doi:10.11887/j.cn.202002023 http://journal.nudt.edu.cn

数据基故障诊断算法更新问题研究*

赵晨旭,涂 遗,邱 静,刘冠军
(国防科技大学 装备综合保障技术重点实验室,湖南 长沙 410073)

摘 要:机内测试被广泛应用于故障诊断、装备健康管理与预测等领域。针对机内测试设备在设计和升级时遇到的分类器更新、样本数量不平衡、硬件条件限制问题,提出初步解决方案。利用基于密度的聚类和人工免疫方法处理原始数据;提出基于代表样本点的混合学习方法;利用支持向量机和仿真案例验证所提方法。结果表明所提方法能够解决上述问题,有助于基于数据的机内测试设备设计与升级。

关键词:机内测试;数据基故障诊断;设计改进;状态基维修
中图分类号:TH17;TP30 **文献标志码**:A **文章编号**:1001-2486(2020)02-171-06

Study of data based fault diagnosis algorithm update problem

ZHAO Chenxu, TU Yi, QIU Jing, LIU Guanjun
(Science and Technology on Integrated Logistics Support Laboratory, National University of Defense Technology, Changsha 410073, China)

Abstract: BITE (built-in test equipment) is widely used in many fields such as fault diagnosis, equipment prognosis and health management. The problems encountered in the process of BITE design and update, including the classifiers update, samples imbalance and hardware limitation, were analyzed, and the initial solutions were proposed. The density-based cluster and artificial immune system were applied to process the raw data; the delegates-based hybrid learning methods were proposed. The evaluation of the solution was validated by the numerical and experiment examples with support vector machine. Results show that the proposed solution can solve the mentioned problems well and is helpful for data based fault diagnosis design and update in the process of BITE maturation.

Keywords: built-in test; data based fault diagnosis; design improvement; condition-based maintenance

1 问题分析

机内测试系统是传感器、数据处理器、诊断软件等部分结合的产物,作为装备的一部分,可以使使用者尽快开展故障诊断与预测。随着计算机技术的不断发展,基于机器学习的故障诊断,尤其是基于数据的分类在故障检测与隔离过程中受到了越来越多的重视和应用[1]。将传感器采集到的数据经过一定的数据处理后得到装备运行的实时特征,将其与装备研制和运行中累积的运行状态"知识"作对比,可快速确定装备的运行状态。对于基于机器学习的故障诊断算法,足够的训练样本是保证得到高精度诊断结论的必要条件,只有经过充分训练的算法才能达到故障诊断设计要求。由于装备实际工作剖面复杂多样,在装备测试性设计时,受限于设计时间和经费,上述前提一般是不成立的;尤其是设计之初,往往难以得到装备故障数据的全样本空间,对于缺少相似产品的新研装备而言,得到正常状态的全样本空间通常也是比较困难的。在这种情况下,为了保证诊断结论的准确性,故障诊断决策算法需要随着试验或者使用过程中数据的累积而不断迭代更新[2]。在这个过程中,机器学习算法通常面临着分类器更新训练问题[3-5]、训练样本不平衡问题[6-8]和硬件存储容量限制[9]等问题。随着机器学习热度的不断增加,上述问题受到了越来越多的关注。但是从目前的文献来看,成果多集中在某个单一问题的解决上,统筹考虑上述三个问题,提出一套系统解决方案的成果还未发现。针对该情况,本文试图提出一套简单实用的方法,解决在测试性设计改进过程中如何开展诊断算法更新工作的问题。

2 问题解决

2.1 基于密度的大样本数据压缩

数据压缩是一种用少量样本表征完整原始样本集大部分特征区域的数据缩减方法[10]。基于

* 收稿日期:2018-12-11
作者简介:赵晨旭(1987—),男,河南郑州人,工程师,博士,E-mail:zhao_chenxu@126.com

密度的聚类是常用的聚类方法之一，本文参考该方法开展大样本数据压缩，在缓解样本量不平衡的同时，解决算法更新时原始样本不断增多、存储耗费大的问题。

表征装备故障状态的特征向量通常是高维的，向量元素通常包含实数值和属性值两种类型。用 s 维特征向量 $\boldsymbol{x}=\{x_1^{\mathrm{r}},\cdots,x_l^{\mathrm{r}},x_{l+1}^{\mathrm{c}},\cdots,x_s^{\mathrm{c}}\}$ 表示样本点 $\boldsymbol{x}$，其中 $\{x_1^{\mathrm{r}},\cdots,x_l^{\mathrm{r}}\}$ 为实数特征，$\{x_{l+1}^{\mathrm{c}},\cdots,x_s^{\mathrm{c}}\}$ 为属性特征，则可以用式(1)定义样本点 $\boldsymbol{x}_i$ 与 $\boldsymbol{x}_j$ 之间的距离[11]。

$$d(\boldsymbol{x}_i,\boldsymbol{x}_j)=\left[\sum_{t=1}^{l}(x_{i,t}^{\mathrm{r}}-x_{j,t}^{\mathrm{r}})^2\right]^{0.5}+\lambda\sum_{t=l+1}^{s}\delta(x_{i,t}^{\mathrm{c}},x_{j,t}^{\mathrm{c}}) \tag{1}$$

式(1)右侧第一项为用欧氏距离表征的实数特征距离，第二项为用相异匹配测度表征的属性特征距离，λ 用以调整属性特征在距离测度中的权重，$\delta(\cdot)$ 为示性函数。

$$\delta(a,b)=\begin{cases}0, a=b\\ 1, a\neq b\end{cases} \tag{2}$$

根据定义，$d(\boldsymbol{x}_i,\boldsymbol{x}_j)$ 可以用来衡量两个样本之间的相似程度：$d(\boldsymbol{x}_i,\boldsymbol{x}_j)=0$ 时，两个样本最相似，甚至相同；距离越大，两个样本之间的相似性越差。

对于容量为 N 的样本集，在距离函数的基础上，可以用式(3)定义任一样本 $\boldsymbol{x}$ 附近的样本分布密度。

$$\rho(\boldsymbol{x})=\frac{1}{N}\sum_{i=1}^{N}d(\boldsymbol{x},\boldsymbol{x}_i) \tag{3}$$

$\rho(\boldsymbol{x})$ 越大说明样本集中与样本 $\boldsymbol{x}$ 相似的样本点越多；反之则说明样本集中与样本 $\boldsymbol{x}$ 相似的样本点较少。

定义以 $\boldsymbol{x}_i$ 为中心、以 ε 为半径的超球为 $\boldsymbol{x}_i$ 的 ε 邻域，令 $N_\varepsilon(\boldsymbol{x}_i,\boldsymbol{X})$ 表示样本集 $\boldsymbol{X}$ 与该邻域交集的样本点个数，则 $N_\varepsilon(\boldsymbol{x}_i,\boldsymbol{X})$ 越大，表示 $\rho(\boldsymbol{x}_i)$ 越大。对于阈值 q，若 $N_\varepsilon(\boldsymbol{x}_i,\boldsymbol{X})\geqslant q$，则可称 $\boldsymbol{x}_i$ 为样本集的核心对象。核心对象通常作为代表样本点被保留至代表样本集中。根据定义可知，ε 越大，核心对象需要代表的区域越大，核心对象数量越少；q 越大，核心对象的代表性越强，核心对象数量也越少。于是通过调整 $\{\varepsilon,q\}$ 取值可调整代表样本集的样本个数。为保证代表样本点的均匀分布，在实际应用中通常令 $q=1$，通过调整 ε 取值控制代表样本容量，样本数量随着 ε 的减小而增多。

令 $\boldsymbol{X}^{\mathrm{un}}$ 表示待选样本集，其中待选样本按 $\rho(\boldsymbol{x})$ 取值从大到小的顺序依次编号。令 $\boldsymbol{P}=\{\boldsymbol{p}_1,\cdots,\boldsymbol{p}_{N_{\mathrm{D}}}\}$ 表示数据压缩后得到的代表样本集，其中 N_{D} 为要求的代表样本集容量，于是可以给出如算法 1 所示的代表样本集生成过程。该代表样本集即可作为数据压缩后的训练样本。该方法能够保证生成的训练样本既涵盖待选样本集的核心对象，又包含奇异特征点，从而张满原始样本集分布空间。

算法 1　获取代表样本集

Alg. 1　Get the sample set

初始化：$\boldsymbol{P}=\boldsymbol{x}=\boldsymbol{X}(1)$，$\boldsymbol{X}^{\mathrm{un}}=\boldsymbol{X}-\{\boldsymbol{x}\}$
1. while　$\boldsymbol{X}^{un}\neq\varnothing$　do
2.　　for $i=1:numel(\boldsymbol{X}^{\mathrm{un}})$
3.　　if $N_\varepsilon(\boldsymbol{x},\boldsymbol{P})<q$
4.　　$\boldsymbol{P}=\boldsymbol{P}+\{\boldsymbol{x}\}$
5.　　end if
6.　　$\boldsymbol{X}^{\mathrm{un}}=\boldsymbol{X}^{\mathrm{un}}-\{\boldsymbol{x}\}$
7.　end for
8. end while

根据基于密度的聚类方法中次级聚类中心的概念，以每个代表样本点为中心，以距离测度为标准，则可将原始样本集分为 N_{D} 类。$\boldsymbol{x}_i$ 属于第 k 类，当且仅当 $\boldsymbol{x}_i$ 满足 $k=\arg\min_{j,\boldsymbol{p}_j\in\boldsymbol{P}}\{\|\boldsymbol{x}_i-\boldsymbol{p}_j\|\}$，则属于第 k 类的原始样本点总数可表示为 $N(\boldsymbol{p}_k,\boldsymbol{P})$。

2.2　基于人工免疫的小样本数据扩充

基于启发式算法的数据扩充是常见的伪数据生成方法之一。特征联合分布密度函数是启发式样本扩充的基础。随着特征维数的增高，联合分布函数也会逐渐复杂，并且当数据量较小时通常难以得到准确的分布函数参数估计。受人工免疫系统[12]启发，将原始样本 $\boldsymbol{x}$ 看作抗原，与免疫系统生成的抗体组成新样本集，则可以在扩充样本容量的同时，丰富数据分布的多样性，并且该方法不需要给出特征联合分布函数，具体流程如下。

Step1：计算原始样本集 $\boldsymbol{X}$ 中每个样本 $\boldsymbol{x}$ 的分布密度系数 $\rho(\boldsymbol{x})$，同时随机生成样本集 $\boldsymbol{X}$ 对应的未成熟抗体种群 $\boldsymbol{A}$。

Step2：对原始样本集 $\boldsymbol{X}$ 中的每个样本 $\boldsymbol{x}$ 执行如下步骤：

1)计算未成熟抗体种群 $\boldsymbol{A}$ 中每个抗体 $\boldsymbol{a}_j$ 与原始样本 $\boldsymbol{x}$ 的亲和度 $Af_j=1/d(\boldsymbol{a}_j,\boldsymbol{x})$，其中 $d(\boldsymbol{a}_j,\boldsymbol{x})$ 如式(1)所示。

2)选出与 $\boldsymbol{x}$ 亲和度最高的 n 个抗体 $\{\boldsymbol{a}_1',\boldsymbol{a}_2',\cdots,\boldsymbol{a}_n',\}$，并对这 n 个抗体执行克隆操作，其中个体 $\boldsymbol{a}_k'(k=1,2,\cdots,n)$ 的克隆数量设定为 $n_k=$

$round(T\times\rho(\boldsymbol{x})\times Af_k)$,$T$ 表示最大允许克隆数。将克隆抗体进行合并,组成克隆抗体种群 $\boldsymbol{C}_i^*=\{\boldsymbol{a}'_{11},\boldsymbol{a}'_{12},\cdots,\boldsymbol{a}'_{1n_1},\cdots,\boldsymbol{a}'_{n1},\boldsymbol{a}'_{n2},\cdots,\boldsymbol{a}'_{nn_n}\}$。

3)对每个抗体 $\boldsymbol{a}'_{kl}\in\boldsymbol{C}_i^*$ 执行变异操作 $\boldsymbol{a}''_{kl}=\boldsymbol{a}'_{kl}+\alpha\times(\boldsymbol{a}'_{kl}-\boldsymbol{x}_i)\times rand(0,1)$,得到成熟抗体种群 $\boldsymbol{C}_i=\{\boldsymbol{a}''_{11},\boldsymbol{a}''_{12},\cdots,\boldsymbol{a}''_{1n_1},\cdots,\boldsymbol{a}''_{n1},\boldsymbol{a}''_{n2},\cdots,\boldsymbol{a}''_{nn_n}\}$,将其与其他抗原的成熟抗体种群进行合并,得到记忆抗体库 $\overline{\boldsymbol{A}}=[\overline{\boldsymbol{A}},\boldsymbol{C}_i]$。

Step3:将记忆抗体库 $\overline{\boldsymbol{A}}$ 与原始样本集 $\boldsymbol{X}$ 合并,可得扩充后的样本集 $\boldsymbol{X}=[\boldsymbol{X},\overline{\boldsymbol{A}}]$。

由于属性特征的取值通常是有限并且少量的,如果对其进行变异操作,极可能产生大量实际上不存在的样本属性特征,从而引入不必要的人为分类误差。于是,本文对属性特征采取不变异仅繁殖的处理方法。

利用上述过程进行样本扩充,既能保持原样本的重要信息,又能得到多种近似样本。通过控制参数{n,T}取值可以调整扩充样本集的容量,调整参数 $\alpha\in[0,1]$ 取值能够控制新样本与原始样本的相似程度,α 取值越大,新样本与原始样本相似度越高。

2.3　基于代表样本点的混合学习

常见的分类器更新学习方法主要包括批量学习和增量学习两种。传统的批量学习虽然能够较好地处理样本容量限制与知识空间退化的矛盾,但算法更新需要利用所有历史样本,导致存储开销大,并且随着训练样本的增多,更新训练时间也相应变长。传统的增量学习虽然能解决历史样本存储的问题,但是又可能存在知识空间随学习过程逐渐退化的问题。本文提出基于代表样本点的混合学习方法,力图在缓解训练样本存储和算法更新训练时间成本的同时,又能较好地解决知识空间退化的问题。第 $i+1$ 次支持样本集与诊断算法更新过程如下。

Step1:利用 2.2 节和 2.1 节提出的样本扩充和压缩方法对第 $i+1$ 个新增样本集 $\boldsymbol{X}_{i+1}^{\text{new}}$ 开展数据扩充或者压缩,从而对新增样本提取满足容量要求的临时代表样本集 $\boldsymbol{P}'_{i+1}$,并计算每个临时代表样本点在 $\boldsymbol{X}_{i+1}^{\text{new}}$ 中的 $N(\boldsymbol{p}'_{\text{new}},\boldsymbol{P}'_{i+1})$。

Step2:将 $\boldsymbol{P}'_{i+1}$ 和原有代表样本集 $\boldsymbol{P}_i$ 合并后,按照式(4)对合并后的样本集进行元素合并,得到新的临时代表点。

Step3:不断重复 Step2 直至元素个数满足要求,即可获得用于第 $i+1$ 次算法更新的最终代表样本集 $\boldsymbol{P}_{i+1}$。

$$\begin{cases}\boldsymbol{p}_{\text{new}}^{\text{c}}=\boldsymbol{x}_k^{\text{c}},k=\arg\max\{N(\boldsymbol{p}_i,\boldsymbol{P}'_{i+1}),N(\boldsymbol{p}_j,\boldsymbol{P}_i)\}\\ \boldsymbol{p}_{\text{new}}^{\text{r}}=\dfrac{N(\boldsymbol{p}_i,\boldsymbol{P}'_{i+1})\boldsymbol{p}_i^{\text{r}}+N(\boldsymbol{p}_j,\boldsymbol{P}_i)\boldsymbol{p}_j^{\text{r}}}{N(\boldsymbol{p}_i,\boldsymbol{P}'_{i+1})+N(\boldsymbol{p}_j,\boldsymbol{P}_i)}\\ N(\boldsymbol{p}_{\text{new}},\boldsymbol{P}_{i+1})=N(\boldsymbol{p}_i,\boldsymbol{P}'_{i+1})+N(\boldsymbol{p}_j,\boldsymbol{P}_i)\\ \boldsymbol{p}_i\in\boldsymbol{P}'_{i+1},\boldsymbol{p}_j\in\boldsymbol{P}_i\end{cases}\tag{4}$$

3　案例应用

Coraddu 等利用 Combined Diesel Electric And Gas 公司建立的护卫舰推进系统仿真模型开展了大量的数值仿真,并获得了丰富的推进系统运行仿真数据[13-14]。本文利用 Coraddu 等在加利福尼亚大学尔湾分校机器学习数据库中提供的数据验证第 2 节所研究方法的有效性。根据文献[14],燃气机压缩系统退化量 kM_{c} 和燃气机总体退化量 kM_{t} 能够较好地表征推进系统故障状态,但这两个参数需要利用 16 种信号综合建模获取。

按照 kM_{c} 和 kM_{t} 的取值,当满足 $kM_{\text{c}}\in[0.95,0.97)\cup kM_{\text{t}}\in[0.975,0.985)$ 时认为系统处于故障状态,机内测试设备(Built-In Test Equipment, BITE)需要及时报警;否则认为系统处于正常状态,BITE 无须报警。

3.1　数据准备

相关研究表明,当用于分类的特征属性过多时可能降低分类效果,因此本文选用了如表 1 所示的 7 种测试信号开展系统故障诊断。需要说明的是,选用这些信号并不能说明这些信号是最佳信号,仅能说明这些信号能够较好地验证本文所述方法的有效性。另外为了绘图展示的方便,后文仅选用左侧螺旋桨推进扭矩和涡轮机出口温度两个参数来绘制二维图形。

表 1　推进器监测信号

Tab. 1　Monitored signals of the propulsion plant

编号	信号类型
1	左侧螺旋桨推进扭矩
2	涡轮机出口温度
3	压缩机空气入口温度
4	压缩机空气出口温度
5	涡轮机出口压力
6	压缩机空气入口压力
7	压缩机空气出口压力

为了模拟设计改进过程中因受限于时间和经

费,仅能获取系统部分运行状态数据的情况,本文利用均匀抽样提取了如表2所示的两批训练和测试数据。第一批数据用于模拟原有数据,第二批数据用于模拟新增加数据。

表2 训练与测试用样本集样本数量

Tab. 2 Data size of the training and testing data

数据类型		第一批	第二批
正常状态	训练数据	226	40
	测试数据	9008	134
故障状态	训练数据	225	40
	测试数据	9008	134

3.2 评价标准

对于BITE设计,故障检测率(Fault Detection Rate,FDR)以及虚警率(Fault Alarm Rate,FAR)是常用评价指标[7]。在概念上,故障检测率与查全率概念相似,用于衡量故障成功检测的概率;虚警率可视为查准率的余集,表征了正常状态被识别为故障状态的概率。

为了与衡量不平衡数据分类效果的F测度相对应,定义如式(5)所示的损失函数,用于对故障检测率和虚警率开展综合衡量。

$$FL = \gamma FAR + \eta(1 - FDR) \tag{5}$$

式中,γ 与 η 分别表示虚警和漏检造成的损失。因为损失函数仅用来定量刻画实际损失,γ 与 η 不必具有实际意义,本文假设虚警和漏检造成的损失相当,并且 $\gamma = \eta = 2$。

为了评价结果的客观性,采用经典支持向量机(Support Vector Machine, SVM)方法开展故障诊断,未专门研究SVM改进算法改进,而是直接利用MATLAB 2010a软件提供的svmtrain( )函数和svmclassify( )函数开展状态分类,将函数设置高斯核函数,方差0.2,并选用序贯最小优化(Sequential Minimal Optimization, SMO)优化函数作为超平面分类函数。

3.3 性能评估

为了全面验证所提方法有效性,设计两个验证案例:案例A用来验证所提方法在处理数据不平衡方面的应用效果;案例B用来验证基于样本点的混合学习方法在分类器算法更新方面的应用效果。

3.3.1 案例A

Case1:首先利用2.2节所提方法扩大故障状态样本量,使故障状态样本量与正常状态样本量相同。然后利用2.1节所提方法对扩充后的故障样本和原始正常样本进行数据压缩,并设定代表样本集的容量限制为90~100。最后将处理后的代表样本集作为训练样本。

Case2:首先利用2.1节所提方法对原始正常状态样本进行数据压缩,然后和原始故障状态样本组成训练样本集。

Case3:首先利用2.2节所提数据扩充方法扩大故障状态训练样本量,使故障状态样本量与正常状态样本量相当,然后和原始正常状态样本组成训练样本集。

Case4:不对原始训练样本进行任何处理,直接将其作为训练样本集。

利用上述4个不同训练样本集训练得到SVM分类器之后,利用3.1节处理的第一批测试数据测试SVM分类效果。为了消除数据处理随机性对分类测试效果的影响,利用一次数据处理结果重复开展了20次SVM训练,并分别开展分类测试,分类效果评价标准的算术平均值如表3所示。

表3 案例A诊断分类平均结果

Tab. 3 Diagnosis results of scenario A

	Case1	Case2	Case3	Case4
FDR	0.537 7	0.425 4	0.673 5	0.410 1
FAR	0.018 6	0.219 2	0.031 8	0.000 0
FL	0.961 9	1.587 6	0.716 6	1.179 1
F测度	0.035 8	0.317 8	0.058 0	0.000 0
SV个数	180.25	73.00	453.10	253.00

对比表中数据可知:①前3个示例训练样本集中正常状态与故障状态的样本数量比分别为91∶90, 45∶40, 200∶226,从训练样本数量上看,三种方法都能较好地解决数据不平衡问题;②Case2的故障检测率虽然与Case4接近,但虚警率明显偏高,主要是由于Case2用于训练的数据较少造成分类器识别效果不佳;③Case3的故障检测率比Case4高出50%,同时虚警率也相对较低,主要是因为训练样本量较多,如果不考虑支持向量(Support Vector,SV)个数,可以认为Case3中的数据处理方法最佳;④Case1中故障检测率比Case4提高了将近25%,虚警率也不高,并且与Case3相比,SV个数明显较少;⑤综合SV个数以及分类效果两个评价指标,Case1中的双向数据处理方法既提高了SVM分类准确度,SV的个数也得到了缩减,可以降低诊断算法

对机内测试系统存储容量的要求，缩短故障诊断时间。

另外，对比 4 个示例的 F 测度和损失函数值可知：损失函数不会因为某个单一因素的极端取值而得到极端结果（如 Case4 中 *FAR* 对 F 测度的影响），因此从故障诊断的角度出发，损失函数能比 F 测度更好地综合反映故障漏检和错检对装备维修造成的损失。

综上所述，由于评价标准不同，在处理具体问题时，是采用单侧数据扩充或压缩，还是采用双侧处理，需要根据实际情况具体判断，但是无论采用何种处理方式，本文所提方法均能较好地解决数据不平衡问题。

3.3.2　案例 B

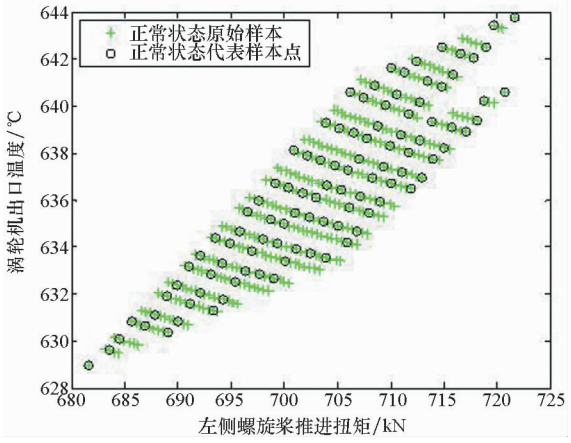
Case1：利用案例 A 中 Case4 的支持向量和表 2所列的第二批原始训练数据更新 SVM，该示例用于模拟简单增量学习过程。

Case2：利用 2.3 节所提混合学习方法更新 SVM，并将正常和故障状态的代表样本集容量分别设置为 90 ~ 100。

Case3：将表 2 所列的两批原始训练样本合并组成完整训练样本集，利用传统的批量学习方法更新 SVM。

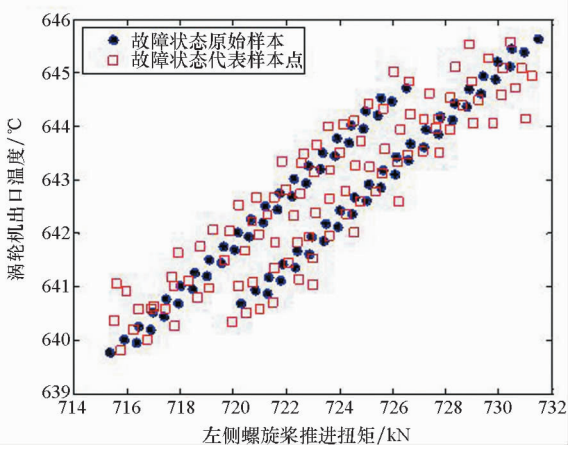
案例 B 中 Case2 和 Case3 用到的训练样本集分别如图 1 中原始样本和代表样本所示。从图中可以看出原始数据的分布特征在代表样本集中得到了较好的保留。

在更新 SVM 之后，利用表 2 所列的第二批测试数据验证 SVM 更新效果。同样，为了降低数据处理随机过程对测试效果的影响，案例 B 也利用一次数据处理结果进行了 20 次训练和测试，分类效果的算术平均值如表 4 所示。图 2 直观展示了 Case1 的原支持向量的分布情况。



（a）正常状态下原始样本与代表样本点特征分布

（a）Distribution of raw data and delegates in normal state



（b）故障状态下原始样本与代表样本点特征分布

（b）Distribution of raw data and delegates in fault state

图 1　原始数据与代表样本集分布情况

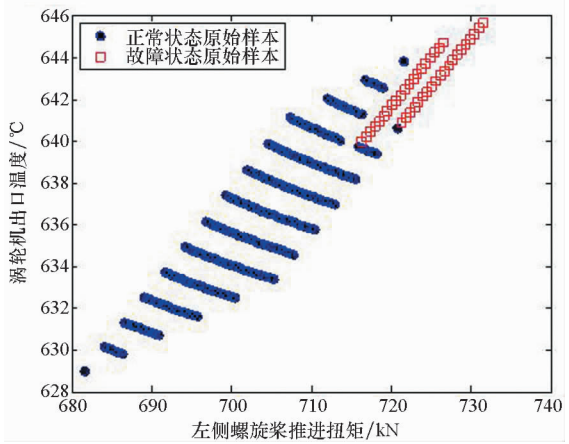
Fig. 1　Distribution of raw data and delegates set

表 4　案例 B 诊断分类平均效果

Tab. 4　Diagnosis results of scenario B

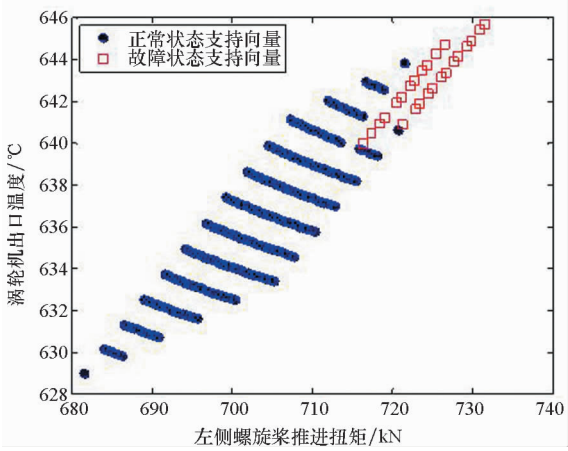
	Case1	Case2	Case3
FDR	0.656 7	0.669 8	0.656 7
FAR	0.083 3	0.008 2	0.083 3
FL	0.853 2	0.676 8	0.853 2
F 测度	0.134 1	0.016 0	0.134 1
SV 个数	469.00	196.85	465.00

比较表 4 和图 2 数据可知：①除了少量故障状态数据外，绝大部分原始训练样本都被处理为 Case1 的原始支持向量，因此可以认为 Case1 和 Case3 的训练样本几乎相同，从而造成 Case1 和 Case3 的分类测试结果完全相同；②由于训练样本数量和分布空间的扩充，无论采用哪种方法，故障诊断系统的分类效果与案例 A 相比都得到了



（a）原始样本特征分布

（a）Distribution of raw data



(b) 支持向量特征分布

(b) Distribution of SVs

图 2 Case1 原支持向量与第一批原始样本分布

Fig. 2 Distribution of the original SVs and the 1st batch raw data in Case1

明显提高;③无论是与 Case1 还是 Case3 相比,Case2 在保持与 Case1 和 Case3 几乎相同的故障检测率的同时,虚警率有了明显的下降,并且需要的支持向量数量也明显减少。

4 结论

首先针对数据不平衡问题提出数据压缩方法和数据扩充方法,其中基于密度的大样本数据压缩既能生成满足样本量要求的代表样本集,又能保持较好的原始数据分布规律;基于人工免疫的小样本数据扩充方法在丰富样本数量及分布特征的同时,又有效降低了噪声数据的引入。然后针对分类器更新训练需求给出了一种新的增量式批量学习方法——基于样本代表点的混合学习方法,既可以降低训练样本硬件存储要求,又能缩短分类器更新训练时间,同时保持较高的分类准确性。最后利用公开仿真数据验证了所提方法的有效性。理论分析和仿真结果表明:所提方法可以有效支持基于数据的故障诊断算法更新,并且对其他领域的机器学习应用问题研究也有一定的借鉴意义。

参考文献(References)

[1] 陈彧赟, 侯博文, 何章鸣, 等. 数据驱动的复杂系统非预期故障诊断通用过程模型[J]. 国防科技大学学报, 2017, 39(6): 126 – 133.
CHEN Yuyun, HOU Bowen, HE Zhangming, et al. General process model for unanticipated fault diagnosis of complex system based on data driven [J]. Journal of National University of Defense Technology, 2017, 39(6): 126 – 133. (in Chinese)

[2] 赵晨旭. 测试性增长试验理论与方法研究[D]. 长沙: 国防科技大学, 2016.
ZHAO Chenxu. Research on testability growth test theory and method [D]. Changsha: National University of Defense Technology, 2016. (in Chinese)

[3] Carbonara L, Borrowman A. A comparison of batch and incremental supervised learning algorithms [C]// European Symposium on Principles of Data Mining and Knowledge Discovery, 1998: 264 – 272.

[4] Leung K, Cheong F. A simple artificial immune system (SAIS) for generating classifier systems [C]// AI 2006: Advances in Artificial Intelligence, 2006: 151 – 160.

[5] 曾文华, 马健. 一种新的支持向量机增量学习算法[J]. 厦门大学学报(自然科学版), 2002, 41(6): 687 – 691.
ZENG Wenhua, MA Jian. A new incremental learning algorithm for support vector machine[J]. Journal of Xiamen University (Natural Science), 2002, 41(6): 687 – 691. (in Chinese)

[6] 李艳玲, 郭文普, 徐东辉. 一种不平衡数据的分类方法[J]. 中国电子科学研究院学报, 2012, 7(3): 246 – 251.
LI Yanling, GUO Wenpu, XU Donghui. A classification method for imbalanced data[J]. Journal of CAEIT, 2012, 7(3): 246 – 251. (in Chinese)

[7] Hulse J V, Khoshgoftaar T M, Napolitano A. Experimental perspectives on learning from imbalanced data [C]// Proceedings of the Twenty-Fourth International Conference on Machine Learning, 2007: 935 – 942.

[8] Sun Y M, Kamel M S, Wong A K C, et al. Cost-sensitive boosting for classification of imbalanced data [J]. Pattern Recognition, 2007, 40(12): 3358 – 3378.

[9] Akin B, Choi S, Orguner U, et al. A simple real-time fault signature monitoring tool for motor-drive-embedded fault diagnosis systems [J]. IEEE Transactions on Industrial Electronics, 2011, 58(5): 1990 – 2001.

[10] Ester M, Kriegel H P, Sander J, et al. A density-based algorithm for discovering clusters in large spatial databases with noise[C]// Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining, 1996: 226 – 231.

[11] Li J, Gao X B, Jiao L C. A GA-based clustering algorithm for large data sets with mixed numeric and categorical values[C]//Proceedings of the Fifth International Conference on Computational Intelligence and Multimedia Applications, 2003: 101 – 107.

[12] Watkins A. AIRS: a resource limited artificial immune classifier[D]. Mississippi, USA: Mississippi State University, 2001.

[13] Altosole M, Benvenuto G, Figari M, et al. Real-time simulation of a COGAG naval ship propulsion system [J]. Proceedings of the Institution of Mechanical Engineers, Part M: Journal of Engineering for the Maritime Environment, 2009, 223(1): 47 – 62.

[14] Coraddu A, Oneto L, Ghio A, et al. Machine learning approaches for improving condition-based maintenance of naval propulsion plants[J]. Proceedings of the Institution of Mechanical Engineers, Part M: Journal of Engineering for the Maritime Environment, 2016, 230(1): 136 – 153.