

DOI: 10.7652/xjtuxb201310001

一种迭代改进的球向量机故障诊断算法

杨清宇^{1,2}, 孙凤伟²

(1. 西安交通大学机械制造系统工程国家重点实验室, 710049, 西安;

2. 西安交通大学电子与信息工程学院, 710049, 西安)

摘要: 针对往复式压缩机故障数据规模大、常规诊断算法训练时间长的问題,提出了一种迭代改进的球向量机算法(IIBVM)。该算法将多个传感器采集到的压缩机不同工况下的运行数据输入球向量机进行训练,在训练过程中,增加了缓存用量,并在采样中跳过点积大于当前最远点点积的缓存点;引入缓存校正措施,每隔数次迭代即用点积和球心模计算公式对缓存中的数据进行一次校正;将距球心小于某个安全距离的点标记为无效点,并在下次采样到无效点时直接跳过;若核向量个数连续几次迭代保持不变,则提前终止迭代。训练后将新的压缩机运行数据代入决策函数,实现压缩机的故障诊断。在 4 个 UCI 标准数据集和实测的压缩机气阀故障数据集上进行的对比实验结果表明, IIBVM 算法与球向量机算法相比,训练时间最多可降低 50%,支持向量个数最多可减少 18%。

关键词: 往复式压缩机;故障诊断;大规模故障数据;球向量机

中图分类号: TP18 **文献标志码:** A **文章编号:** 0253-987X(2013)10-0001-06

An Iteratively Improved Ball Support Vector Machine Algorithm for Fault Diagnosis

YANG Qingyu^{1,2}, SUN Fengwei²

(1. State Key Laboratory for Manufacturing Systems Engineering, Xi'an Jiaotong University, Xi'an 710049, China;

2. School of Electronics and Information Engineering, Xi'an Jiaotong University, Xi'an 710049, China)

Abstract: An iteratively improved ball support vector machine (IIBVM) algorithm is proposed to focus on the problem that conventional diagnosis algorithms take long training time in dealing with large scale fault data of reciprocating compressor. Operating data collected by several sensors in different working conditions are inputted into a ball support vector machine for training. More points are cached and the cache points with cached dot products larger than that of the current farthest point will be skipped in the next iteration during training procedure. Then the cache correction is performed using the dot product and the center norm formula every certain iterations. Moreover, a point will be marked as invalid and skipped in the next iteration if its distance from the center is less than a safe distance. The iteration is terminated if the number of core vectors remains the same for several iterations. The new operating data generated after training are substituted into a decision function to realize fault diagnosis. Experimental results on four UCI datasets and an actual compressor fault dataset show that the training time is reduced at most by 50% and the number of support vectors is reduced by up to 18%.

Keywords: reciprocating compressor; fault diagnosis; large scale fault data; ball vector machine

收稿日期: 2013-05-26。 作者简介: 杨清宇(1974—),男,副教授,博士生导师。 基金项目: 国家自然科学基金资助项目(61075001)。

网络出版时间: 2013-07-18

网络出版地址: <http://www.cnki.net/kcms/detail/61.1069.T.20130718.0917.001.html>

往复式压缩机是工业生产中的一类关键设备,一旦发生故障将会造成巨大的生命财产损失,对其进行准确实时的故障诊断具有重要意义。

支持向量机因其优越的性能被广泛应用于故障诊断领域中^[1-2]。然而,传统支持向量机由于需要求解二次规划问题,其时间复杂度为 $o(N^3)$,空间复杂度为 $o(N^2)$ ^[3]。国内外众多学者提出了多种支持向量机改进算法,其中包括利用分解思想减小二次规划问题规模的选块算法、固定子集分解算法^[4]、序列最小优化算法^[5-7]等;采用长方形核矩阵以减小求解难度的约减支持向量机算法^[8];对支持向量机标准形式进行变形从而避免求解二次规划问题的最小二乘支持向量机算法;利用聚类方法排除非支持向量的算法^[9-10]等。这些改进算法在一定程度上可以降低支持向量机的复杂度,然而当数据规模过大时,仍然存在训练时间长、内存消耗大、诊断正确率低的问题。

Tsang 等人将计算几何的方法引入支持向量机的求解过程中,利用最小闭包球与二范数支持向量机的等价性,首先提出了核向量机(Core Vector Machine, CVM)算法^[3],该算法将最优分类超平面的求解转换为最小闭包球的求解,但求解过程中仍需求解1个规模较小的二次规划问题,为此 Tsang 等人又提出了球向量机(Ball Vector Machine, BVM)算法^[11]。与 CVM 算法不同的是, BVM 算法使用了半径固定的闭包球,迭代过程中只需利用迭代公式更新球心位置即可,无需求解二次规划问题,从而减小了求解难度,在大规模数据处理方面具有一定的优势,但将其用于往复式压缩机故障诊断时,训练时间仍需进一步降低。

本文针对 BVM 算法在往复式压缩机故障诊断中训练时间长的问題,提出了一种迭代改进的球向量机(Iteration Improved Ball Vector Machine, IIBVM)算法,并在4个UCI数据集以及1个真实的往复式压缩机故障数据集上进行了仿真测试。实验结果表明,本文算法与 BVM 算法在测试正确率相当的情况下,可有效降低算法的训练时间。

1 BVM 算法回顾

1.1 算法原理及求解步骤

以二分类问题为例,给定训练集 $S = \{\mathbf{z}_1, \dots, \mathbf{z}_m\}$,其中 $\mathbf{z}_i = (\mathbf{x}_i, \mathbf{y}_i)$ 为训练样本, $i = 1, \dots, m$, $\mathbf{x}_i \in \mathbf{R}_d$ 为1个 d 维向量, $\mathbf{y}_i \in \{1, -1\}$ 为 $\mathbf{x}_i$ 对应的标签。BVM 算法首先将二范数支持向量机等价为最小闭

包球问题,然后再将最小闭包球问题转换为闭包球问题进行求解。通过问题的等价与转换,训练样本 $\mathbf{z}_i$ 可以映射为高维空间内的点 $\boldsymbol{\varphi}(\mathbf{z}_i)$ 。

设训练集 S 的闭包球为 B_S ,则对于特征空间内的所有点 $\boldsymbol{\varphi}(\mathbf{z}_i)$ ($i = 1, \dots, m$) 满足 $\|\boldsymbol{\varphi}(\mathbf{z}_i) - \mathbf{c}\|^2 \leq R^2$,其中 $\mathbf{c}$ 为 B_S 的球心, R 为 B_S 的半径。

BVM 算法的求解步骤如下^[11]。

(1) 设置 B_S 的求解近似度系数 ϵ 为 ϵ_{goal} ,将球心更新次数 t 初始化为1,并初始化 $\mathbf{c}_0 = \boldsymbol{\varphi}(\mathbf{z}_0)$, $\epsilon = \epsilon_0$ 。

(2) 如果 $\epsilon < \epsilon_{\text{goal}}$,则已达到目标,算法终止;否则,转到步骤3。

(3) 检查是否存在 $\boldsymbol{\varphi}(\mathbf{z}_i)$,使 $\|\mathbf{c}_t - \boldsymbol{\varphi}(\mathbf{z}_i)\| > (1 + \epsilon)R$,如果不存在,则令 $\epsilon = \epsilon/2$,并转到步骤2;否则,利用式(1)找出距离球心 $\mathbf{c}_t$ 最远的点 $\boldsymbol{\varphi}(\mathbf{z}_{\max})$ 。

(4) 利用式(4)更新球心位置,使闭包球刚好接触到点 $\boldsymbol{\varphi}(\mathbf{z}_{\max})$ 。

(5) t 增加1,转到步骤3。

引入核函数 $k(\mathbf{z}_i, \mathbf{z}_j) = \boldsymbol{\varphi}^T(\mathbf{z}_i)\boldsymbol{\varphi}(\mathbf{z}_j)$, $j = 1, \dots, m$,当 $i = j$ 时, $k(\mathbf{z}_i, \mathbf{z}_i)$ 为常数,则点 $\boldsymbol{\varphi}(\mathbf{z}_i)$ 和球心 $\mathbf{c}_t$ 间的距离计算公式如下^[11]

$$\|\mathbf{c}_t - \boldsymbol{\varphi}(\mathbf{z}_i)\|^2 = \|\mathbf{c}_t\|^2 + k(\mathbf{z}_i, \mathbf{z}_i) - 2p_{t,i} \quad (1)$$

式中: $\|\mathbf{c}_t\|^2$ 为第 t 次更新后的球心模; $p_{t,i}$ 为第 t 次更新后 $\boldsymbol{\varphi}(\mathbf{z}_i)$ 与 $\mathbf{c}_t$ 的点积。

BVM 算法中球心 $\mathbf{c} = \sum_{i=1}^m \alpha_i \boldsymbol{\varphi}(\mathbf{z}_i)$,其中 α_i 为拉格朗日乘子,则点积 $p_{t,i}$ 和球心模 $\|\mathbf{c}_t\|^2$ 的计算公式分别为

$$p_{t,i} = \mathbf{c}_t^T \boldsymbol{\varphi}(\mathbf{z}_i) = \sum_{j=1}^m \alpha_j k(\mathbf{z}_i, \mathbf{z}_j) \quad (2)$$

$$\|\mathbf{c}_t\|^2 = \sum_{i,j=1}^m \alpha_i \alpha_j \boldsymbol{\varphi}^T(\mathbf{z}_i) \boldsymbol{\varphi}(\mathbf{z}_j) = \sum_{i,j=1}^m \alpha_i \alpha_j k(\mathbf{z}_i, \mathbf{z}_j) \quad (3)$$

球心更新公式如下

$$\mathbf{c}_{t+1} = \boldsymbol{\varphi}(\mathbf{z}_{\max}) + \beta_t (\mathbf{c}_t - \boldsymbol{\varphi}(\mathbf{z}_{\max})) \quad (4)$$

式中: $\beta_t = R / \|\mathbf{c}_t - \boldsymbol{\varphi}(\mathbf{z}_{\max})\|$ 。

1.2 算法实现方式

假设 N_{core} 和 N_{all} 分别为核向量个数和样本总数,则利用式(1)寻找离球心最远的点时,需要进行 $(N_{\text{core}} + N_{\text{core}}^2)N_{\text{all}}$ 次核函数运算。当数据集规模过大时,训练时间是难以接受的。为此, Tsang 在计算 BVM 的 LibCVM 实现方法中采用了以下几种措施来解决这一问题^[12]。

1.2.1 概率采样 在 BVM 算法的步骤3中,需找

到近似闭包球外的最远点,若遍历整个数据集则耗时过大。为此,Tsang 使用了概率采样法寻找近似闭包球外最远点,步骤如下:①设定采样次数 N_{sample} ,初始化 $n=0$,并将当前球心与最远点间的距离 d_{max} 初始化为 0;②如果 $n > N_{\text{sample}}$,则迭代终止,否则随机选取 1 个点 $\varphi(\mathbf{z}_i)$,计算该采样点与当前球心 $\mathbf{c}_t$ 的距离 d_i 后转到步骤 3;③计算 $d_{\text{max}} = \max(d_i, d_{\text{max}})$, n 增加 1 后返回步骤 2。

1.2.2 缓存点积和球心的模 如果核集内的某个点 $\varphi(\mathbf{z}_i)$ 被 2 次采样到,则将 $\varphi(\mathbf{z}_i)$ 与 $\mathbf{c}_t$ 的点积 $p_{t,i}$ 存入缓存,并在球心位置改变后,使用迭代方式进行 $p_{t,i}$ 缓存值的更新。将式(4)代入式(2),可得点积更新公式

$$p_{t+1,i} = \beta_t p_{t,i} + (1 - \beta_t) k(\mathbf{z}_{\text{max}}, \mathbf{z}_i) \quad (5)$$

将 $\|\mathbf{c}_t\|^2$ 存入缓存,在球心位置改变后,也可使用迭代方式进行缓存值更新。将球心更新式(4)代入 $\|\mathbf{c}_t\|^2$ 中,可得球心模更新公式

$$\|\mathbf{c}_{t+1}\|^2 = \beta_t \|\mathbf{c}_t\|^2 + (1 - \beta_t) k(\mathbf{z}_i, \mathbf{z}_i) + (\beta_t^2 - \beta_t) d_{\text{max}} \quad (6)$$

式(5)和式(6)相比式(2)和式(3),核函数计算次数大大减少。当下次概率采样开始后,如果采样到已经缓存过的点,可以直接读取缓存中的 $p_{t,i}$ 和 $\|\mathbf{c}_t\|^2$,并代入式(1)计算出该点与新球心的距离,从而节省了距离计算的时间。

2 迭代改进的球向量机算法

本文在深入研究 BVM 算法的求解步骤和具体实现方法后,提出了一种改进的球向量机算法——IIBVM,从 4 个方面对 BVM 算法进行了改进:①改变了缓存使用方式;②加入了缓存校正措施;③标记了无效点;④修改了迭代终止条件。

2.1 改变缓存使用方式

本文通过概率采样首先找出位于闭包球外且距离球心最远的 N_{cache} 个点,然后将这 N_{cache} 个点及其点积 $p_{t,i}$ 存入缓存中,若采样得到的点数小于 N_{cache} ,则按实际点的个数进行缓存。由于更多点存入缓存,增加了采样命中率,因此在 LibCVM 实现方法的基础上进一步节省了运算时间。

BVM 算法在采样到序号为 i 的点之后,首先查看缓存中 $p_{t,i}$ 的值。如果 $p_{t,i} \neq 0$,则认为该点已存入缓存,然后利用式(1)计算出该点与球心间的距离,最后与临时最远距离比较,判断该点能否成为最远点。

本文算法在已缓存点的使用上采用了与 BVM

算法不同的方式,具体如下。

设 $\varphi(\mathbf{z}_i)$ 为本次随机采样到的点, $\varphi(\mathbf{z}_{\text{tmp}})$ 为当前距离球心 $\mathbf{c}_t$ 最远的点, $p_{t,i}$ 为 $\varphi(\mathbf{z}_i)$ 和 $\mathbf{c}_t$ 的点积, $p_{t,\text{tmp}}$ 为 $\varphi(\mathbf{z}_{\text{tmp}})$ 和 $\mathbf{c}_t$ 的点积。根据式(1)可得

$$\|\mathbf{c}_t - \varphi(\mathbf{z}_i)\|^2 - \|\mathbf{c}_t - \varphi(\mathbf{z}_{\text{tmp}})\|^2 = 2(p_{t,\text{tmp}} - p_{t,i}) \quad (7)$$

如果 $p_{t,\text{tmp}} > p_{t,i}$,则 $\|\mathbf{c}_t - \varphi(\mathbf{z}_i)\|^2 > \|\mathbf{c}_t - \varphi(\mathbf{z}_{\text{tmp}})\|^2$,即点 $\varphi(\mathbf{z}_i)$ 可以替代 $\varphi(\mathbf{z}_{\text{tmp}})$ 成为最远点,如果 $p_{t,\text{tmp}} < p_{t,i}$,则表明点 $\varphi(\mathbf{z}_i)$ 不可能成为最远点,因此无需再计算距离,直接跳过该点进行下一次采样即可,从而节省了计算时间。

2.2 加入缓存校正措施

缓存中的点积 $p_{t,i}$ 可由点积更新式(5)进行更新,但是由于数据存储精度的影响,式(5)求得的 $p_{t,i}$ 与点积计算式(2)直接计算得到的 $p_{t,i}$ 之间会出现一定的误差,从而使距离计算精度变差,最终对算法的精度造成影响。同理可知缓存中的球心模 $\|\mathbf{c}_t\|^2$ 也会因更新式(6)产生一定的误差。

本文在增加缓存使用量之后,在算法求解过程中加入了缓存校正措施,即闭包球球心每更新 N_{fix1} 次,则利用式(2)对缓存中的 $p_{t,i}$ 进行一次校正;闭包球球心每更新 N_{fix2} 次,则利用式(3)对缓存中的 $\|\mathbf{c}_t\|^2$ 进行一次校正。

由于点积和球心模的计算量较大(分别为 N_{core} 和 N_{core}^2 次核运算),因此缓存校正的频率不宜过高,即 N_{fix1} 和 N_{fix2} 通常应取较大的数值。IIBVM 算法中若将 N_{fix1} 和 N_{fix2} 设置为 -1,则表示不进行缓存校正。

2.3 标记无效点

BVM 算法采用了贪婪的球心迭代方式,随着 ϵ 迭代过程的进行,当前球心位置将逐渐接近闭包球的最优球心位置。因此,概率采样过程中得到的闭包球外的点将逐渐减少,而闭包球内的点将逐渐增多。由于闭包球内部的点无助于闭包球的球心位置更新,因此通过一定的策略减少闭包球内点的计算量将有助于算法速度的提升。

图 1 为闭包球球心位置更新过程示意图。由图 1 可知,若点 $\varphi(\mathbf{z})$ 与当前球心 $\mathbf{c}_0$ 的距离为 R_1 ,闭包球半径为 R ,则球心移动到 $\mathbf{c}_1$ 后, $\mathbf{c}_0$ 与 $\mathbf{c}_1$ 的距离 $l = R_1 - R$ 。与球心 $\mathbf{c}_0$ 距离小于 r 的点在球心更新之后仍然在新的闭包球内,其中

$$r = R - l = 2R - R_1 \quad (8)$$

基于上述原理,本文给出了标记无效点的改进策略。在概率采样过程中计算了 $\mathbf{c}_t$ 与 N_{sample} 个采样

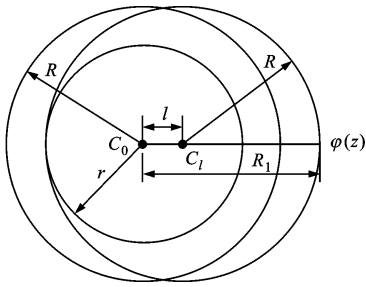


图 1 闭包球球心位置更新

点的距离,但这 N_{sample} 个距离在 BVM 算法中进行了一次距离比较后就被舍弃,已有的计算结果没有得到充分利用。在本文的改进算法 IIBVM 中,将这 N_{sample} 个距离先进行暂存,当闭包球外的最远点找到后,利用式(8)计算出 r ,并与暂存的 N_{sample} 个距离进行比较,若某点距离小于 r 则标记为无效点。在下次概率采样进行时,若采样到无效点,无需进行距离计算,可直接跳过。

2.4 修改迭代终止条件

在 LibCVM 实现方法中,多分类问题采用一对一方式进行,即任意两类样本之间训练 1 个二分类器,因此对于一个类数为 N_{class} 的多分类问题,需要训练 $N_{\text{class}}(N_{\text{class}} - 1)/2$ 个二分类器,且这些二分类器采用一个相同的 ϵ_{goal} 。这种做法默认所有的二分类器训练难度都相同,因此必须指定一个足够小的 ϵ_{goal} 才能达到训练精度要求。然而,在实际分类问题中,不同的二分类器的训练难度是不同的,有的二分类器的两类数据之间近乎线性可分,即使指定一个较大的 ϵ_{goal} 也能达到精度要求,而有的二分类器的两类数据之间重叠程度较高,算法必须精确调整球心位置才能得到满足精度需求的分类超平面。

设置较小的 ϵ_{goal} 固然会提高算法精度,但算法将在易分的二分类器的训练上消耗不必要的时间,从而使整体训练时间增加;设置较大的 ϵ_{goal} ,虽然会加快训练速度,但又无法满足难分的二分类器的精度需求,从而导致算法精度下降。

为此,本文将 LibCVM 实现方法中的迭代终止条件修改为:在 ϵ 的迭代中,若连续 N_{stop} 次迭代核向量数目保持不变,则认为当前的二分类器已经达到了精度要求,提前终止迭代(此时 $\epsilon \geq \epsilon_{\text{goal}}$)。

3 实验验证

在 CPU 为 E6500(双核 2.93 GHz)、内存为 2 GB 的计算机上,对本文提出的 IIBVM 算法进行了实验验证。首先在 4 个 UCI 标准数据集上对

IIBVM 算法进行了验证和对比分析,然后利用真实的往复式压缩机运行数据验证了 IIBVM 算法在故障诊断中的实际应用效果。

3.1 标准数据集上的实验验证

实验中使用了如表 1 所示的 4 个 UCI 标准数据集。

表 1 实验数据集

数据集	分类数	维数	训练样本数	测试样本数
optdigits	10	64	3 823	1 797
pendigits	10	16	7 494	3 498
reuters	2	8 135	7 770	3 299
letter	26	16	15 000	5 000

注:分类数指数据集中的样本种数;维数指对每个样本进行表征的数据的维数。

实验中选取 $10^{-7} \leq \epsilon_{\text{goal}} \leq 10^{-3}$,核函数选为高斯核函数 $\exp(-\|x_i - x_j\|^2/\sigma)$,其中 σ 为样本点之间平均距离的平方。

实验过程中记录了 BVM 算法和 IIBVM 算法在不同数据集上的训练时间 T_{train} 、测试正确率 R_{test} 以及支持向量个数 N_{sv} 这 3 个性能指标。每个数据集上的实验各进行多次,取其平均值作为最终的性能指标。实验结果如图 2~图 4 所示。

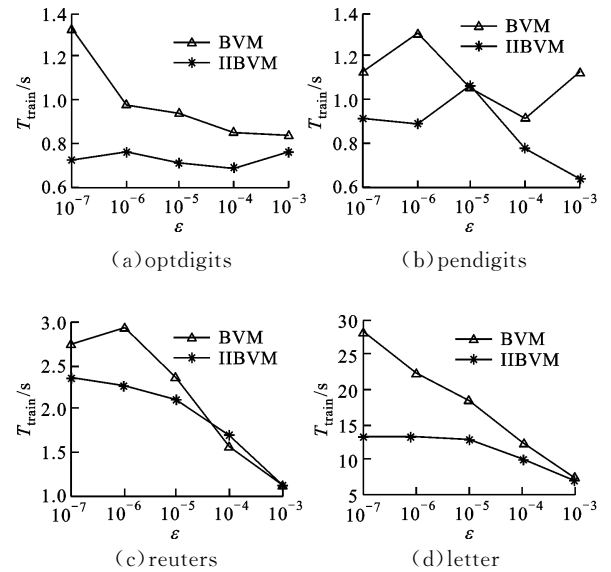


图 2 4 个数据集的训练时间对比

由图 2 的训练时间对比可以看出,IIBVM 算法在 4 个数据集的测试中都快于 BVM 算法,letter 数据集在 $\epsilon_{\text{goal}} = 10^{-7}$ 时甚至可加速 50% 以上。由于 IIBVM 算法将更多点进行缓存,增加了概率采样命中率,减少了距离计算中核运算的次数,因此对于维数高的数据,每采样到 1 个缓存点,可减少 $(N_{\text{core}} +$

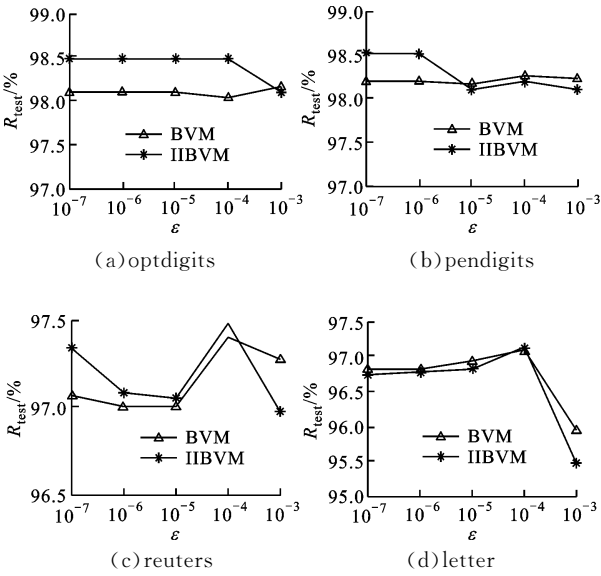


图 3 4 个数据集的测试正确率对比

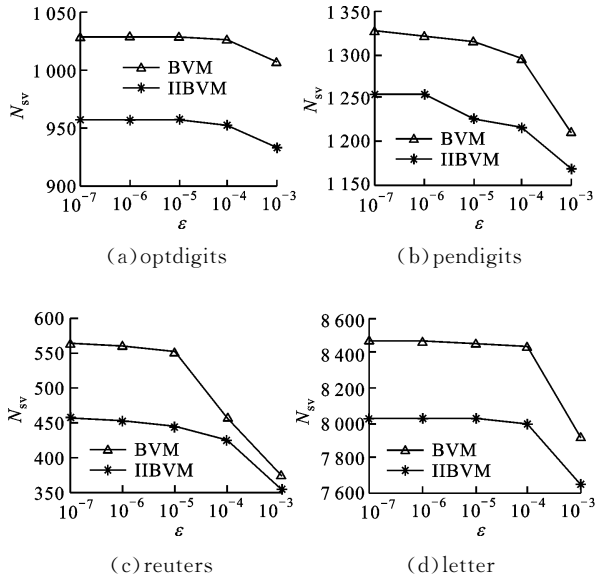


图 4 4 个数据集的支持向量数对比

$N_{\text{core}}^2)N_{\text{dim}}$ 次运算,采样到的缓存点越多,则算法加速越明显。另外对于多类数据集,无论 BVM 算法或 IIBVM 算法都需要训练多个二分类器,但不同的二分类器训练难度不同,若全局定义同一个 ϵ_{goal} ,将使算法在易分数数据集训练上消耗不必要的时间。由于 IIBVM 算法使用了不同的迭代终止条件,在训练时可根据核向量数目的变化决定是否提前终止迭代,因此 IIBVM 算法在类数多的数据集上展现出一定的速度优势。

在测试正确率上,IIBVM 算法也取得了较好的效果。如图 3 所示,在 optdigits、pendigits 和 reuters 数据集上 IIBVM 算法的最高正确率以及多个 ϵ

下的正确率均高于 BVM 算法。虽然在 letter 数据集上 IIBVM 算法在多个 ϵ 下的正确率略低于 BVM 算法,但差距均在 0.5% 以内,且 IIBVM 算法的最高正确率高于 BVM 算法,可以认为 2 种算法的测试精度相当。从实验结果可以看出,IIBVM 算法的测试精度虽然没有全面超越 BVM 算法,但可以认为 IIBVM 算法的训练速度提升并没有以牺牲正确率为代价。

另外,由图 4 可知,IIBVM 算法在 4 个数据集上得到的支持向量数均少于 BVM 算法,支持向量数的减少可带来两方面的优势:一方面,既可以在训练完成时减少支持向量的存储空间消耗,又可以在测试时减少内存中支持向量占用的空间;另一方面,支持向量数的减少可使决策函数的计算复杂度降低,从而加快算法对新样本的分类速度。

实验中选用的 4 个 UCI 标准数据集具有类数多、维数高、数据规模大的特点,由图 2~图 4 的实验结果可知,IIBVM 算法达到了预期设计要求,但在往复式压缩机故障诊断中的应用效果仍需进一步验证。

3.2 往复式压缩机故障诊断实例

在一个两级三缸的往复式压缩机进、排气口附近安装了 4 个温度传感器以及 4 个压力传感器,并在 5 种人工设定的故障工况下采集运行数据。采样频率为 200 Hz,采样时间为 5 s,总计获得 5 000 组温度和压力数据。该数据集中不同故障工况下的数据分布在非规则的复杂空间内并呈现一定的流形分布,同时由于测量噪声的影响和数据集本身的分布特点,使不同工况下的数据之间有一定程度的重叠与交叉。

在故障诊断实验中,使用 2 000 组数据作为训练集,训练完成之后利用剩余的 3 000 组数据进行算法性能测试,实验结果如图 5 所示。

由图 5 可以看出,IIBVM 算法相比 BVM 算法训练时间减少了 38.5%,诊断正确率提高了 0.6%,正确率达到了 98% 以上。在支持向量数方面,IIBVM 算法则略高于 BVM 算法,这是由压缩机故障数据集重叠程度大的特点决定的,较多的重叠点分布于闭包球球面附近。为了提高训练精度,算法在精确调整球心位置的过程中,更多的点成为支持向量,从而使支持向量数稍有增加。在实际应用中可以在正确率提升和支持向量数增加两者之间进行折中。

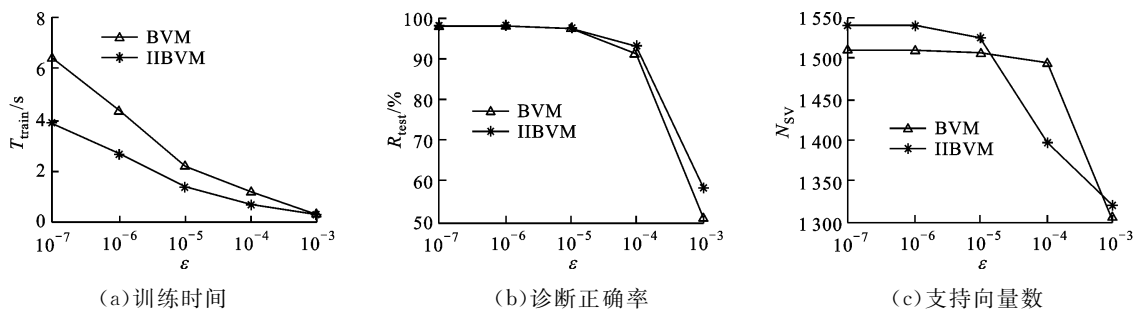


图 5 压缩机故障诊断实验结果

4 结 论

本文在深入研究 BVM 算法的基本原理、求解步骤和具体实现方法之后,通过改进缓存使用方式、加入缓存校正措施、标记无效点和修改迭代终止条件等措施,提出了一种迭代改进的球向量机算法。在 4 个 UCI 标准数据集以及 1 个真实的往复压缩机气阀故障数据集上进行了仿真实验。实验结果验证了 IIBVM 算法在往复压缩机故障诊断中应用的有效性,同时该算法也可以应用于其他具有大规模、多类别、多维度特点的数据处理问题中。

参考文献:

- [1] 易辉, 宋晓峰, 姜斌, 等. 基于结点优化的决策导向无环图支持向量机及其在故障诊断中的应用 [J]. 自动化学报, 2010, 36(3): 427-432.
YI Hui, SONG Xiaofeng, JIANG Bin, et al. Support vector machine based on nodes refined decision directed acyclic graph and its application to fault diagnosis [J]. Acta Automatic Sinica, 2010, 36(3): 427-432.
- [2] 王德成, 林辉. 一种 SVM 不平衡分类方法及其在故障诊断中的应用 [J]. 电机与控制学报, 2012, 16(9): 48-52.
WANG Decheng, LIN Hui. Imbalanced pattern classification method based on support vector machine and its application on fault diagnosis [J]. Electric Machines and Control, 2012, 16(9): 48-52.
- [3] TSANG I W, KWOK J T, CHEUNG P. Core vector machines: fast SVM training on very large data sets [J]. Journal of Machine Learning Research, 2006, 6 (1): 363-392.
- [4] OSUNA E, FREUND R, GIROSIT F. Training support vector machines: an application to face detection [C] // Proceedings of Computer Vision and Pattern Recognition. Piscataway, NJ, USA: IEEE, 1997:

130-136.

- [5] PIATT J C. Fast training of support vector machines using sequential minimal optimization [M] // SCHÖLKOPF B, BURGESS C J C, SMOLA A J. Advances in Kernel Methods: Support Vector Learning. Cambridge, MA, USA: MIT Press, 1999: 185-208.
- [6] JOACHIMS T. Making large-scale SVM learning practical [M] // SCHÖLKOPF B, BURGESS C J C, SMOLA A J. Advances in Kernel Methods: Support Vector Learning. Cambridge, MA, USA: MIT Press, 1999: 169-184.
- [7] CHANG Chih Chung, LIN Chih Jen. LIBSVM: a library for support vector machines [J]. ACM Transactions on Intelligent Systems and Technology, 2011, 2 (3): 1-27.
- [8] LEE Y J, MANGASARIAN O L. RSVM: reduced support vector machines [C] // Proceedings of SIAM International Conference on Data Mining. Chicago, IL, USA: SIAM, 2001: 5-7.
- [9] BOLEY D, CAO Dongwei. Training support vector machine using adaptive clustering [C] // Proceedings of SIAM International Conference on Data Mining. Chicago, IL, USA: SIAM, 2004: 126-137.
- [10] 张学工. 关于统计学习理论与支持向量机 [J]. 自动化学报, 2000, 26(1): 32-42.
ZHANG Xuegong. Introduction to statistical learning theory and support vector machine [J]. Acta Automatic Sinica, 2000, 26(1): 32-42.
- [11] TSANG I W, KOCOR A, KWOK J T. Simpler core vector machines with enclosing balls [C] // Proceedings of the 24th International Conference on Machine Learning. New York, USA: ACM, 2007: 911-918.
- [12] TSANG I W, KOCOR A, KWOK J T. Libsvm toolkit [EB/OL]. (2011-08-29) [2013-04-12]. <http://c2inet.sce.ntu.edu.sg/ivor/cvm.html>.

(编辑 刘杨)