$CD = t_\alpha \sqrt{\frac{k(k+1)}{6N}} \tag{36}$$

其中，t_α 是 α 显著性水平的临界值，CD 是 N 个数据集上的 k 算法的临界差。对于 $k=5$，$N=16$，在 $\alpha=0.1$ 的显著性水平下，得到 $t_\alpha=2.459$，将其代入式(36)，通过计算得到 $CD=1.374\,6$。如果两种算法的平均秩差至少相差 CD，则两种方法之间存在显著差异。线性 TWSVC、TBSVC、LSTBSVC 和 LSPTSVC 算法与线性 NIWLSPTSVC 算法之间的平均秩之差分别为：4.06－1.47＝2.59，3.53－1.47＝2.06，3.38－1.47＝1.91，2.61－1.47＝1.09，可以看到，线性 NIWLSPTSVC 与线性 TWSVC、TBSVC 和 LSTBSVC 之间存在显著差异，而与线性 LSPTSVC 之间未存在显著差异，表明线性 NIWLSPTSVC 算法的性能优于线性 TWSVC、TBSVC 和 LSTBSVC 算法，而与线性 LSPTSVC 算法之间的差异性不显著。

2)非线性情况

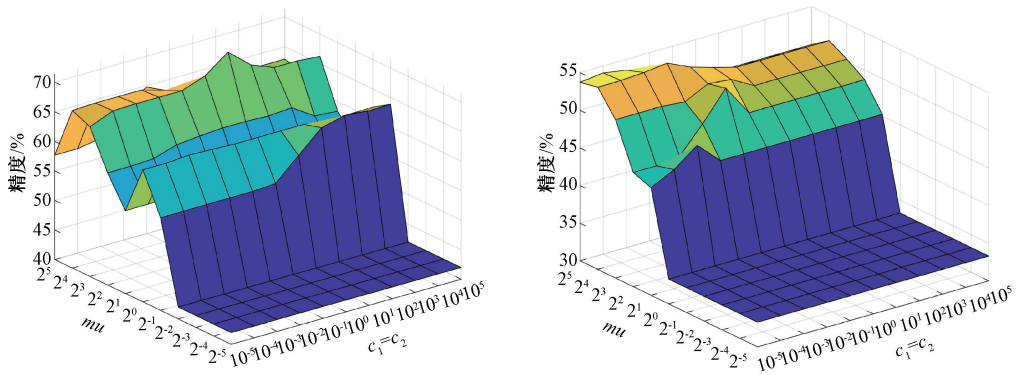
由表 5 可以看到非线性 TWSVC、TBSVC、LSTBSVC、LSPTSVC 和 NIWLSPTSVC 算法的平均秩分别为 3.78、3.53、3.19、2.66 和 1.84，非线性情况下，所提算法获得了最低平均秩排名。与线性情况类似，使用同样的方法，根据表 5 中每一种算法的平均秩计算 χ^2 值和 F_F，将数据代入式(32)和式(33)中得到：$\chi_F^2 \approx 15.2742$，$F_F \approx 4.7021 > 2.525$，拒绝零假设，Friedman 检验通过。进行 Nemenyi 检验，非线性 TWSVC、TBSVC、LSTBSVC 和 LSPTSVC 算法与非线性 NIWLSPTSVC 算法之间的平均秩差分别为：3.78－1.84＝1.94，3.53－1.84＝1.69，3.19－1.84＝1.35，2.66－1.84＝0.82，因此，非线性 NIWLSPTSVC 算法与非线性 TWSVC 和 TBSVC 之间存在显著的统计差异，与非线性 LSTBSVC 和 LSPTSVC 之间未存在显著的统计差异，认为非线性 NIWLSPTSVC 算法的性能优于非线性 TWSVC 和 TBSVC 算法，但由于非线性 NIWLSPTSVC 算法与非线性 LSTBSVC 和 LSPTSVC 算法之间的平均秩差均小于 CD 值，所以非线性 NIWLSPTSVC 算法与非线性 LSTBSVC 和 LSPTSVC 算法之间的差异不够显著。

3.4　参数敏感性分析

为进一步分析参数对所提算法性能的影响，本文对参数 c_1，c_2，mu 和 k 进行了敏感性分析。

1)参数 c_1，c_2 和 mu 对所提算法性能的影响

图 4 评估了非线性 NIWLSPTSVC 算法对参数 c_1，c_2 的敏感性，为了简便，设置 $c_1=c_2$。由图 4(a)可以看到，mu 的值偏大时，聚类性能对 c_1，c_2 的灵敏度较高，改变参数 c_1 和 c_2 会得到更高的聚类精度。如图 4(b)所示，mu 的值越大，改变参数 c_1 和 c_2 会得到更好的聚类精度。这是由于在非线性情况下，核映射起着非常重要的作用，参数 mu 决定核函数的值，导致数据的非线性变换。此外，从三维图中还可以看到，参数 $c_1=c_2$ 时对聚类精度并没有显著的影响。因此，为了获得更好的泛化性能，需要仔细选择所提算法的参数。



(a) Balance-scale数据集　(b) Hayes-roth数据集
(a) Balance-scale dataset　(b) Hayes-roth dataset

图 4　参数 c_1，c_2 和 mu 对非线性 NIWLSPTSVC 算法聚类性能的影响

Fig. 4　Effects of parameters c_1，c_2 and mu on the clustering performance of the nonlinear NIWLSPTSVC algorithm

2)参数 k 对所提算法性能的影响

图 5 分析了参数 k 对线性 NIWLSPTSVC 算法性能的影响。如图 5(a)所示，k 的值在 1 至 4 范围内时，聚类精度最低，当 k 的值大于 4 时，聚类精度急剧增加，在 k 值等于 5 至 7 时，聚类精度达到最高，聚类性能最好。如图 5(b)所示，聚类性能随着 k 值的增加而逐渐增长，当 k 等于 9 时，聚类精度达到顶峰。如图 5(c)所示，聚类性能随着 k 值的增加而波动变化，在 k 的值等于 2 时，聚类精度达到最大，而后又继续波动变化。如图 5(d)所示，聚类性能在 k 的不同水平下变化，当 k 的值等于 3 时，聚类精度达到最大，而后逐渐降低，当 k 等于 9 时，聚类精度达到了最低。因此，为了本文所提算法具有更好的性能，需要仔细选择参数 k。

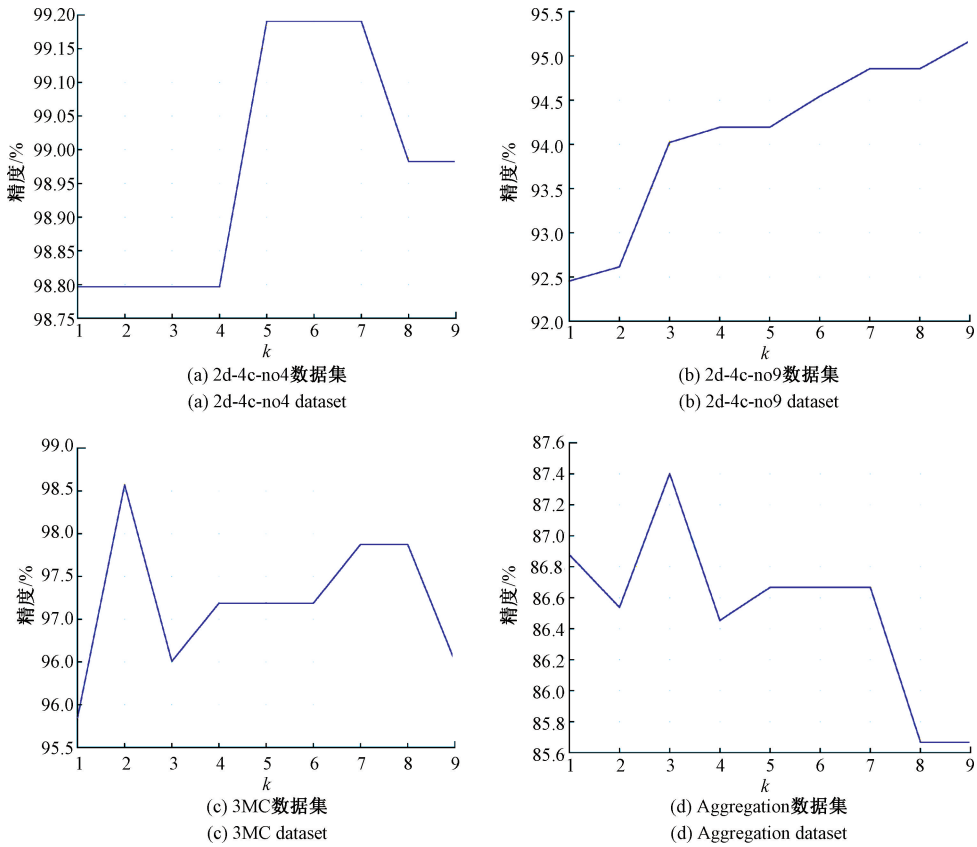


图 5 参数 k 对线性 NIWLSPTSVC 算法聚类性能的影响

Fig. 5 Influence of parameter k on the clustering performance of linear NIWLSPTSVC algorithm

4 应 用

在本节中介绍了所提算法 NIWLSPTSVC 的一些实际应用,并与现有算法进行比较。在来自 UCI 存储库的生物医学数据集上进行实验,旨在探索所提算法对生物医学数据集的适应性、实用性以及准确性等问题。在本节应用中,高斯核被用于所提算法和现有算法中。实验结果如表 6 所示,加粗字体为对应算法精度的最优值。

表 6 非线性算法在应用数据集上的结果

Table 6 Results of nonlinear algorithms on applied datasets

数据集	TWSVC	TBSVC	LSTBSVC	LSPTSVC	所提算法
HCV	80.947 7	81.093 3	81.328 5	**81.493 3**	**81.493 3**
CCBR	58.730 2	62.735 7	**64.444 4**	61.428 6	**64.444 4**

4.1 丙型肝炎病毒数据聚类

在丙型肝炎病毒(HCV)聚类的应用中,该数据集包含献血者和丙型肝炎患者的实验室值以及年龄等人口统计值,共 615 个实例,12 个属性和 5 个类别。本文对该数据集执行了一些基本的预处理,如标签编码,为所提算法做准备。结果如表 6 所示,从表中可以看到,所提算法的聚类精度表现是最好的。使用该算法系统对丙型肝炎病毒数据集进行聚类分析有助于深入理解患者之间的多样性和复杂性,为个性化医疗和治疗策略的制定提供一定的支持和帮助。

4.2 宫颈癌行为风险数据聚类

在宫劲癌行为风险(CCBR)聚类的应用中,该数据集共 72 个实例,19 个关于宫颈癌行为风险的属性和 2 个类别,其类别标签分别表示有宫颈和无宫颈癌的受访者。结果如表 6 所示,在该数据集中,所提算法在所有算法中的聚类精度是最好的。使用该算法系统对宫颈癌行为风险数据集进行聚类分析,可以深入了解不同个体之间的风险因素之间的关系,从而帮助医疗研究人员和决策者更好地制定预防和干预策略,在一定程度上降低宫颈癌的发生率和相关健康风险。

5　结　　论

本文提出了一种具有邻域信息加权的最小二乘投影双支持向量聚类算法，该算法构造了训练集的类内权重矩阵来提取同类数据点的相对密度，根据相对密度计算该点的相对权重，使用加权均值而不是标准均值作为类均值，考虑了每个样本的重要性，充分挖掘了数据集中具有相同标签数据点对之间的相关性信息，同时更好地反映同类样本的几何结构。通过求解一组线性方程，找到每个类别的聚类投影轴，而不是传统的超平面，继承了 LSPTSVC 算法优点。实验结果表明，所提算法在与其他算法保持相似的时间复杂度的情况下不仅提高了聚类质量，而且在不平衡数据集中也表现出了良好的泛化性能，此外，在生物医学数据集的应用中也表现不错，为医学数据集的实际应用提供了良好的理论基础。最后，本文所提算法需要耗费一定的时间找到所有点的 k 最近邻。因此，未来的工作需要研究一些新的技术来度量数据点的重要性。

参考文献

[1] 杜淑颖，丁世飞，邵长龙. 基于簇间连接的元聚类集成算法[J]. 南京大学学报(自然科学)，2023，59(6)：961-969.
DU SH Y，DING SH F，SHAO CH L. Meta-clustering integration algorithm based on intercluster connection[J]. Journal of Nanjing University (Natural Science)，2023，59(6)：961-969.

[2] 郑玉兰，徐伟. 基于时差和聚类算法的闪电定位方法[J]. 电子测量与仪器学报，2022，36(12)：176-184.
ZHENG Y L，XU W. Lightning location method based on time difference and clustering algorithm[J]. Journal of Electronic Measurement and Instrumentation，2022，36(12)：176-184.

[3] 杨滔，孙博，杨晓君. 基于超像素锚层收敛选点的高光谱图像聚类算法[J]. 电子测量技术，2023，46(6)：77-83.
YANG T，SUN B，YANG X J. Hyperspectral image clustering algorithm based on convergence point selection of super-pixel anchor layer[J]. Electronic Measurement Technology，2023，46(6)：77-83.

[4] REZVANI S，WANG X ZH，POURPANAH F. Intuitionistic fuzzy twin support vector machines[J]. IEEE Transactions on Fuzzy Systems，2019，27(11)：2140-2151.

[5] TANVEER M，GANAIE M A，BHATTACHARJEE A，et al. Intuitionistic fuzzy weighted least squares twin SVMs[J]. IEEE Transactions on Cybernetics，2022，53(7)：4400-4409.

[6] HUA X P，DING SH F. Weighted least squares projection twin support vector machines with local information [J]. Neurocomputing，2015，160：228-237.

[7] 常兴亚，武云鹤，陈东岳，等. 基于多任务学习的视频异常检测方法[J]. 仪器仪表学报，2023，44(8)：21-29.
CHANG X Y，WU Y H，CHEN D Y，et al. Video anomaly detection method based on multi-task learning[J]. Chinese Journal of Scientific Instrument，2023，44(8)：21-29.

[8] WANG Z，SHAO Y H，BAI L，et al. Twin support vector machine for clustering[J]. IEEE Transactions on Neural Networks and Learning Systems，2015，26(10)：2583-2588.

[9] BAI L，SHAO Y H，WANG Z，et al. Clustering by twin support vector machine and least square twin support vector classifier with uniform output coding [J]. Knowledge-Based Systems，2019，163：227-240.

[10] RICHHARIYA B，TANVEER M. Least squares projection twin support vector clustering(LSPTSVC)[J]. Information Sciences，2020，533：1-23.

[11] TANVEER M，GUPTA T，SHAH M，et al. Pinball loss twin support vector clustering [J]. ACM Transactions on Multimedia Computing，Communications，and Applications(TOMM)，2021，17(2s)：1-23.

[12] RICHHARIYAI B，TANVEER M. An efficient angle-based universum least squares twin support vector machine for classification [J]. ACM Transactions on Internet Technology(TOIT)，2021，21(3)：1-24.

[13] PHALKE S，VAIDYA Y，METKAR S. Big-O time complexity analysis of algorithm [C]. Pune：2022 International Conference on Signal and Information Processing(IConSIP)，IEEE，2022：1-5.

[14] UDDIN S，HAQUE I，LU H，et al. Comparative performance analysis of K-nearest neighbour(KNN) algorithm and its different variants for disease prediction[J]. Scientific Reports，2022，12(1)：6256.

[15] 陈素根，刘玉菲. 改进的 Ramp 孪生支持向量机聚类[J]. 计算机科学与探索，2023，17(11)：2767-2776.
CHEN S G，LIU Y F. Improved Ramp twin support vector machine clustering [J]. Exploration of Computer Science and Technology，2023，17(11)：2767-2776.

作者简介

王顺霞，硕士研究生，主要研究方向为机器学习与模式识别。
E-mail：2689826749@qq.com

黄成泉(通信作者)，博士，教授，主要研究方向为机器学习、模式识别与图像处理。
E-mail：hcq863@163.com

罗森艳，硕士研究生，主要研究方向为机器学习与模式识别。
E-mail：1563770769@qq.com

杨贵燕，硕士研究生，主要研究方向为机器学习与模式识别。
E-mail：2393350042@qq.com

蔡江海，硕士研究生，主要研究方向为模式识别与图像处理。
E-mail：870152989@qq.com