

控制与决策

Control and Decision

并行集成具有高可解释的TSK模糊分类器

张雄涛, 蒋云良, 胡文军, 王士同

引用本文:

张雄涛, 蒋云良, 胡文军, 等. 并行集成具有高可解释的TSK模糊分类器[J]. *控制与决策*, 2020, 35(10): 2535–2542.

在线阅读 View online: <https://doi.org/10.13195/j.kzyjc.2018.1794>

您可能感兴趣的其他文章

Articles you may be interested in

具有强解释性的贝叶斯MA型模糊系统

High interpretative Bayesian Mamdani–Assilan type fuzzy system

控制与决策. 2018, 33(1): 16–26 <https://doi.org/10.13195/j.kzyjc.2016.1418>

模糊聚类光滑支持向量机

Fuzzy clustering smooth support vector machine

控制与决策. 2017, 32(3): 547–551 <https://doi.org/10.13195/j.kzyjc.2015.1526>

贝叶斯L2型TSK模糊系统

Bayesian L2–norm–Takagi–Sugeno–Kang fuzzy system

控制与决策. 2017, 32(10): 1871–1878 <https://doi.org/10.13195/j.kzyjc.2016.0853>

适合大规模数据集且基于LLM的0阶TSK模糊分类器

Zero–order TSK fuzzy classifier based on LLM for large–scale data sets

控制与决策. 2017, 32(1): 21–30 <https://doi.org/10.13195/j.kzyjc.2015.1579>

基于模糊子空间聚类的0阶岭回归TSK模糊系统

Fuzzy subspace clustering based 0–order ridge regression TSK fuzzy system

控制与决策. 2016, 31(5): 882–888 <https://doi.org/10.13195/j.kzyjc.2015.0182>

并行集成具有高可解释的TSK模糊分类器

张雄涛^{1,2†}, 蒋云良², 胡文军², 王士同¹

(1. 江南大学 数字媒体学院, 江苏 无锡 214122; 2. 湖州师范学院 信息工程学院, 浙江 湖州 313000)

摘要: 针对分层Takagi-Sugeno-Kang (TSK) 模糊分类器可解释性差, 以及当增加或删除一个TSK 模糊子分类器时 Boosting 模糊分类器需要重新训练所有 TSK 模糊子分类器等问题, 提出一种并行集成具有高可解释的 TSK 模糊分类器 EP-Q-TSK. 该集成模糊分类器每个 TSK 模糊子分类器可以使用最小学习机 (LLM) 被并行地快速构建. 作为一种新的集成学习方式, 该分类器利用每个 TSK 模糊子分类器的增量输出来扩展原始验证数据空间, 然后采用经典的模糊聚类算法 FCM 获取一系列代表性中心点, 最后利用 KNN 对测试数据进行分类. 在标准 UCI 数据集上, 分别从分类性能和可解释性两方面验证了 EP-Q-TSK 的有效性.

关键词: 集成 TSK 模糊分类器; 并行学习; 最小学习机; 代表性中心点

中图分类号: TP181 文献标志码: A

DOI: 10.13195/j.kzyjc.2018.1794

引用格式: 张雄涛, 蒋云良, 胡文军, 等. 并行集成具有高可解释的 TSK 模糊分类器 [J]. 控制与决策, 2020, 35 (10): 2535-2542.

Ensemble TSK fuzzy classifiers with parallel learning and high interpretability

ZHANG Xiong-tao^{1,2†}, JIANG Yun-liang², HU Wen-jun², WANG Shi-tong¹

(1. School of Digital Media, Jiangnan University, Wuxi 214122, China; 2. School of Information Engineer, Huzhou University, Huzhou 313000, China)

Abstract: Traditional ensemble Takagi-Sugeno-Kang (TSK) fuzzy subclassifiers face such challenges, hierarchical learning have no interpretability, because of the presence of intermediate variables, and when a new TSK fuzzy subclassifier is added to or removed from the structure of the current fuzzy classifier, boosting learning must retrain each TSK fuzzy subclassifier by appropriately assigning new weights. Therefore, An ensemble framework EP-Q-TSK of TSK fuzzy subclassifiers with parallel learning way is proposed. The proposed framework has the following distinctive characteristics: 1) Each TSK fuzzy subclassifier can be built quickly with least learning machine (LLM) in parallel; 2) As a novel ensemble learning, the proposed framework augments the original validation data space with the outputs of each TSK subclassifier in an incremental and inexpensive way, and then speed up the final classification on the validation data by using the FCM and the KNN method; 3) Enhanced classification performance by FCM & KNN is experimentally revealed, and the experimental results on benchmark datasets indicate the effectiveness of EP-Q-TSK and its parallel learning method in the sense of both enhanced classification performance and interpretability.

Keywords: ensembling Takagi-Sugeno-Kang (TSK) fuzzy subclassifiers; parallel learning; least learning machine; representative centroids

0 引言

Takagi-Sugeno-Kang (TSK) 模糊分类器^[1-3] 由于其良好的分类性能和高可解释性, 在金融预测、模式识别^[4-5]、智能数据处理和自适应模糊控制^[3,6] 等数据驱动应用中取得了广泛的成功. 然而, 到目前为止, 如何在实际应用中保持较高分类精度的同时具有高可解释性仍然是一个巨大的挑战^[3]. 为此, 本文试图解

决以下问题: 1) 模糊规则数目不当和模糊划分不当, 可能严重降低模糊规则 if 部分的可解释性, 或严重降低 TSK 模糊分类器的分类性能; 2) 在传统的分层集成 TSK 模糊分类器^[7] 中, 由于中间变量的存在, 模糊分类器的中间子分类器无可解释性; 3) 在 Boosting 集成模糊分类器^[8] 中, 每次增加或删除一个子分类器时, 都要对所有子分类器重新进行训练, 由此导致较

收稿日期: 2018-12-29; 修回日期: 2019-05-07.

基金项目: 国家自然科学基金项目 (61572236, 61300151, 61772198, 61771193).

责任编委: 赵珺.

†通讯作者. E-mail: 1047897965@qq.com.

高的复杂度.

现有的分层TSK模糊分类器^[9-11]如图1所示,既具有良好的分类性能,又能够很好地克服规则组合爆炸问题^[7,9]. 由于中间变量的存在,中间层和输出层中的TSK模糊子分类器很难解释每条模糊规则的语义,随着层次TSK模糊分类器中层数的增加,这一问题变得尤为突出. Boosting模糊分类器^[8-12]如图2所示. 由于各TSK模糊子分类器之间的相关性,当增加或删除一个TSK模糊子分类器时,Boosting模糊分类器必须放弃每个TSK模糊子分类器已学习到的权重,然后通过适当分配新的权重对其进行重新训练,从而造成繁重的计算负担.

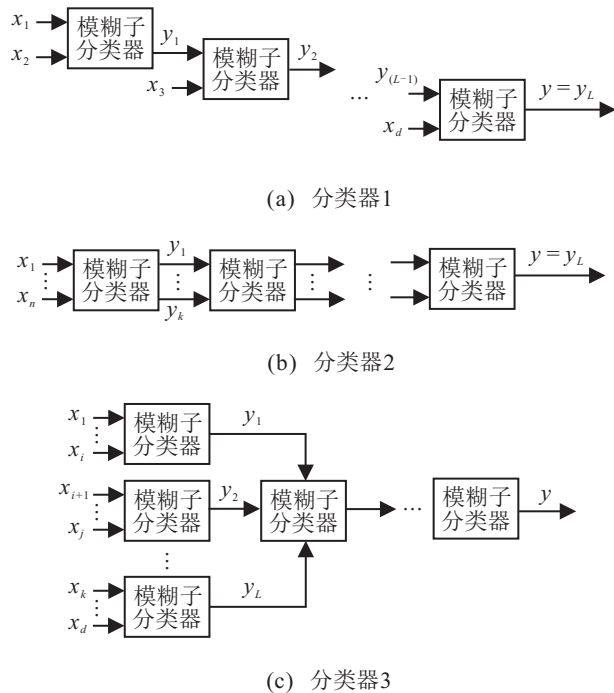


图1 分层TSK模糊分类器

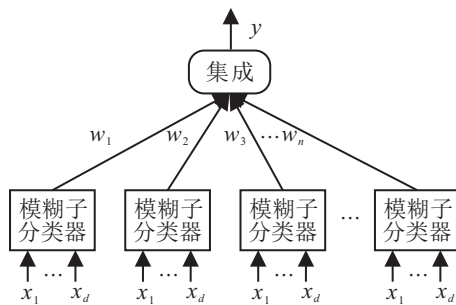


图2 Boosting模糊分类器

为克服上述问题,本文提出一种新的具有并行结构的集成TSK模糊分类器EP-Q-TSK. 它以并行学习方式组织其所有零阶TSK模糊子分类器,没有中间变量的存在. 在EP-Q-TSK中,每个零阶TSK子分类器的输出被扩充到原始(验证)输入空间,以便以并行方式打开原始(验证)输入空间中存在的流形结构,因此根据堆栈泛化原理^[13-14]可以保证提高分类精度. 由于EP-Q-TSK是以并行学习方式构建的,容易

增加或者删除TSK模糊子分类器. 在EP-Q-TSK中采用新的预测机制FCM & KNN,能够实现快速预测. 采用经典的FCM^[15]聚类算法,以便从增强验证集上获得一系列具有标签的代表点. 然后采用KNN^[16]方法求出训练样本或隐含样本的 k 个最近代表点,并以它们及其类标代替整个验证数据,快速确定训练样本或隐含样本的标签.

1 EP-Q-TSK分类器

1.1 EP-Q-TSK框架

图3为EP-Q-TSK的整体框架. 图3中: D_{tr} 为总训练数据集; D_1, D_2, \dots, D_L 为 D_{tr} 子集;TSK₁, TSK₂, ..., TSK_L为 L 个零阶TSK子分类器, $f_{TSK_1}(x), f_{TSK_2}(x), \dots, f_{TSK_L}(x)$ 为输出函数. EP-Q-TSK包括增强验证数据生成和代表性中心点生成及预测两个阶段. 本文假设数据被划分为训练数据集、验证数据集和测试数据集,下面详述EP-Q-TSK的工作流程.

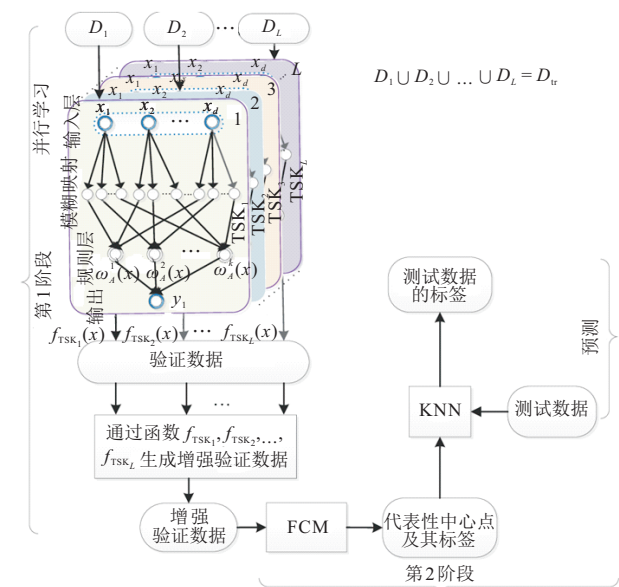


图3 EP-Q-TSK框架

EP-Q-TSK首先从生成训练子集 D_1, D_2, \dots, D_L 开始, D_i 从训练数据集 D_{tr} 中随机抽取,并且 $D_1 \cup D_2 \cup \dots \cup D_L = D_{tr}$. 这些子集作为 L 个零阶TSK模糊子分类器的输入,为了保证零阶TSK模糊子分类器之间的多样性^[12,17],为每个分类器分配不同的模糊规则数. 将每个输入特征等间隔划分为5个固定的模糊集,由此相应地产生5种高斯隶属度函数,固定中心分别为0、0.25、0.5、0.75、1. 虽然每个高斯隶属函数的核宽度可能不同,但不会显著影响模糊子分类器的语义解释,分别是:非常小、小、中、大、非常大^[18-19]. 基于之前的工作^[20-21],本文将随机正数作为每个隶属函数的宽度,相应地使用LLM^[22]独立学习每个子分类器的后件参数,算法1给出了每个子分类

器的学习算法.

算法1 零阶TSK模糊分类器学习算法.

输入: 第 l 个训练数据子集 $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_{N_l}]^T$, 其对应的标签集 $\mathbf{y} = [y_1, y_2, \dots, y_{N_l}]^T \in R^N$, $\mathbf{x}_i \in R^d, y_i \in R, i = 1, 2, \dots, N_l$, 模糊规则数 K_l , 正则化参数 C ;

输出: 子分类器的模糊规则及其输出函数.

step 1: 构造5种高斯型隶属度函数. 中心点集合为 $\{0, 0.25, 0.5, 0.75, 1\}$, 在每一维上从中心点集合中随机指定一个值并构造规则组合矩阵 $\Theta_l = [v_{ik}^l]_{K_l \times d}$.

step 2: 通过给每个元素分配一个随机正数来构造核宽度矩阵 $\Phi_l = [\delta_{ik}^l]_{K_l \times d}$.

step 3: 根据规则组合矩阵、核宽度矩阵和上述5种高斯隶属度函数, 利用式(1)计算各模糊规则的模糊隶属度 $\omega_{A_i^k}^l(\mathbf{x}) = \prod_{i=1}^d \omega_{A_i^k}^l(\mathbf{x}_i)$, 有

$$\omega_{A_i^k}^l(\mathbf{x}_i) = \exp(-(\mathbf{x}_i - v_{ik}^l)^2 / \delta_{ik}^l). \quad (1)$$

其中: $i = 1, 2, \dots, d, k = 1, 2, \dots, K_l$. 对于所有样本, 将模糊隶属度正则化并得到矩阵 \mathbf{X}_g , 即

$$\mathbf{X}_g = \begin{bmatrix} \tilde{\omega}_1^1 & \dots & \tilde{\omega}_1^{K_l} \\ \vdots & \ddots & \vdots \\ \tilde{\omega}_{N_l}^1 & \dots & \tilde{\omega}_{N_l}^{K_l} \end{bmatrix}_{N_l \times K_l}. \quad (2)$$

从而得到第 l 个零阶TSK模糊分类器的所有前件参数 \mathbf{a}_g .

step 4: 通过LLM计算后件参数

$$\mathbf{a}_g = ((1/C)/\mathbf{I} + \mathbf{X}_g^T \mathbf{X}_g)^{-1} \mathbf{X}_g^T \mathbf{y}. \quad (3)$$

其中: \mathbf{I} 为 $K \times K$ 单位矩阵, C 为给定的正则化参数.

step 5: 由式(4)得到第 l 个零阶TSK模糊分类器的输出函数 $f_{\text{TSK}_l}(\mathbf{x})$, 有

$$\mathbf{y}_l = \mathbf{X}_g \mathbf{a}_g. \quad (4)$$

对每个子分类器进行并行训练后, 将其输出函数作用在验证数据集上, 从而生成 L 个增强特征. 通过将验证集上的原始特征和增强特征组合并生成具有 $d + L$ 维的增强验证数据集, 如此可以在接下来的分类活动中将原始输入空间中的流形分开, 从而以堆栈方式实现更好的线性可分. 换句话说, 根据堆栈泛化原理^[13], 与原始验证数据集相比, 增强验证数据集更有助于提高分类性能.

在生成增强验证数据集后, EP-Q-TSK 进入到第2阶段. 为了实现对训练或隐含数据的快速KNN预测, 希望有少量且有代表性的数据样本作为代表性中心点供KNN使用. 然而, 增强验证数据集有时数据量较大, 因此使用经典的FCM聚类算法找出增强验证数据集各类的代表性中心点. 在运行FCM后, 所有去

掉增强特征的代表性中心点及其标签都被作为数据字典, 用于KNN对训练或隐含数据的分类. 整个第2阶段称为FCM & KNN预测机制, 这种机制比直接在原始验证数据集上执行KNN能够实现更快的预测.

EP-Q-TSK具有以下特点:

1) 由于每个子分类器中的每条模糊规则都是高度可解释的, 根据EP-Q-TSK的集成框架, 各子分类器以及EP-Q-TSK都是可解释的. 由于三重随机性(5个固定高斯隶属函数的随机分配及其对每个特征的精确语义解释, 每个子分类器模糊规则数从不同区间随机选择, 以及训练样本的随机 L 份划分), EP-Q-TSK有着潜在的适应环境的潜力.

2) EP-Q-TSK可以以快速的方式实现. 尽管本文使用了具有高可解释性的零阶TSK模糊分类器, 并提出了算法1, 但理论上, 任何模糊分类算法都可以被并行地应用于每个子分类器. 在所提出的框架中, FCM和KNN都可以很容易地从现有的工具箱中获取. 由于FCM & KNN是以单独的方式执行, 增加或删除子分类器不需要对第2阶段的编程代码进行任何更改, 只需更改作为输入的增强验证数据集即可.

3) EP-Q-TSK框架为这样一些实际应用场景提供了解决方案: 需要数据隐私保护或在分布式系统中运行, 或者当足够的验证数据是昂贵的甚至很难获取. 例如, 由于患者的隐私保护, 直接从医院或相关机构获取医学数据作为训练数据集是不方便甚至不切实际的. 然而, 将TSK模糊子分类器的训练算法传递给每个医院或相关机构后, 在没有隐私泄露的情况下, 每个医院或相关机构可以通过在各自的数据集上运行所传递的训练算法来训练TSK模糊子分类器, 然后传递所得到的TSK模糊子分类器. 此外, 当带有标签的真实医学数据作为验证数据集时, 由于其非常稀缺甚至昂贵, 可以利用医院或相关机构建立的TSK模糊子分类器, 以堆栈的方式获得增强验证数据集, 从而确保获得满意的分类性能.

1.2 EP-Q-TSK学习算法及复杂度分析

训练完各TSK模糊子分类器, 将FCM作用在增强验证数据集上, 在得到数据字典后, 对于任何训练或隐含样本, 利用KNN方法对它们进行训练和预测. 基于上述算法, 将EP-Q-TSK的整个学习算法描述为算法2, 它包含两个过程: 训练过程构建增强验证数据集并利用FCM生成一系列代表性中心点; 预测过程使用KNN方法对所有测试样本的标签进行预测.

算法2 EP-Q-TSK学习算法.

输入: 训练数据集 $\mathbf{D}_{\text{tr}} = [\mathbf{X}_{\text{tr}} \ \mathbf{Y}_{\text{tr}}]$, 验证数据集 $\mathbf{D}_v = [\mathbf{X}_v \ \mathbf{Y}_v]$, \mathbf{X}_{tr} 和 \mathbf{X}_v 分别表示训练数据和验

证数据,对应的标签集分别为 \mathbf{Y}_{tr} 和 \mathbf{Y}_v ,子分类器数 L, K_1, K_2, \dots, K_L 表示每个子分类器的模糊规则数;
输出:EP-Q-TSK结构及代表性中心点。

初始化:为每个子分类器从 \mathbf{D}_{tr} 中随机抽取训练数据子集 $\mathbf{D}_1, \mathbf{D}_2, \dots, \mathbf{D}_L$, 且 $\mathbf{D}_1 \cup \mathbf{D}_2 \cup \dots \cup \mathbf{D}_L = \mathbf{D}_{tr}$ 。

step 1: 并行训练 L 个零阶 TSK 子分类器。

step 1.1: 为每个子分类器随机分配模糊规则数;

step 1.2: 以并行的方式通过调用算法1生成 L 个子分类器;

step 1.3: 输出 L 个子分类器的输出函数

$$f_{\text{TSK}_1}(\mathbf{x}), f_{\text{TSK}_2}(\mathbf{x}), \dots, f_{\text{TSK}_L}(\mathbf{x}).$$

step 2: 生成增强验证数据集. 对于验证数据集的每个样本,计算对应的输出函数 $f_{\text{TSK}_1}(\mathbf{x}), f_{\text{TSK}_2}(\mathbf{x}), \dots, f_{\text{TSK}_L}(\mathbf{x})$, 并将其作为原样本的增强特征,形成增强验证数据集 $\mathbf{D}_v^{\text{new}} = [\mathbf{X}_v \ \bar{\mathbf{X}}_v \ \mathbf{Y}_v]$, 其中 $\bar{\mathbf{X}}_v$ 为验证数据的增强特征集。

step 3: 在增强验证数据集 $\mathbf{D}_v^{\text{new}}$ 上对每一类分别调用FCM算法生成一系列代表性的中心点及其对应的标签,并去掉代表性中心点的增强特征。

预测过程如下。

step 1: 对于任何测试样本,利用KNN找到距离其最近的 k 个代表性中心点,基于投票策略确定其类标。

step 2: 输出测试样本的预测标签。

下面分析EP-Q-TSK算法的时间复杂度. 构造规则组合矩阵 Θ_l 和核宽度矩阵 Φ_l 的时间复杂度分别为 $O(5dK_l)$ 和 $O(5N_l d^2 K_l)$, 通过LLM生成后件参数矩阵 \mathbf{a}_g 的时间复杂度为 $O(K_l^3)$. 其中: K_l 为第 l 个子分类器的模糊规则数, N_l 为第 l 个子分类器的训练样本数, d 为维度. 第2阶段,FCM的时间复杂度为

$$O(N_v c^2 (d + L) \text{Iter}_{\max}).$$

其中: $d + L$ 组合了增强验证数据集中原始特征和增强特征, N_v 为增强验证数据集样本数, c 为代表性中心点数, Iter_{\max} 为最大迭代次数. KNN对于整个训练数据集的时间复杂度为 $O(NQcd)$, 其中 N 是训练样本的总数. 因此EP-Q-TSK的总时间复杂度为 $O(5N_{\max} d^2 K_{\max} + K_{\max}^3 + N_v c^2 (d + L) \text{Iter}_{\max} + NQcd)$. 其中: K_{\max} 为最大模糊规则数, N_{\max} 为包含最大模糊规则数的子分类器的样本数. 由于 N_{\max} 和 K_{\max} 针对于某一个子分类器, N_v 是珍贵的增强验证数据集的样本数, 它们通常都小于 N , 算法的时间复杂度可以粗略地看成与训练样本的总数 N 线性相关。

1.3 理论分析

引理1^[23] 不一致概率 $P(c^1 \neq c^2)$ 是两个子分类器分类误差的严格上界。

当FCM & KNN作用于增强验证数据集 $\tilde{\mathbf{X}}_v = [\mathbf{X}_v, \bar{\mathbf{X}}_v]$ 上的任何训练或隐含样本时,FCM & KNN可以等价地看作是具有两个分类器 h^1, h^2 的多任务分类方法,即在增强验证集上的FCM可以分解为在原特征集上和增强特征集上的分类,它们均采用相同的KNN方法. 由引理1可知,通过降低不一致概率 $P(c^1 \neq c^2)$, 可以提高两个任务场景中每个子分类器的分类性能,因此本文采用两种方法:1) 采用类似文献[23]式(3)的方法,在目标函数中增加一个关于不一致概率的显式惩罚项;2) 考虑隐式的方法,使分类器 c^1 和 c^2 尽可能一致. FCM & KNN属于第2种方法. 本质上,本文EP-Q-TSK通过以堆栈方式将每个子分类器的输出对原始验证数据集进行扩充以生成增强验证数据集,因此原始验证数据集中的流形结构通过增强特征相继被打开. 根据堆栈泛化原理^[13], 对于任何作用在该增强验证数据集上的分类器,其分类性能肯定会高于在原始验证数据集上的结果. 这意味着当要对任何训练或隐含样本进行预测时,与原始验证数据相比,增强验证数据有更强大的分类潜能,即作用在增强验证数据集而不是原始验证数据集,具有同样KNN的两个分类器 c^1 和 c^2 , 能够有强大的能力,以达到一致的分类结果. 基于以上分析,同时根据引理1,所提出的EP-Q-TSK框架在理论层面上其高效性得以证明。

2 实验分析

2.1 数据集及实验设置

实验运行的硬件环境为: i7-8700K 3.70 GHz CPU and 64 GB RAM, Microsoft Windows 10, Matlab 2016a. 实验中所有11个数据集均来自UCI^[24] 和 KEEL^[25], 详细信息如表1所示。

表1 数据集

数据集	类别数	特征数	样本数
wdbc (WDB)	2	14	569
spectheart (SPE)	2	44	267
air (AIR)	3	64	359
satimage (SAT)	6	36	6435
chess (CHE)	2	36	3 196
spambase (SPA)	2	57	4 597
waveform3 (WAV)	3	21	5 000
penbased (PENB)	10	16	10 992
pendigits (PEND)	10	16	7 494
landsat (LAN)	6	36	2 000
adult (ADU)	2	14	48 841

由表 1 可见, EP-Q-TSK 在这些数据集上运行时, 实验设置应该考虑所有子分类器之间的多样性, 以及与数据集大小、维度和类别数之间的关系, 为此, 如表 2 所示, 为每个子分类器设置不同的模糊规则区间. 根据数据集的大小, 为 6 个数据集安排 3 个子分类器, 其余数据集安排 5 个子分类器. 为了更具多样性, 采用两种方案设置模糊规则数: 对数据集 *wdbc* 和 *spectfheart*, 每个子分类器的模糊规则数从 2~15 进行网格搜索并获得 3 个分类器最佳的规则数组合; 对于其他数据集, 给每个子分类器分配一个不同的模糊规则数区间, 然后从预先指定的规则区间内随机抽取规则数训练每个子分类器 20 次. 对于采用的 FCM 聚类中心点数, 表 2 给出了搜索范围, 即每类代表性中心点的范围. KNN 中的最近邻 k 搜索区间范围设置为 {3, 5, 7}.

表 2 EP-Q-TSK 参数设置

数据集	分类器	规则数	代表性中心点
WDB	3	2~15	3~4
SPE			2~3
AIR	3	1. 5~80	2~3
WAV		2. 8~110	17~20
PENB		3. 10~130	10~13
PEND		80~100	
LAN	5	1. 5~80	3~4
SAT		2. 8~100	20~23
CHE		3. 10~110	17~20
SPA		4. 13~130	25~27
ADU		5. 15~150	40~45

本实验中, 每个数据集被随机抽取 70% 作为训练数据集, 20% 作为验证数据集, 10% 作为测试数据集. 组织 4 组分类器进行比较: 第 1 组为集成分类器, 它们是 GFS-LogitBoost-C 和 Ensemble-C; 第 2 组为基于深度学习的分类器 (即 DBN^[26]); 第 3 组包括经典的零阶 (zero-order-TSK-FC) 和一阶 TSK 模糊分类器 (first-order-TSK-FC); 第 4 组为常规分类器 (即 KNN^[16]). 由于 GFS-LogitBoost-C 和 Ensemble-C 来自于 KEEL 工具箱并用 Java 实现, 其他分类器均在 Matlab 中实现, 因此, 实验中只给出平均训练/测试分类精度及标准方差、平均模糊规则数.

2.2 实验结果与分析

表 3 为 EP-Q-TSK 和 7 个比较分类器在 10 个数据集上的实验结果, 实验比较了平均训练和测试精度及其标准方差以及平均模糊规则数, 其中最好的结果用粗体表示. 稍后给出关于数据集 *wdbc* 更详细的实验结果. 注意到, KEEL 中比较分类器的结果没有给出模糊规则数, 因此不为每个数据集提供相应的值, 在表 3 中将它们表示为 “-”.

1) 从分类精度角度看, 与其他分类器相比, EP-Q-TSK 在平均训练和测试分类精度方面表现出满意的结果. 具体而言, EP-Q-TSK 在 10 个数据集集中的 7 个数据集具有最高的平均训练精度和 / 或平均测试精度. 此外, EP-Q-TSK 在多数情况下比 Ensemble-C 和 GFS-LogitBoost-C、零阶 TSK 模糊分类器和 KNN 具有更好的性能. 一阶 TSK 模糊分类器胜出 3 次, DBN 胜出 2 次. 其原因在于, 根据堆栈泛化原理^[13], EP-Q-TSK 中 FCM 得到的增强特征能够帮助打开原始验证数据中的流形结构, 从而更好地实现对数据的分类.

2) 在采用的数据集中 (如 *spectfheart*、*air*、*chess* 和 *spambase*), EP-Q-TSK 的性能比其他分类器略差. 然而, 当 DBN、KNN、Ensemble-C、GFS-LogitBoost-C 不具有可解释性, 一阶 TSK 模糊分类器中的模糊规则可解释性低时, EP-Q-TSK 及其每个子分类器都具有高可解释性. 此外, EP-Q-TSK 中的每个子分类器比零阶 TSK 模糊分类器在大多数情况下平均需要较少的模糊规则数, 这是因为每个子分类器的输出仅作为鉴别信息来形成一个增强特征, 从而依次打开验证数据集中的流形结构, 进而提高 EP-Q-TSK 的性能. 通常情况下, 较少的规则数意味着更强的可解释性.

3) 通过仔细观察发现, EP-Q-TSK 在多分类数据集 (如 *pendigits*、*penbased*、*landsat*、*satimage*) 上优于其他分类器. 一般而言, 由于多类数据集的流形结构比两分类数据集更为复杂, 这一发现实际上表明在 EP-Q-TSK 中采用多个子分类器生成相应增强特征的帮助下, EP-Q-TSK 具有打开更复杂数据集的流形结构的能力.

为了更加详细地阐述 EP-Q-TSK, 以数据集 *wdbc* 为例, 从分类性能和模糊规则的可解释性两个方面对其进行详细展示. *wdbc* 数据集由 569 个样本组成, 包括两个类别, 该数据集的特征由乳腺肿块细针抽吸物 (FNA) 的数字化图像计算生成, 描述了图像中细胞核的特征. 任务是要确定发现的肿瘤是良性还是恶性 (分别表示为 +1 和 -1). EP-Q-TSK 在 *wdbc* 上采用 3 个子分类器, 每个子分类器中模糊规则数目的网格搜索范围均为 2~15. 表 4 列出了所有分类器在数据集 *wdbc* 上的最大训练和测试精度、平均规则数、平均训练和测试精度及其方差、平均训练和测试时间的实验结果. 由于 GFS-LogitBoost-C 和 Ensemble-C 由 Java 实现, 不给出训练和测试时间, 用 “-” 表示. 如表 4 所示, EP-Q-TSK 在最大和平均训练精度、最大和平均测试精度方面不及其他分类器, 但是平均每个子分类器需要的模糊规则数最少.

表3 各数据集在比较分类器上的性能对比结果

	Zero-order-TSK-FC			First-order-TSK-FC			DBN			KNN		
	training	testing	rules	training	testing	rules	training	testing	rules	training	testing	rules
WAV	0.673 9 (0.052 1)	0.671 3 (0.054 0)	76.9	0.763 7 (0.015 1)	0.737 7 (0.020 6)	43.5	0.815 4 (0.054 0)	0.808 2 (0.055 9)	—	0.834 3 (0.005 3)	0.825 6 (0.015 7)	—
SPE	0.796 0 (0.017 5)	0.791 0 (0.031 9)	12.5	0.826 3 (0.015 4)	0.746 1 (0.054 5)	7.6	0.817 5 (0.021 0)	0.809 0 (0.047 6)	—	0.800 0 (0.010 0)	0.776 1 (0.029 9)	—
AIR	0.416 7 (0.045 9)	0.396 1 (0.059 1)	70.2	0.796 3 (0.023 9)	0.742 2 (0.036 3)	41.4	0.861 0 (0.041 2)	0.777 8 (0.057 9)	—	0.776 7 (0.054 3)	0.738 7 (0.056 3)	—
LAN	0.568 7 (0.046 1)	0.564 3 (0.040 4)	83.7	0.738 8 (0.011 6)	0.725 5 (0.021 3)	51.2	0.750 0 (0.059 0)	0.732 7 (0.052 5)	—	0.742 0 (0.040 9)	0.740 0 (0.039 4)	—
PEND	0.447 2 (0.042 4)	0.401 0 (0.041 5)	98.9	0.686 7 (0.013 8)	0.668 4 (0.019 7)	69.9	0.742 8 (0.055 6)	0.731 0 (0.055 7)	—	0.888 5 (0.004 5)	0.887 8 (0.008 5)	—
CHE	0.827 2 (0.050 8)	0.824 2 (0.049 0)	75.6	0.993 1 (0.005 4)	0.970 1 (0.009 1)	59.7	0.896 8 (0.021 6)	0.889 6 (0.029 3)	—	0.513 9 (0.018 2)	0.513 5 (0.025 1)	—
SAT	0.608 4 (0.034 4)	0.570 9 (0.042 7)	90.2	0.689 6 (0.008 8)	0.673 6 (0.016 2)	63.9	0.675 1 (0.051 2)	0.666 1 (0.059 9)	—	0.806 6 (0.038 1)	0.806 1 (0.046 4)	—
PENB	0.457 3 (0.042 4)	0.447 9 (0.058 0)	67.4	0.675 8 (0.038 3)	0.665 3 (0.036 4)	61.9	0.676 5 (0.059 5)	0.675 0 (0.056 8)	—	0.897 5 (0.004 2)	0.891 7 (0.004 9)	—
SPA	0.820 5 (0.035 8)	0.815 7 (0.038 1)	107.5	0.920 8 (0.005 2)	0.906 9 (0.009 4)	71.4	0.838 5 (0.026 5)	0.837 7 (0.027 5)	—	0.610 4 (0.000 9)	0.602 2 (0.006 5)	—
ADU	0.826 0 (0.053 8)	0.801 4 (0.048 0)	135.5	0.848 6 (0.028 5)	0.785 9 (0.048 4)	91.3	0.825 2 (0.056 5)	0.821 3 (0.054 5)	—	0.823 0 (0.056 9)	0.803 9 (0.010 1)	—
	Ensemble-C			CFS-LogitBoost-C			EP-Q-TSK					
	training	testing	rules	training	testing	rules	training	testing	rules			
WAV	0.789 7 (0.033 7)	0.785 8 (0.022 5)	—	0.837 3 (0.002 5)	0.830 0 (0.001 7)	—	0.854 0 (0.009 3)	0.840 8 (0.016 1)	52.4			
SPE	0.758 3 (0.024 1)	0.753 2 (0.020 3)	—	0.794 8 (0.001 9)	0.793 1 (0.002 4)	—	0.851 9 (0.054 6)	0.793 4 (0.056 4)	5.8			
AIR	0.701 9 (0.046 3)	0.694 8 (0.001 9)	—	0.567 0 (0.033 5)	0.552 6 (0.019 4)	—	0.567 9 (0.056 2)	0.546 1 (0.041 6)	59.7			
LAN	0.614 5 (0.032 8)	0.585 4 (0.023 2)	—	0.655 5 (0.031 2)	0.635 5 (0.024 6)	—	0.757 7 (0.012 4)	0.757 5 (0.014 7)	70.7			
PEND	0.532 1 (0.003 4)	0.523 6 (0.004 7)	—	0.755 2 (0.015 3)	0.750 7 (0.016 3)	—	0.896 7 (0.011 1)	0.886 2 (0.020 3)	69.2			
CHE	0.596 9 (0.041 0)	0.592 0 (0.039 8)	—	0.573 4 (0.000 6)	0.521 5 (0.029 5)	—	0.810 0 (0.021 9)	0.802 9 (0.023 2)	56.8			
SAT	0.690 1 (0.017 8)	0.688 4 (0.021 0)	—	0.667 6 (0.022 6)	0.666 9 (0.029 2)	—	0.824 7 (0.014 0)	0.823 9 (0.014 3)	47.6			
PENB	0.539 2 (0.041 7)	0.538 5 (0.002 4)	—	0.747 8 (0.010 8)	0.741 0 (0.019 0)	—	0.900 6 (0.013 1)	0.900 3 (0.018 8)	46.3			
SPA	0.652 4 (0.010 9)	0.605 8 (0.000 3)	—	0.605 8 (0.000 3)	0.604 1 (0.001 5)	—	0.820 9 (0.025 2)	0.816 6 (0.030 4)	67.4			
ADU	0.680 0 (0.032 7)	0.674 8 (0.010 7)	—	0.826 0 (0.003 6)	0.813 9 (0.016 7)	—	0.834 7 (0.049 6)	0.834 2 (0.047 8)	78.6			

表 4 wdbc 数据集在分类器上性能比较

分类器	训练结果			测试结果			规则数
	最大值	均值	训练时间	最大值	均值	测试时间	
Zero-order-TSK-FC	0.9580	0.8973	0.0096	0.9577	0.8836	0.0015	10.1
First-order-TSK-FC	0.9830	0.9578	0.0178	0.9759	0.9573	0.0020	8.6
DBN	0.9859	0.9746	0.0946	0.9850	0.9671	0.0019	—
KNN	0.9408	0.8912	0.0131	0.9266	0.8541	0.0115	—
GFS-Adab-cost-C	0.9549	0.9366	—	0.9472	0.9090	—	—
Ensemble-C	0.9771	0.9366	—	0.9507	0.9012	—	—
GFS-Logit-Boost-C	0.9829	0.9331	—	0.9472	0.9362	—	—
EP-Q-TSK	0.9374	0.9064	0.0085	0.9241	0.9035	0.0013	3.8

表 5 wdbc 在 3 个分类器上的规则展示

分类器	if-part														then-part
	L	VH	VH	M	M	VH	VH	VH	L	M	M	VL	VM	M	
1	M	M	H	VL	VL	VH	L	M	M	L	M	VL	VL	L	$a_0^1 = -0.1223$
															$a_0^2 = 0.4450$
2	L	VH	VL	VH	M	VL	L	L	VH	VH	H	M	M	VH	$a_0^1 = -0.6877$
	H	L	VH	VL	H	L	VL	M	VL	L	M	M	M	VL	$a_0^2 = 0.5696$
	VL	L	H	L	VH	VL	M	VL	VL	M	M	H	L	L	$a_0^3 = -0.9116$
3	L	M	H	H	VL	L	M	H	H	M	H	H	L	M	$a_0^1 = 0.0445$
	H	M	VH	M	VL	VL	H	M	L	VL	L	M	VH	L	$a_0^2 = -4.5517$
	H	M	VH	VL	H	L	VL	M	VL	L	M	L	M	VL	$a_0^3 = 1.3103$
	L	M	L	VL	M	H	M	VH	VH	H	VH	L	L	VH	$a_0^4 = 0.3070$

由表 4 可见, EP-Q-TSK 与其他分类器相比, 训练时间最少, 测试时间令人满意. 由于篇幅有限, 本文没有给出这些比较分类器在所有数据集上的训练和测试时间, 但实验记录表明所有其他数据集在各分类器上的训练和测试时间的结果与表 4 中列出的 wdbc 的结果几乎一致. 这是因为 EP-Q-TSK 对所有的模糊子分类器都采用并行学习方式, 增强验证数据集的规模要远远少于训练数据集, 在其上运行的 FCM 和 KNN 时间都很短. 为了展示 EP-Q-TSK 的可解释性, 在某一次取得最优值的运行结果中列出每个子分类器相应模糊规则的前件和后件参数. 如表 5, 对于本次运行, 得到最优规则组合为 (2, 3, 4), 这表明第 1、第 2 和第 3 子分类器分别具有 2、3、4 条模糊规则. 根据表 5, 虽然所有 5 个高斯隶属度函数在不同规则的特征中可能具有不同的宽度值, 但它们总是将其中心保持在 {0, 0.25, 0.5, 0.75, 1} 中, 因此具有各自的语义解释: 非常小、小、中、大、非常大, 简记为 VL、L、M、H、VH. 因此每条模糊规则都是可解释的.

3 结 论

本文提出了一种并行集成具有高可解释的 TSK 模糊分类器 EP-Q-TSK, 以解决现有集成 TSK 模糊分类器可解释性差和计算复杂度高等问题. 根据该框架, 通过并行学习方式快速构建多个具有高可解释性的零阶 TSK 模糊子分类器, 然后利用每个子分类器的输出值作为增强特征对验证数据集进行扩充, 从而形成增强验证数据集. 通过在增强验证数据集上

执行经典的模糊聚类算法 FCM, 获得代表性中心点, 使用 KNN 对预测样本或未知样本的标签进行快速预测, 实验结果验证了 EP-Q-TSK 及其学习算法的有效性.

未来的工作将主要集中在理论和应用方面. 在理论方面, 虽然从实验上验证了增强验证数据集上的 FCM&KNN 比在原始验证数据集上直接使用 KNN 要好, 但应该设法从理论上加以证明. 在实际应用中, 可以开发 EP-Q-TSK 在医疗信息学或众包建模等实际应用场景中的应用^[27].

参考文献 (References)

- [1] Wang L. Fuzzy systems are universal approximators[C]. Proceedings of IEEE International Conferences Fuzzy Systems. San Diego: IEEE, 1992, 10: 1163-1170.
- [2] Jang J S. ANFIS: Adaptive-network-based fuzzy inference systems[J]. IEEE Transactions on Systems, Man, and Cybernetics, 1993, 23(3): 665-685.
- [3] Takagi T, Sugeno M. Fuzzy identification of systems and its application to modeling and control[J]. IEEE Transactions on System, Man, and Cybernetics, 1985, 15(1): 116-132.
- [4] Pedrycz W, Gomide F. Fuzzy systems engineering: Toward human-centric computing[M]. Hoboken: Wiley, 2007: 85-101.
- [5] 杭文龙, 梁爽, 刘解放, 等. 贝叶斯 L2 型 TSK 模糊系统[J]. 控制与决策, 2017, 32(10): 1871-1878. (Hang W L, Liang S, Liu J F, et al. Bayesian

- L2-norm-Takagi-Sugeno-Kang fuzzy system[J]. Control and Decision, 2017, 32(10): 1871-1878.)
- [6] 邓赵红, 张江滨, 蒋亦樟, 等. 基于模糊子空间聚类的0阶岭回归TSK模糊系统[J]. 控制与决策, 2016, 31(5): 882-888.
(Deng Z H, Zhang J B, Jiang Y Z, et al. Fuzzy subspace clustering based 0-order ridge regression TSK fuzzy system[J]. Control and Decision, 2016, 31(5): 882-888.)
- [7] Wei C, Wang L. A note on universal approximation by hierarchical fuzzy systems[J]. Information Science, 2000, 123(3): 241-248.
- [8] Kuncheva L. "Fuzzy" versus "Nonfuzzy" in combining classifiers designed by boosting[J]. IEEE Transactions on Fuzzy Systems, 2003, 11(6): 729-742.
- [9] Torra V. A review of the construction of hierarchical fuzzy systems[J]. International Journal of Intelligent Systems, 2002, 17(5): 531-543.
- [10] Chung F L. On multistage fuzzy neural network modeling[J]. IEEE Transactions on Fuzzy Systems, 2000, 8(2): 125-142.
- [11] Wang Shitong, Chung F L, Shen H B, et al. Cascaded centralized TSK fuzzy system: Universal approximator and high interpretation[J]. Applied Soft Computing, 2005, 5(2): 131-145.
- [12] Kuncheva L I, Whitaker C. Measures of diversity in classifier ensembles[J]. Machine Learning, 2003, 51(2): 181-207.
- [13] Wolpert D H. Stacked generalization[J]. Neural Networks, 1992, 5(2): 241-259.
- [14] Bishop M. Pattern recognition and machine learning[M]. New York: Springer-Verlag, 2006: 128-145.
- [15] 王骏, 王士同, 邓赵红. 聚类分析研究中的若干问题[J]. 控制与决策, 2012, 27(3): 321-328.
(Wang J, Wang S T, Deng Z H. Survey on challenges in clustering analysis research[J]. Control and Decision, 2012, 27(3): 321-328.)
- [16] Nicolás García-Pedrajas, Juan A Romero del Castillo, Gonzalo Cerruela-García. A proposal for local k values for k -nearest neighbor rule[J]. IEEE Transactions on Neural Networks and Learning Systems, 2017, 28(2): 470-475.
- [17] Zadeh L A. Fuzzy sets[J]. Information Control, 1965, 8(3): 338-353.
- [18] Zhang Y, Ishibuchi H, Wang S T. Deep takagi-sugeno-kang fuzzy classifier with shared linguistic fuzzy rules[J]. IEEE Transactions on Fuzzy Systems, 2018, 28(3): 1535-1549.
- [19] Cordon O, Herrera F, Zwir I. Linguistic modeling by hierarchical systems of linguistic rules[J]. IEEE Transactions on Fuzzy Systems, 2002, 10(1): 2-20.
- [20] Deng Z, Jiang Y, Chung F L, et al. Knowledge-leverage-based fuzzy system and its modeling[J]. IEEE Transactions on Fuzzy Systems, 2013, 21(4): 597-609.
- [21] Jiang Y, Chung K F L, Ishibuchi H, et al. Multitask TSK fuzzy system modeling by mining intertask common hidden structure[J]. IEEE Transactions on Cybernetics, 2015, 45(3): 534-547.
- [22] Wang S T, Chung F L. On least learning machine[J]. Journal of Jiangnan University, 2010, 9: 505-510.
- [23] Gu X, Chung F L, Ishibuchi H, et al. Multi-task coupled logistic regression and its fast implementation for large multi-task datasets[J]. IEEE Transactions on Cybernetics, 2015, 5(9): 1953-1966.
- [24] Bache K, Lichman M. UCI machine learning repository 2015[DB/OL]. (2015-10-08)[2018-01-15]. <http://archive.ics.uci.edu/ml>.
- [25] Alcal-Fdez J, Fernández A, Luengo J, et al. KEEL data-mining software tool: Data set repository integration of algorithms and experimental analysis framework[J]. Journal of Multiple-Valued Logic and Soft Computing, 2011, 17(2): 255-287.
- [26] Hinton G E, Osindero S, Teh Y W. A fast learning algorithm for deep belief nets[J]. Neural Computation, 2006, 18(7): 1527-1554.
- [27] Raykar V, Yu S, Zhao L, et al. Learning from crowds[J]. Journal of Machine Learning Research, 2010, 11(2): 1297-1322.

作者简介

张雄涛(1984—), 男, 博士生, 从事人工智能与模式识别、机器学习的研究, E-mail: 1047897965@qq.com;

蒋云良(1967—), 男, 教授, 博士生导师, 从事智能计算、数据挖掘等研究, E-mail: jyl@zjhu.edu.cn;

胡文军(1979—), 男, 教授, 博士, 从事机器学习、模式识别等研究, E-mail: hooweujun@yahoo.com.cn;

王士同(1964—), 男, 教授, 博士生导师, 从事模式识别与机器学习等研究, E-mail: wxwangst@yahoo.com.cn.

(责任编辑: 郑晓蕾)