

基于 RDK-ELM 的航空发动机控制系统故障诊断

陈虹璐, 黄向华

(南京航空航天大学 能源与动力学院, 南京 210016)

摘要: 为保持较高诊断正确率, 缩短训练时间, 满足航空发动机故障诊断对于实时性和高诊断率的需求, 提出 1 种对深度核极限学习机的简约改进方法。输入数据中随机选取部分数据作为支持向量, 结合深度学习网络的多层结构, 完成了对输入样本的特征提取, 通过核函数实现了高维空间映射分类。数字仿真表明: 算法分类正确率高, 训练时间短, 可应用于航空发动机控制系统的故障诊断。

关键词: 控制系统; 故障诊断; 深度学习; 极限学习机; 简约改进; 航空发动机

中图分类号: V233.7

文献标识码: A

doi: 10.13477/j.cnki.aeroengine.2020.02.006

Fault Diagnosis of Aeroengine Control System Based on RDK-ELM

CHEN Hong-lu, HUANG Xiang-hua

(College of Energy and Power Engineering, Nanjing University of Aeronautics & Astronautics, Nanjing 210016, China)

Abstract: In order to maintain the high diagnostic accuracy and shorten the training time, a method of Reduced Deep Kernel Extreme Learning Machine (RDK-ELM) was proposed for meeting the needs of aeroengine fault diagnosis for real-time and high diagnostic rate. Partial data was selected randomly as support vector in the input data. The feature extraction of the input sample was accomplished by combining the multi-layer structure of the deep learning network. The high-dimensional spatial mapping classification was realized by kernel function. Digital simulation shows that the algorithm has high classification accuracy and short training time, and can be applied to fault diagnosis of aeroengine control system.

Key words: control system; fault diagnosis; deep learning; extreme learning machine; reduced method; aeroengine

0 引言

现代航空发动机数字控制系统主要由传感器、执行机构及电子控制器等组成^[1], 其中, 传感器和执行机构组件繁杂, 容易发生故障, 是控制系统中十分重要又脆弱的环节, 因此必须对传感器和执行机构进行实时故障诊断, 从而采取必要的容错控制措施以保障发动机工作性能不受影响。

故障诊断方法主要有 3 种: 基于模型的、基于人工智能的和基于数据驱动的方法。基于模型的故障诊断方法从 1977 年被 Spang 等^[2]提出, 目前已较为成熟。在 NASA 和 MTU 合作进行的关于先进智能发动机传感器和执行器性能要求的研究中所提出的未来

智能发动机发展的可行技术中, 就包括用于提高发动机的气路部件性能和延长发动机的在役寿命的通过基于模型的控制和监控技术^[3]。基于模型的方法应用发动机部件级模型作为机载模型, 通过扩展卡尔曼滤波器(Extended Kalman Filter, EKF)自适应跟踪发动机感测信号的输出, 在发动机发生故障时, EKF 的信号估计值偏离发动机实际信号值产生残差, 通过计算对残差进行分析即可诊断发动机的故障类型^[4]。

近年来人工智能算法兴起, 不少学者将 BP 神经网络、径向基网络、自联想网络、支持向量机等算法应用于故障诊断中^[5-8], 其中基于梯度训练的神经网络算法泛化能力较差, 易陷入局部最优; 而支持向量机参

收稿日期: 2019-03-14 基金项目: 国家自然科学基金(51576097)资助

作者简介: 陈虹璐(1993), 女, 硕士, 主要从事系统控制与仿真工作; E-mail: 190373810@qq.com。

引用格式: 陈虹璐, 黄向华. 基于 RDK-ELM 的航空发动机控制系统故障诊断[J]. 航空发动机, 2020, 46(2): 34-40. CHEN Honglu, HUANG Xianghua. Fault diagnosis of aeroengine control system based on RDK-ELM[J]. Aeroengine, 2020, 46(2): 34-40.

数选择复杂,训练速度慢,不适于故障诊断的工程应用。Huang 等^[9]提出的极限学习机(Extreme Learning Machine, ELM)算法是1种新型前向神经网络算法,无需迭代,学习速度快、精度高且易于调参;随着核函数的引入,ELM 获得了更高精度^[10]。Wong 等^[11]提出具有深度学习网络结构的多层核极限学习机(Multilayer Kernel Extreme Learning Machine, ML-KELM);逢珊等^[12]将其应用于航空发动机的气路部件故障诊断中,获得了更优的诊断精度。但深度学习结构通过多层网络逐层进行输入数据的特征提取,计算耗时严重,对需要高实时性的发动机故障诊断来说是不容忽视的缺陷。

针对深度核极限学习机的训练时间过长的不足,本文将 Deng 等^[13]提出的核极限学习机的简约方法与深度学习网络结构结合,提出简约的深度核极限学习机(Reduced Deep Kernel Extreme Learning Machine, RDK-ELM)。该算法利用随机支持向量缩短算法训练时间,并保留深度学习网络结构的高精度优点,对 EKF 产生的残差数据进行模式识别分类,实现航空发动机控制系统传感器与执行器的故障诊断。

1 多层核极限学习机的推导

1.1 极限学习机

极限学习机是 Huang 等提出的1种新的单隐层前向神经网络的快速学习方法。在 ELM 算法中,给定训练样本: $\{(\mathbf{x}_i, \mathbf{t}_i) | \mathbf{x}_i \in \mathbf{R}^d, \mathbf{t}_i \in \mathbf{R}^m\}_{i=1}^N$, 其中, N 为样本数量, d 为维数, m 为输出节点数,对于如本文所述的故障诊断分类问题, m 为类别数。单隐层前向神经网络的 ELM 输出函数为

$$\mathbf{f}_{\text{ELM}}(\mathbf{x}) = \sum_{i=1}^L \beta_i h(\mathbf{x}, \mathbf{a}_i, b_i) = \mathbf{h}(\mathbf{x}) \boldsymbol{\beta} \quad (1)$$

式中: L 为隐藏节点的数目; $\boldsymbol{\beta} = [\beta_1, \dots, \beta_L]^T$, 为输出权重向量; $h(\cdot)$ 为激活函数; $\mathbf{a}_i$ 为隐藏节点的权重; b_i 为隐藏节点的偏差。

式(1)写成矩阵形式为

$$\mathbf{H} \boldsymbol{\beta} = \mathbf{T} \quad (2)$$

其中, $\mathbf{H}$ 为隐层输出矩阵

$$\mathbf{H} = \begin{bmatrix} h(\mathbf{a}_1, \mathbf{x}_1, b_1) & \cdots & h(\mathbf{a}_L, \mathbf{x}_1, b_L) \\ \vdots & \ddots & \vdots \\ h(\mathbf{a}_1, \mathbf{x}_N, b_1) & \cdots & h(\mathbf{a}_L, \mathbf{x}_N, b_L) \end{bmatrix} \quad (3)$$

$$\mathbf{T} = \begin{bmatrix} \mathbf{t}_1^T \\ \vdots \\ \mathbf{t}_N^T \end{bmatrix} \quad (4)$$

输出权值 $\boldsymbol{\beta}$ 可通过寻找给定系统的最小二乘解获得,即求解式(2)的最小范数的最小二乘解为

$$\boldsymbol{\beta} = \begin{cases} \mathbf{H}^T \left(\frac{\mathbf{I}}{C} + \mathbf{H} \mathbf{H}^T \right)^{-1} \mathbf{T} & N \geq L \\ \left(\frac{\mathbf{I}}{C} + \mathbf{H} \mathbf{H}^T \right)^{-1} \mathbf{H}^T \mathbf{T} & N \leq L \end{cases} \quad (5)$$

式中: C 为正则化系数。

1.2 核极限学习机

如果激活函数 $h(\cdot)$ 是1个隐函数,则可以对 ELM 算法应用 Mercer 条件,并引入核矩阵^[10]

$$\boldsymbol{\Omega} = \mathbf{H} \mathbf{H}^T \quad (6)$$

即对任意每2个训练样本之间求核函数

$$\Omega_{ij} = \kappa(\mathbf{x}_i, \mathbf{x}_j) \quad (7)$$

将式(6)、(7)与式(5)代入式(1)中,可以得到核形式的输出函数

$$\mathbf{f}_{\text{KELM}}(\mathbf{x}) = \mathbf{h}(\mathbf{x}) \mathbf{H}^T \left(\frac{\mathbf{I}}{C} + \boldsymbol{\Omega} \right)^{-1} \mathbf{T} \quad (8)$$

$$\mathbf{h}(\mathbf{x}) \mathbf{H}^T = \begin{bmatrix} \kappa(\mathbf{x}, \mathbf{x}_1) \\ \vdots \\ \kappa(\mathbf{x}, \mathbf{x}_N) \end{bmatrix} \quad (9)$$

这样就得到了核极限学习机 KELM。

1.3 多层核极限学习机

多层核极限学习机 ML-KELM 是将核方法引入到多层极限学习机的1种算法。多层极限学习机^[14] (Multilayer Extreme Learning Machine, ML-ELM) 与深度置信网络(Deep Belief Net-work, DBN)相似, DBN 是由多个受限玻尔兹曼机^[15] (Restricted Boltzmann Machine, RBM) 叠加而成,逐层提取特征获得更抽象的概念。ML-ELM 由多个极限学习机自编码器(ELM Auto-Encoder, ELM-AE) 叠加而成,但相比于 DBN, ML-ELM 不需要对网络权值进行整体微调,且无需迭代,因此训练速度较 DBN 有显著提高。

ML-KELM 由多个核极限学习机自编码器(Kernel Extreme Learning Machine Auto-Encoder, KELM-AE) 构成, KELM-AE 的结构如图1所示。

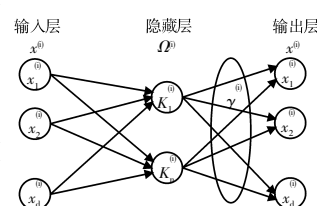


图1 KELM-AE 结构

文献[11]中给出了 ML-KELM 的详细推导过程。类似于式(7)的核函数映射计算, KELM-AE 的输入矩阵 $\mathbf{X}^{(i)}$ 通过核函数 $\kappa^{(i)}(\mathbf{x}_k, \mathbf{x}_j)$ 被映射为核矩阵 $\mathbf{\Omega}^{(i)}$, 核函数可以选用 RBF 核函数: $\kappa^{(i)}(\mathbf{x}_k, \mathbf{x}_j) = \exp\left(-\frac{\|\mathbf{x}_k - \mathbf{x}_j\|}{2\sigma_i^2}\right)$ 。其中, σ_i 为核参数。计算得到核矩阵后, 输入向量的特征提取可以表示为

$$\mathbf{\Omega}^{(i)} \widetilde{\mathbf{T}}^{(i)} = \mathbf{X}^{(i)} \quad (10)$$

式中: $\widetilde{\mathbf{T}}^{(i)}$ 为 KELM-AE 中的第 i 个变换矩阵

$$\widetilde{\mathbf{T}}^{(i)} = \left(\frac{\mathbf{I}}{C} + \mathbf{\Omega}^{(i)} \right)^{-1} \mathbf{X}^{(i)} \quad (11)$$

在变换过程的最后 1 步, 得到的数据特征 $\mathbf{X}^{(i+1)}$ 为

$$\mathbf{X}^{(i+1)} = g(\mathbf{X}^{(i)} (\widetilde{\mathbf{T}}^{(i)})^T) \quad (12)$$

式中: $g(\cdot)$ 为激活函数, 与式(1)中的 $h(\cdot)$ 都是激活函数, 可从多种激活函数类型中根据实际情况与实际问题的自定义选取, 在不同的算法中可以不相同。

在 ML-KELM 的训练过程中, 输入数据由多层 KELM-AE 逐层进行特征提取, 每层的 $\widetilde{\mathbf{T}}^{(i)}$ 和 $\mathbf{X}^{(i)}$ 都可由式(11)、(12)计算得到。最终得到的数据特征 $\mathbf{X}^{\text{final}}$ 作为 1 个 KELM 分类器的输入来计算最终分类结果, KELM 分类器的模型为

$$\mathbf{\Omega}^{\text{final}} \boldsymbol{\beta}^{\text{final}} = \mathbf{T} \quad (13)$$

式中: $\mathbf{\Omega}^{\text{final}}$ 为由 $\mathbf{X}^{\text{final}}$ 计算得到的核矩阵, 则 $\boldsymbol{\beta}^{\text{final}}$ 为

$$\boldsymbol{\beta}^{\text{final}} = \left(\frac{\mathbf{I}}{C} + \mathbf{\Omega}^{\text{final}} \right)^{-1} \mathbf{T} \quad (14)$$

2 简约深度核极限学习机的推导

2.1 简约核极限学习机

简约核极限学习机 (Reduced Kernel Extreme Learning Machine, RKELM) 是在 KELM 的基础上, 从输入样本中随机选择部分样本作为支持向量进行计算, 在保持较高精度的同时大幅提高了训练速度。

对给定的 N 个训练样本 $\mathbf{S} = \{(\mathbf{x}_i, \mathbf{t}_i) | \mathbf{x}_i \in \mathbf{R}^d, \mathbf{t}_i \in \mathbf{R}^m\}_{i=1}^N$, 给定支持向量数目 S , 则支持向量 $\mathbf{X}_S = \{\mathbf{x}_i | \mathbf{x}_i \in \mathbf{R}^d\}_{i=1}^S$ 为

$$\sum_{sv=1}^S \beta_{sv} \kappa(\mathbf{x}_i, \mathbf{x}_{sv}) = \mathbf{t}_i, i=1, 2, \dots, N \quad (15)$$

写成矩阵形式为

$$\mathbf{\Omega}_{N \times S} \boldsymbol{\beta} = \mathbf{T} \quad (16)$$

式中: $\mathbf{\Omega}_{N \times S} = K(\mathbf{X}, \mathbf{X}_S)$, 为简约的核矩阵。

式(9)在求取核矩阵时, 是对每个输入向量都进行核函数计算, 而此处简约的核矩阵仅对 S 个支持向量进行核函数计算, 计算量大大简化。 $\boldsymbol{\beta} = [\beta_1, \beta_2, \dots, \beta_S]$

是输出权重矩阵

$$\boldsymbol{\beta} = \left(\frac{\mathbf{I}}{C} + \mathbf{\Omega}_{N \times S}^T \mathbf{\Omega}_{N \times S} \right)^{-1} \mathbf{\Omega}_{N \times S}^T \mathbf{T} \quad (17)$$

2.2 简约深度核极限学习机

在 ML-KELM 的计算中, 每层 KELM-AE 的特征提取都是对全部输入数据计算核矩阵, 虽然可以保证高精度, 但付出了高耗时的代价。对此, 本文提出简约的深度核极限学习机 (Reduced Deep Kernel Learning Machine, RDK-ELM), 每层特征提取由 RKELM-AE 完成。对于第 i 层 RKELM-AE, 编码器从输入数据 $\mathbf{X}^{(i)}$ 中随机选择 $L^{(i)}$ 个支持向量, 根据式(16)、(17)计算变换矩阵 $\widetilde{\mathbf{T}}^{(i)}$, 逐层完成特征提取, 最后通过 1 个 RKELM 分类器计算出分类结果。RKELM-AE 算法的计算过程见表 1, RDK-ELM 算法的计算过程见表 2。

表 1 RKELM-AE 算法的计算过程

算法 1 第 i 层 RKELM-AE
输入: 输入矩阵 $\mathbf{X}^{(i)}$, 支持向量数目 $S^{(i)}$, 正则化因子 $C^{(i)}$, 核参数 $\sigma^{(i)}$, 激活函数 $g(\cdot)$
输出: 新数据特征 $\mathbf{X}^{(i+1)}$, 变换矩阵 $\widetilde{\mathbf{T}}^{(i)}$
Step 1: 随机选择支持向量 $\mathbf{X}_S^{(i)}$
Step 2: 计算核矩阵 $\mathbf{\Omega}^{(i)} \leftarrow K(\mathbf{X}^{(i)}, \sigma^{(i)}, \mathbf{X}_S^{(i)})$
Step 3: 计算变换矩阵:
$\widetilde{\mathbf{T}}^{(i)} = \left(\frac{\mathbf{I}}{C^{(i)}} + \mathbf{\Omega}_{N \times S^{(i)}}^T \mathbf{\Omega}_{N \times S^{(i)}} \right)^{-1} \mathbf{\Omega}_{N \times S^{(i)}}^T \mathbf{X}^{(i)}$
Step 4: 更新数据特征 $\mathbf{X}^{(i+1)} \leftarrow g(\mathbf{X}^{(i)} \widetilde{\mathbf{T}}^{(i)})^T$
返回 $\mathbf{X}^{(i+1)}, \widetilde{\mathbf{T}}^{(i)}$

表 2 RDK-ELM 算法的计算过程

算法 2 RDK-ELM 算法
输入: 输入矩阵 $\mathbf{X}^{(1)}$, 输出矩阵 $\mathbf{T}$, 层数 N , 各层支持向量数目 $S^{(i)}$, 各层正则化因子 $C^{(i)}$, 各层核参数 $\sigma^{(i)}$, 各层激活函数 $g(\cdot)$
输出: 特征矩阵 $\mathbf{X}^{\text{final}}$, 输出权重矩阵 $\boldsymbol{\beta}^{\text{final}}$
Step 1: 随机选择支持向量 $\mathbf{X}_S^{(1)}$
Step 2: 计算 $\mathbf{X}^{(2)}, \widetilde{\mathbf{T}}^{(1)}$: RKELM-AE($\mathbf{X}^{(1)}, \mathbf{X}_S^{(1)}, C_1, \sigma_1, g_1$)
Step 3: for $i = 2 : (N - 1)$ do:
计算 $\mathbf{X}^{(i+1)}, \widetilde{\mathbf{T}}^{(i-1)}$: RKELM-AE
Step 4: $\mathbf{X}^{(i+1)} \leftarrow \mathbf{X}^{(i+1)}$
Step 5: 随机选择支持向量 $\mathbf{X}_S^{(i+1)}$
Step 6: 计算核矩阵 $\mathbf{\Omega}^{(i+1)} \leftarrow K(\mathbf{X}^{(i+1)}, \sigma^{(i+1)}, \mathbf{X}_S^{(i+1)})$
Step 7: 计算输出权重矩阵 $\boldsymbol{\beta}^{\text{final}}$:
$\boldsymbol{\beta}^{\text{final}} = \left(\frac{\mathbf{I}}{C^{(i+1)}} + \mathbf{\Omega}_{N \times S^{(i+1)}}^T \mathbf{\Omega}_{N \times S^{(i+1)}} \right)^{-1} \mathbf{\Omega}_{N \times S^{(i+1)}}^T \mathbf{X}^{\text{final}}$
返回 $\mathbf{X}^{\text{final}}, \boldsymbol{\beta}^{\text{final}}$

相比于 ML-KELM 算法,RDK-ELM 算法中每层 RKELM-AE 都选择部分样本作为支持向量,当支持向量为全部样本时,RDK-ELM 算法与 ML-KELM 算法等效^[3]。所以通过合理选择各层支持向量的数量,可以得到训练时间短、分类精度高的 RDK-ELM 算法。

3 算法在通用分类数据上的比较验证

为验证本文提出的 RDK-ELM 算法在速度与精度上的性能,首先将其与基本 ELM、KELM、RKELM 以及基于深度学习结构的 ML-KELM 算法在 4 个典型 UCI 分类数据集上进行综合比较。4 个 UCI 数据集样本的相关参数见表 3。

表 3 UCI 分类数据集的数据				
数据集	训练样本数	测试样本数	属性数	类别数
Satimage	4435	2000	36	6
Segment	2100	210	18	7
Wine	118	60	13	3
Waveform	4000	1000	20	3

研究采用的计算机配置为:Core i5-8250U,内存 8G。算法验证的软件平台为:Matlab R2016b。

算法参数设置:所有算法的激活函数均设置为 Sigmoid 函数,核函数均设置为 RBF 核函数。试验验证发现,ML-KELM 和 RDK-ELM 算法均在隐含层为 2 层时精度最好,故层数均设置为 3。正则化因子 C 和核参数 σ 的取值在 $10^{-7} \sim 10^7$ 范围内经过试验选取最优设置。具体每种算法的正则化因子、核参数、支持向量数的设置见表 4~8。

表 4 ELM 算法的隐含节点数设置			
Satimage	Segment	Wine	Waveform
240	110	15	200

表 5 KELM 算法的参数设置				
参数	Satimage	Segment	Wine	Waveform
σ	10	10	10^2	10
C	10^2	10^2	1	0.1

表 6 RKELM 算法的参数设置				
参数	Satimage	Segment	Wine	Waveform
σ	1	10	1	10
C	10^5	10^5	10	10^3
L	400	10^2	10^2	10^2

表 7 ML-KELM 算法的参数设置				
参数	Satimage	Segment	Wine	Waveform
σ	$1, 10^2, 10^2$	$10^2, 10^3, 10$	$10^2, 10^3, 0.1$	$10, 10^3, 10^3$
C	$0.1, 10^2, 10^2$	$1, 10, 10^2$	$0.1, 0.1, 10^2$	$10^2, 10^4, 1$

表 8 RDK-ELM 算法的参数设置				
参数	Satimage	Segment	Wine	Waveform
σ	$10^2, 10^2, 10^4$	$10, 10, 10^3$	$1, 10^2, 10^3$	$1, 10^2, 10^5$
C	$10, 10^2, 10^6$	$0.1, 10, 10^5$	$10, 1, 10^2$	$10, 1, 10^2$
L	600, 600, 800	300, 300, 400	40, 40, 60	300, 300, 500

5 种算法在 4 个 UCI 分类数据集上的性能结果见表 9~11。由于 ELM 算法需要随机生成权重与偏差,RKELM 和 RDK-ELM 算法需要随机选择支持向量,其运算结果具有不确定性,因此这 3 种算法的性能参数是取 100 次运算的平均值,KELM 和 ML-KELM 则是 1 次的运算结果。表中,Tr.time 为 Training time 的缩写,表示训练时间,s;Ts.rate 为 Testing rate 的缩写,表示测试分类正确率。

表 9 ELM 与 KELM 的训练时间与分类正确率				
数据集	ELM		KELM	
	Tr.time/s	Ts.rate/%	Tr.time/s	Ts.rate/%
Satimage	0.1457	88.10	1.5704	89.35
Segment	0.0217	90.17	0.2691	95.24
Wine	0.000682	90.28	0.000763	94.78
Waveform	0.0927	85.53	1.1725	87.40

表 10 RKELM 与 ML-KELM 的训练时间与分类正确率				
数据集	RKELM		ML-KELM	
	Tr.time/s	Ts.rate/%	Tr.time/s	Ts.rate/%
Satimage	0.0887	89.02	14.8979	90.75
Segment	0.0096	93.91	2.3033	96.67
Wine	0.000708	92.98	0.0026	96.67
Waveform	0.0165	86.85	12.4264	87.70

表 11 RDK-ELM 的训练时间与分类正确率		
数据集	RDK-ELM	
	Tr.time/s	Ts.rate/%
Satimage	0.8346	89.19
Segment	0.1692	94.97
Wine	0.0015	93.95
Waveform	0.3398	87.10

从表中可见,ML-KELM 算法的分类正确率在全部数据集中都是最高的,说明基于深度学习网络结构的 KELM 算法具有较好的泛化性能和分类能力。本

文所提出的 RDK-ELM 算法的正确率与 KELM 算法的差距非常小,且样本数据越大,差距越小,可以说二者的分类能力几乎持平。

但从训练速度方面而言,RDK-ELM 算法的训练速度明显比 KELM 算法的更快。在 5 种算法中 RKELM 算法的训练速度最快,但精度不如 RDK-ELM 算法的;而 ML-KELM 虽然精度非常高,训练速度却最慢,对工程应用来说是不可回避的缺陷。

RDK-ELM 算法通过深度学习网络结构,在 RKELM 算法的基础上获得了更好的泛化性能与分类能力,同时比 KELM 和 ML-KELM 算法具有更快的训练速度。在这 4 个 UCI 分类数据集上的试验表明,本文提出的 RDK-ELM 算法是同时具有较好的精度与速度的算法,可用于航空发动机控制系统中的传感器与执行器的故障诊断。

4 算法在涡扇发动机故障诊断中的验证

为验证 RDK-ELM 算法在航空发动机控制系统故障诊断中的性能,本文以某型涡扇发动机为研究对象,采集数据样本进行故障诊断仿真试验。

4.1 故障模式和故障数据样本

实际的航空发动机传感器与执行器的故障数据是非常难获得的。为了进行故障诊断的仿真试验,采用文献[16]中的方法,利用某型涡扇发动机部件级模型模拟发动机的故障,通过自适应机载模型产生 EKF 残差,采集故障状态下的 EKF 残差数据,作为训练样本和测试样本数据集。

本文主要研究发动机传感器和执行器的故障诊断,但实际情况中也可能发生气路部件的故障,为验证算法区分控制系统故障与气路部件故障的能力,选取高压压气机(High Pressure Compressor, H_{PC})和高压涡轮(High Pressure Turbine, H_{PT})2 个气路部件的故障进行研究。选取的传感器有:低压转子转速传感器(N_l),高压转子转速传感器(N_h),高压压气机出口压力传感器(P_3);选取的执行器为主燃油阀 W_f 。研究假设传感器与执行器最多只有 2 个同时发生故障,则有 12 种故障模式,6 种单故障研究样本见表 12,6 种双故障研究样本见表 13。表中“F”表示该部件发生故障,“-”表示无故障。

对于实际应用于发动机的故障诊断算法来说,发动机的飞行高度、马赫数及其工作状态会影响发动机

表 12 单故障研究样本

参数	1	2	3	4	5	6
N_l	F	-	-	-	-	-
N_h	-	F	-	-	-	-
P_3	-	-	F	-	-	-
W_f	-	-	-	F	-	-
H_{PC}	-	-	-	-	F	-
H_{PT}	-	-	-	-	-	F

表 13 双故障研究样本

参数	7	8	9	10	11	12
N_l	F	F	F	-	-	-
N_h	F	-	-	F	F	-
P_3	-	F	-	F	-	F
W_f	-	-	F	-	F	F

感测信号的值,是非常重要的影响因素。但研究表明,发动机的状态对于故障诊断算法的可行性研究影响不大,在额定和最大工作状态下,由算法得到的性能结果相同^[16],因此本文仅选择在发动机高度 $H=0$,飞行马赫数 $Ma=0$,最大工作状态下进行数字仿真试验验证算法的可行性。

发动机传感器、执行器、气路部件的故障程度根据文献 [4] 所给定的数值注入发动机模型进行模拟。 N_l 、 N_h 传感器的故障偏置范围为 $-0.5\% \sim -2.5\%$, P_3 传感器的故障偏置范围为 $-2\% \sim -10\%$, W_f 执行器的故障偏置范围为 $-1\% \sim -8\%$, H_{PC} 和 H_{PT} 的故障偏置范围为 $-1\% \sim -5\%$ 。故障偏置数值通过随机数的方式在上述范围内产生,注入模型仿真来生成故障 EKF 残差数据样本。

为避免类不平衡的问题,对于表 12 中 1~6 的单故障情况,每种情况采集 200 个样本,对于表 13 中 7~12 的双故障情况,每种情况采集 400 个样本。则共有 1200 个单故障样本,2400 个双故障样本,总计 3600 个样本。选取其中的 80% 作为训练样本,其余 20% 为测试样本。即训练样本数 2880,测试样本数 720。

由于 9 个 EKF 残差数据中有转速、压强、温度,各参数单位不一致且数量级相差较大,会影响算法进行故障诊断的精度,因此对 EKF 残差数据进行归一化处理

$$x'_{(i)} = \frac{x_{(i)} - x_{(i)\min}}{x_{(i)\max} - x_{(i)\min}} \quad (16)$$

4.2 算法的故障诊断试验验证比较

为验证 RDK-ELM 算法在故障诊断中精度与速度的综合优势,将 ELM、KELM、RKELM、ML-KELM 和 RDK-ELM 算法应用于发动机控制系统故障仿真数据的诊断分类中进行比较分析。

算法参数设置:所有算法的激活函数均设置为 Sigmoid 函数,核函数均设置为 RBF 核函数。ML-KELM 和 RDK-ELM 算法均在隐含层为 2 层时精度最好,故层数均设置为 3。正则化因子 C 和核参数 σ 的取值在 $10^{-7} \times 10^7$ 范围内经过试验选取最优设置。ELM 算法的隐含层节点设置为 200;KELM 算法的 $\sigma=10, C=10^5$;RKELM 算法 $\sigma=1, C=10^3, L=200$;ML-KELM 算法 3 层核参数选择为 $\sigma=[10, 10^4, 10^3]$, 正则化因子 $C=[10^2, 10^{-1}, 10]$;RDK-ELM 算法 3 层核参数选择为 $\sigma=[10^2, 10^2, 10^4]$, 正则化因子 $C=[10, 1, 10^4]$, 支持向量 $L=[200, 200, 300]$ 。ELM、RKELM、RDK-ELM 算法的运行结果是取 100 次运行结果的平均数,KELM 和 ML-KELM 算法只运行 1 次。结果见表 14。

表 14 各算法进行故障诊断的性能结果

参数	ELM	KELM	RKELM	ML-KELM	RDK-ELM
Tr.time/s	0.0647	0.5014	0.0252	4.3867	0.1320
Ts.rate/%	95.57	97.22	96.38	97.92	97.12

从表中可见,ML-KELM 算法精度最高,RKELM 算法速度最快,RDK-ELM 和 KELM 算法精度基本持平,但 RDK-ELM 比 KELM 算法的训练时间缩短了 1/4。可见,利用深度学习结构结合核方法,可以提高算法的泛化性能与分类能力,在样本数量大、类别多的情况下精度更高;应用简约方法则明显改善了单纯核方法的时间性能。RDK-ELM 算法将 2 种优点结合起来,得到优越的综合性能。

4.3 算法参数分析

RDK-ELM 算法可以调整的参数有网络深度(隐含层层数)、每层核参数和正则化因子、每层支持向量数目。

(1)网络深度。虽然深度学习网络结构是通过逐层对输入数据提取特征来获得更高精度的,但本文对 UCI 分类数据集及发动机故障数据的试验表明,并不是层数越多精度越高,精度在隐含层层数为 2 时的精度最高,层数再高精度反而降低。此外,网络深度直接

影响训练时长。因此将隐含层层数设置为 2 层具有最稳定的精度和可接受的速度。

(2)核参数和正则化因子。核参数 σ 和正则化因子 C 是在 $10^{-7} \sim 10^7$ 范围内按 10 的倍数选取的,这 2 个参数对算法的精度有非常大的影响^[1]。目前对机器学习算法的参数中,这 2 个参数的选取通常根据经验选择。通过试验调参,可以得到令人满意的精度。

(3)支持向量的数量。RDK-ELM 算法选取的支持向量越多,精度越高,训练时间也越长。当支持向量为全部样本时,RDK-ELM 与 ML-KELM 算法等效^[13]。而在试验中,支持向量达到一定数目时,精度的提高就不明显了,而同时速度则明显降低。为了获得精度和速度的综合优势,支持向量的数目需要选取 1 个合适数值。在目前基于支持向量的算法的研究中,对支持向量数目的选取均没有理论上的方法,一般根据经验和试验选取。本文的算法是先取输入向量样本数目的 1/10,然后以此为基准增加或减少支持向量的数目,以达到训练速度和训练精度的综合最优为止。

5 总结

(1)将 RKELM 算法与深度学习网络结构各自的优点结合,提出 RDK-ELM 算法。

(2)对 UCI 分类数据集和发动机故障仿真数据进行试验,验证了 RDK-ELM 算法在精度与速度上的综合优势。

(3)RDK-ELM 算法可调参数较多,需要针对具体情形进行参数调整,通过试验验证以获得最佳性能。

参考文献:

[1] 姜洁. 航空发动机控制系统传感器故障诊断与容错控制研究[D]. 南京:南京航空航天大学, 2015.
JIANG Jie. Research on aero-engine control system sensor fault diagnosis and fault tolerant control [D]. Nanjing: Nanjing University of Aeronautics & Astronautics, 2015. (in Chinese)

[2] Spang H A, Corley R C. Failure detection and correction for turbofan engines [M]. General Electric, Corporate Research and Development, 1977: 15-25.

[3] Sanjay G, Klaus S, Wolfgang H, et al. Sensor and actuator needs for more intelligent gas turbine engines[R]. NASA-TM-2010-216746.

[4] Kumar A, Viassolo D E. Model-based fault tolerant control [R]. NASA-CR-2008-215273.

[5] 郝红勋, 白杰, 郝英. 基于 BP 神经网络方法的民用航空发动机故障诊断[J]. 中国民航飞行学院学报, 2006, 17(5): 36-38.

- HAO Hongxun,BAI Jie,HAO Ying. Civil aviation engine fault diagnosis based on BP neural network method [J]. Journal of Civil Aviation Flight University of China,2006,17(5):36-38.(in Chinese)
- [6] Guo T H,Nurre J. Sensor failure detection and recovery by neural networks[C]//Ijcn-91-Seattle International Joint Conference on Neural Networks. IEEE,1991:221-226.
- [7] 黄向华. 航空发动机智能容错数字控制[J]. 应用基础与工程科学学报,2005,13(4):37-43.
- HUANH Xianghua. Aeroengine intelligent fault tolerant digital control [J]. Journal of Basic Science and Engineering,2005,13(4):37-43.(in Chinese)
- [8] 蔡开龙,孙云帆,姚武文. 航空发动机传感器故障诊断与自适应重构控制[J]. 电光与控制,2009,16(6):1118-1126.
- CAI Kailong,SUN Yunfan,YAO Wuwen. Fault diagnosis and adaptive reconfiguration control for sensors in aeroengine [J]. Electronics Optics & Control,2009,16(6):1118-1126.(in Chinese)
- [9] Huang G B,Zhu Q Y,Siew C K. Extreme learning machine:a new learning scheme of feed-forward neural networks [C]//In 2004 IEEE international joint Conference on Neural Networks. Proceedings,2004:985-990.
- [10] Huang G,Zhou H,Ding X,et al. Extreme learning machine for regression and muticlass classification [J]. IEEE Trans Syst,Man, Cybern,Part B:Cybern. 2011,42(2):513-529.
- [11] Wong C M,Vong C M,Wong P K,et al. Kernel-based multilayer extreme learning machines for representation learning [J]. IEEE Transactions on Neural Networks & Learning Systems,2016,99:1-6.
- [12] 逢珊,杨欣毅,张勇,等. 应用深度核极限学习机的航空发动机部件故障诊断[J]. 推进技术,2017,38(11):2613-2621.
- PANG Shan,YANG Xinyi,ZHANG Yong,et al. Application of deep kernel extreme learning machine in aeroengine components fault diagnosis [J]. Journal of Propulsion Technology, 2017,38(11):2613-2621.(in Chinese)
- [13] Deng W Y,Ong Y S,Zheng Q H. A fast reduced kernel extreme learning machine[J]. Neural Networks,2016,76(2):29-38.
- [14] 魏洁. 深度极限学习机的研究与应用 [D]. 太原: 太原理工大学, 2016.
- WEI Jie. Research and application of deep extreme learning machine [D]. Taiyuan:Taiyuan University of Technology,2016.(in Chinese)
- [15] Tamilselvan P,Wang P. Failure diagnosis using deep belief learning based health state classification[J]. Reliability Engineering & System Safety,2013,115(7):124-135.
- [16] 李本威,林学森,杨欣毅. 深度置信网络在发动机部件性能退化故障诊断中的应用研究[J]. 推进技术,2016,37(11):2173-2180.
- LI Benwei,LIN Xuesen,YANG Xinyi. Research on the application of deep belief network in aeroengine component performance deterioration fault diagnosis [J]. Journal of Propulsion Technology, 2016,37(11):2173-2180.(in Chinese)

(编辑:刘 亮)