文章编号: 1001-0920 (2016) 03-0551-04 **DOI:** 10.13195/j.kzyjc.2014.1692

基于快速聚类分析的支持向量数据描述算法

程昊翔, 王 坚

(同济大学 a. 电子与信息工程学院，b. CIMS 研究中心，上海 201804)

摘 要: 针对数据中存在的噪声对数据描述建模的影响, 提出一种基于快速聚类分析的支持向量数据描述算法. 该算法通过快速聚类分析算法对所要建模的数据进行预处理, 通过预处理快速剔除数据中存在的影响建模的噪声; 然后再将基于 kNN 算法计算获得的权重值加权在每一个数据上, 进行支持向量数据描述算法的建模. 在标准数据集上的实验分析表明, 所提出的支持向量数据描述算法较传统的支持向量数据描述算法和密度驱动支持向量数据描述算法在准确度上具有较明显的提升.

关键词: 聚类分析；支持向量数据描述；加权

中图分类号: TP183 **文献标志码:** A

Support vector data description based on fast clustering analysis

CHENG Hao-xiang, *WANG Jian*

(a. College of Electronics and Information Engineering，b. CIMS Research Center，Tongji University，Shanghai 201804，China．Correspondent：CHENG Hao-xiang，E-mail：373668304@qq.com)

Abstract: To minimize the negative effects of the outliers in the training data set during the data description modeling, a support vector data description algorithm(SVDD) based on fast clustering analysis is proposed. The proposed approach consists of two stages of strategy. Firstly, a fast clustering analysis algorithm is applied to preprocessing the training data set. The outliers in the training dataset that affect the model are removed. Then, a weighted support vector data description based on kNN is used in the second stage. Experimental results on benchmark datasets show that the performance of the present approach is superior to conventional SVDD and density-induced SVDD in accuracy.

Keywords: clustering analysis；support vector data description；weighted

0 引 言

数据描述的主要目的是能够紧凑地描述一个数据集, 使目标数据在描述边界内并排除数据集内存在的噪声. 数据描述被广泛应用在噪声检测等领域, 也常被称为一类分类算法. Tex 等[1]于 2004 年提出的支持向量数据描述算法 (SVDD), 原理源自于支持向量机[2-3], 将数据集从输入空间通过核函数映射至高维空间, 在高维特征空间里寻找到的最小体积的球面边界将绝大部分的数据包含在边界内, 并用一小部分的支持向量来描述边界. 支持向量数据描述算法已经从不同方面进行了发展, 如支持向量聚类算法[4]、基于 SVDD 的 K-means 算法[5]等.

尽管支持向量数据描述算法已经获得较大的成功, 但仍存在支持向量数据描述算法产生的支持向量不能完整地描述数据的特性, 以及不能反映出数据集的分布. 对于克服支持向量数据描述算法存在的缺点, 人们已经展开很多的研究工作. Lee 等[6]提出了密度驱动的支持向量数据描述算法 (D-SVDD), Liu 等[7]提出了快速支持向量机数据描述算法等, 在一定程度上提高了支持向量数据描述算法对于噪声处理的性能. 但是, 上述算法都没有将数据进行预处理来过滤数据集中存在的噪声, 在一定程度上影响了数据描述模型的准确度.

本文提出一种基于快速聚类分析的支持向量数据描述算法 (FC-SVDD), 通过快速聚类分析算法将数据集中的噪声进行过滤, 然后应用 SVDD 进行数据描述建模. 快速聚类算法[8]是由 Alex 等提出的一种简洁快速的聚类分析算法, 在不同类簇上识别取得了很好的效果, 并且其参数非常容易确定. 通过该算法能够准确快速地过滤出奇异值, 然后产生新的训练数据集,

收稿日期: 2014-11-06；**修回日期:** 2015-02-01.

基金项目: 国家自然科学基金面上项目(71273188)；国家自然科学基金重大项目(91024031).

作者简介: 程昊翔(1986−), 男, 博士, 从事系统工程、机器学习的研究；王坚(1961−), 男, 教授, 博士生导师, 从事系统工程、智能优化算法等研究.

新产生的数据集能较准确地反映出数据集的内在分布. 第2阶段, 采用基于密度加权的支持向量数据描述算法进行建模. 通过对8个UCI数据集的实验分析表明, 本文所提出的算法较传统SVDD算法以及D-SVDD算法在一定程度上提高了对于数据描述的准确度.

1 支持向量数据描述算法

支持向量数据描述算法(SVDD)通过训练数据来学习得到超球形的数据描述空间. SVDD可通过核函数将输入空间映射到高维空间来学习得到灵活并且准确的数据描述模型, 得到的超球形数据描述边界通过一小部分的支持向量进行表示.

假设训练数据集为

$$T=\{x_i\in R^d|i=1,2,\cdots,N\},\tag{1}$$

N为训练数据的总数, d为数据的特征维度, $\Phi(x_i)$表示训练数据从输入空间非线性转换到高维高斯核空间. SVDD的目标是在高维特征空间寻找到以R为半径、以a为中心的最小超球面来描述目标数据集. 为了得到上述的数据集描述模型, 得到了如下的优化问题:

$$\min_{R,a,\xi} R^2+C\sum_{i=1}^{N}\xi_i;\tag{2}$$

$$\text{s.t.}\|x_i-a\|^2\leqslant R^2+\xi_i,\ \xi_i\geqslant 0,\ i=1,2,\cdots,N.\tag{3}$$

其中: C为超球体体积与描述错误误差之间的权衡值, ξ_i为松弛变量, 可以使部分噪声数据在超球面界外.

为解决式(2)和(3)的问题, 引入拉格朗日乘子, 得到

$$L(R,a,\xi,\alpha,\gamma)=$$
$$R^2+C\sum_{i=1}^{N}\xi_i-\sum_{i=1}^{N}\gamma_i\xi_i+\sum_{i=1}^{N}\alpha_i(\|x_i-a\|^2-R^2-\xi_i).\tag{4}$$

其中: $\alpha_i\geqslant 0,\gamma_i\geqslant 0$为拉格朗日因子. 由Karush-Kuhn-Tucker(KKT)条件, 可以得到

$$\begin{cases}\dfrac{\partial L}{\partial R}=0\rightarrow\sum\limits_{i=1}^{N}\alpha_i=1,\\ \dfrac{\partial L}{\partial a}=0\rightarrow a=\sum\limits_{i=1}^{N}\alpha_i x_i,\\ \dfrac{\partial L}{\partial \xi_i}=0\rightarrow C-\alpha_i-\gamma_i=0.\end{cases}\tag{5}$$

由式(5)可以得到, 超球面的中心由训练数据集中所有的数据点线性组合而成. 将式(5)中的3个式子代入式(4), 得到式(2)和(3)的对偶表示, 并且引入核函数K, 有

$$\max_{\alpha}\sum_{i=1}^{N}\alpha_iK(x_i,x_i)-\sum_{i,j=1}^{N}\alpha_i\alpha_jK(x_i,x_j);\tag{6}$$

$$\text{s.t. }0\leqslant\alpha_i\leqslant C,\ \sum_{i=1}^{N}\alpha_i=1,\ i=1,2,\cdots,N.\tag{7}$$

本文中, 将核函数$K(x_i,x_j)$用高斯核进行代替, 通过对上面对偶优化问题的求解, 从结果中得到大部分的α_i的值都是0. 可以将求解不为0的α_i分为两类:

1) $0<\alpha_i<C$为支持向量, 这些数据点分布在超球面的边界上;

2) $\alpha_i=C$为界外支持向量, 这部分数据点分布在超球面的外部.

尽管支持向量数据描述算法通过引入核函数能够灵活地对训练数据集进行紧密地描述, 但是支持向量数据描述仅仅基于支持向量而没有考虑训练数据集的密度分布以及噪声的影响, SVDD所获得的数据描述模型可能会错过最优描述模型. 近几年, 针对这些问题有不少的研究, 也提出了新的改进算法, 如基于密度加权的SVDD算法等都是要解决支持向量数据描述算法的鲁棒问题, 但是都没有考虑到在数据加权前对训练数据集进行预处理, 清除数据集中噪声的干扰. Alex等[8]提出了一种快速聚类分析算法, 可以同时过滤多类问题数据集中的噪声.

2 基于快速聚类分析的支持向量数据描述算法

本文提出了一种新的基于快速聚类分析的SVDD算法, 算法主要由2个阶段组成. 第1阶段主要是对训练数据集进行快速聚类分析, 这里采用了Alex等[8]提出的快速聚类分析算法, 快速准确地过滤训练数据集中存在的噪声数据; 第2阶段采用基于密度估计的支持向量数据描述算法对数据集进行建模.

首先, 对数据集T进行快速聚类分析. 快速聚类分析算法是一种简洁快速的聚类分析算法, 其可以识别各种形状的聚类. 通过分析, 可以得到算法的复杂度为$O(N^2)$. 下面介绍该算法如何完成第1阶段的噪声数据过滤. 算法中定义了两个参数: ρ_i为每个数据点的局部密度, δ_i为数据点到高局部密度的距离. ρ_i的计算方法如下:

$$\rho_i=\sum_j\chi(d_{ij}-d_c).\tag{8}$$

其中

$$\chi(x)=\begin{cases}1,\ x<0;\\ 0,\ \text{otherwise};\end{cases}\tag{9}$$

d_c为一个截断距离(超参数), ρ_i相当于到数据点i的距离小于d_c的点的个数. 算法通过证明后得到, ρ_i的

相对值在 d_c 值确定的情况下是比较鲁棒的, 一般做法是选择 d_c 使得平均每个数据点的附近点数为数据集总数的 1%∼2%.

本文采用 k 近邻法来获取 d_c 的值, 具体方法如下: N 表示数据集总数, $k = N \times 1\%$ 取整数来获取 k 值; $Q(x) = \{x_1^k, x_2^k, \cdots, x_N^k\}$ 中 $x_i^k\ (i = 1, 2, \cdots, N)$ 表示 x_i 的第 k 个近邻值, $Q(x)$ 表示数据集 $\{x_i | i = 1, 2, \cdots, N\}$ 中的每个数据集的第 k 个近邻值的集合; d_c 的计算公式如下:

$$d_c = \frac{(\min(Q(x)) + \max(Q(x)))}{2};$$

δ_i 的计算公式如下:

$$\delta_i = \min_{j:\rho_j > \rho_i} (d_{ij}), \tag{10}$$

其中 δ_i 表示数据点 i 到高密度点的最小距离. 对于密度最大的点, 设置 $\delta_i = \max_j (d_{ij})$, 通过上述计算, 可以得到数据集中所有的数据点所对应的 ρ_i 和 δ_i. 从论文定义的规则中可以得到, 对于拥有较大的 ρ_i 值且很大的 δ_i 的数据点, 认为是训练集中的某个类簇的中心, 而对于拥有较小 ρ_i 值且较大 δ_i 的数据点, 认为是训练数据集中存在的噪声. 在这一阶段, 通过快速聚类分析算法能过滤训练数据集中的噪声, 这里将去噪后获得的数据总数为 M 的数据集 T_1 表示为

$$T_1 = \{x_m^D \in R^d | m = 1, 2, \cdots, M\}, \tag{11}$$

其中 x_m^D 表示去除噪声后的数据集 T_1 中的数据点.

第 2 阶段, 对于每个数据点 x_m^D 计算其对应的加权权重值 W_m, 本文采用经典的 kNN 算法[9]进行计算, 用 $d(x_m^D, x_m^{D(k)})$ 表示 x_m^D 与第 k 个最邻近数据点 $x_m^{D(k)}$ 之间的距离, W_m 定义如下:

$$V_m = d(x_m^D, x_m^{D(k)}), \tag{12}$$

$$W_m = 1 - \frac{V_m}{\max\limits_{h=1,2,\cdots,M} V_h}. \tag{13}$$

通过上述的 W_m 定义, 可以推得当数据点 x_m^D 与其第 k 个邻近点距离较小时, 数据点具有较大的 W_m, 反之, 则 W_m 较小. 将 W_m 范围控制在 [0,1]. 将式 (12) 代入 SVDD 算法优化方程, 得到

$$\min_{R,a,\xi} R^2 + C \sum_{m=1}^{M} W_m \xi_m; \tag{14}$$

$$\text{s.t. } \|x_m^D - a\|^2 \leqslant R^2 + \xi_m,\ \xi_m \geqslant 0,\ m = 1, 2, \cdots, M. \tag{15}$$

可以得到拉格朗日对偶问题

$$\max_{\alpha} \sum_{m=1}^{M} \alpha_m K(x_m^D, x_m^D) - \sum_{m,n=1}^{N} \alpha_m \alpha_n K(x_m^D, x_n^D);$$

$$\text{s.t. } 0 \leqslant \alpha_m \leqslant W_m C,\ \sum_{m=1}^{M} \alpha_i = 1,\ m = 1, 2, \cdots, M. \tag{16}$$

由对偶问题 (16) 可以计算得到 $a = \sum\limits_{m=1}^{M} \alpha_m x_m$. 对于一个测试点 x, 可以计算得到到超球面中心的距离

$$f^2 = \|x - a\|^2 = K(x, x) - 2 \sum_{m=1}^{M} K(x_m, x) + \sum_{m,n=1}^{M} \alpha_m \alpha_n K(x_m, x_n). \tag{17}$$

当 $R - f \leqslant 0$ 时, 测试点被认为是噪声, 反之被认为是可以接受的数据点.

3　实验分析

将本文所提出的基于快速聚类分析的支持向量数据描述算法 (FC-SVDD) 与 SVDD 算法、D-SVDD 算法在 8 个 UCI 数据集上进行测试比较. FC-SVDD 算法复杂度为 $O(M^3 + N^2)$, SVDD 算法和 D-SVDD 算法的复杂度为 $O(N^3)$, 其中 M, N 为训练数据集数据量.

鉴于实验所采用的数据集都是多类别的数据集, 当对某一类进行数据描述建模时, 其他类别则当作噪声数据进行使用. 实验在 UBUNTU 14.04 环境下进行, 并与 Scikit-learn 机器学习工具包[10]中的 SVDD 算法相结合, 对上述测试算法采用 python 实现快速聚类分析. 采用高斯核函数 $K(x_i, x_i) = \exp(-q\|x_i - x_j\|^2)$. 对于上述算法中的参数 C 和 q, 采用 5 折交叉验证方法在 $\{2^i | i = -8, -7, \cdots, 7, 8\}$ 中选取. 表 1 给出了实验中所使用的 8 个 UCI 数据集信息.

表 1　测试数据集

数据集名称	数据量	数据维度	数据类别
Isolet	900	617	3
Wisconsin	683	10	2
Wine	178	13	3
Pendigits	3 000	16	3
Iris	150	4	3
Acute Inflammations	120	6	2
heart	270	13	2
Balance Scale	625	4	3

对于算法的评价标准, 首先引入两个概念: 真正率 (TPR) 和真负率 (TNR)[11], 即

$$\text{TPR} = \frac{\text{TP}}{(\text{TP} + \text{FN})},\ \text{TNR} = \frac{\text{TN}}{(\text{TN} + \text{FP})}.$$

其中: TP 为预测为正的正样本数, FN 为预测为负的正样本数, TN 为预测为负的负样本数, FP 为预测为正的负样本数. 对于不同算法的评价标准是

$$\text{准确度} = \frac{(\text{TP} + \text{TN})}{(\text{TP} + \text{TN} + \text{FN} + \text{FP})}.$$

通过对上述 5 个 UCI 数据集进行试验, 得到如表 2 所示的 3 种算法的准确度计算结果.

表 2 算法准确度测试结果

数据集	类别	SVDD	D-SVDD	FC-SVDD
Isolet	class 1	0.860 3	0.891 0	**0.902 7**
	class 2	0.799 2	0.938 7	**0.952 0**
	class 3	0.801 5	0.864 2	**0.882 0**
Wisconsin	class 1	0.945 0	**0.986 6**	0.968 0
	class 2	0.923 5	0.945 0	**0.972 2**
Wine	class 1	0.855 3	0.886 2	**0.897 4**
	class 2	0.891 2	0.920 6	**0.943 3**
	class 3	0.874 0	0.902 2	**0.916 0**
Pendigits	class 1	0.761 1	0.941 3	**0.960 5**
	class 2	0.814 3	0.923 2	**0.931 2**
	class 3	0.876 6	0.979 2	**0.985 4**
Iris	class 1	0.986 0	**0.991 0**	0.978 2
	class 2	0.918 2	0.937 7	**0.962 1**
	class 3	0.939 0	0.952 8	**0.966 2**
Acute Inflammations	class 1	0.952 0	0.963 3	**0.966 5**
	class 2	0.932 2	0.940 6	**0.951 6**
heart	class 1	0.866 7	0.906 6	**0.915 0**
	class 2	0.896 0	0.913 4	**0.921 8**
Balance Scale	class 1	0.920 0	0.935 0	**0.938 8**
	class 2	0.903 5	**0.931 6**	0.922 0
	class 3	0.932 0	0.941 6	**0.952 0**

由表 2 可以得到, 在 8 个 UCI 数据集中, 基于快速聚类分析的支持向量描述算法 (FC-SVDD) 准确率明显高于 SVDD 和 D-SVDD. 只有 Wisconsin 数据集的类别 1、Iris 数据集的类别 1 和 Balance Scale 数据集的类别 2, FC-SVM 算法的准确率稍差于 D-SVDD 算法. 相对于传统的 SVDD, 本文所提出的算法准确率有明显提高. FC-SVDD 算法通过快速聚类分析算法将数据集中存在的噪声数据进行过滤, 形成新的数据量较小且更为真实的数据集. 该数据集能够更好地反映数据集内在的真实分布, 大大减小了噪声数据对于数据描述模型的影响. 通过在第 2 阶段对每个训练集的数据引入加权值, 使得不同的数据对于数据描述模型影响的大小变得更为准确. 实验结果也反映出, 本文所提出的 FC-SVDD 算法较传统 SVDD 算法和 D-SVDD 算法具有更好的数据描述准确度. 综上所述, 本文所提出的 FC-SVDD 算法对于分类数据集的数据描述具有更好的泛化性, 表明了该算法的有效性.

4 结 论

针对训练数据集中存在噪声数据对于支持向量数据描述算法鲁棒性的影响, 本文提出了基于快速聚类分析的支持向量数据描述算法. 该算法首先引入快速聚类算法对数据集中存在的噪声进行过滤; 然后通过 kNN 算法计算加权值对数据集中的每个数据进行加权; 最后训练并建立数据描述模型. 本文算法采用快速聚类算法能够同时过滤分类数据集中的所有类别中存在的噪声, 大大减少噪声对数据描述模型的影响, 并且采用 kNN 算法对数据进行了加权, 能够较为准确地反映数据集的内在分布. 实验分析表明, 本文所提出的 FC-SVDD 算法较传统 SVDD 算法与 D-SVDD 算法具有较高的准确率, 能较好地描述数据集.

参考文献(References)

[1] Tax D M J, Duin R P W. Support vector data description[J]. Machine Learning, 2004, 54(1): 45-66.

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

[3] Smola A J, Scholkopf B. A tutorial on support vector regression[J]. Statistics and Computing, 2004, 14(3): 199-222.

[4] Ben-Hur B, Horn D, Sieglmann H T. Support vector clustering[J]. J of Machine Learning Research, 2001, 2: 125-137.

[5] Camastra F, Verri A. A novel kernel method for clustering[J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 2005, 27(5): 801-805.

[6] KiYong Lee, Dae-Won Kim, Kwang H Lee, et al. Density-induced support vector data description[J]. IEEE Trans on Neural Networks, 2007, 18(1): 284-289.

[7] Yi-Hung Liu, Yan-Chen Liu, Yen-Jen Chen. Fast support vector data descriptions for novelty detection[J]. IEEE Trans on Neural Networks, 2010, 21(8): 1296-1313.

[8] Alex Rodriguez, Alessandro Laio. Clustering bu fast search and find of density peaks[J]. Science, 2014, 344(619): 1492-1496.

[9] Peter Harrington. Machine learning in action[M]. Shelter Island: Manning Publications, 2012: 18-36.

[10] Scikit-learn. Machine learning in python[J]. J of Machine Learning Research, 2011, 12: 2825-2830.

[11] David A Freedom. Statistical models: Theory and practice[M]. Cambridge: Cambridge University Press, 2005: 115-130.

(责任编辑：孙艺红)