计 算 机 研 究 与 发 展
Journal of Computer Research and Development
https://crad.ict.ac.cn
62(11): 2710−2724, 2025

基于粗糙超立方的联邦进化特征选择算法

陈雪颖[1] 罗 川[1] 李天瑞[2] 陈红梅[2]
[1](四川大学计算机学院 成都 610065)
[2](西南交通大学计算机与人工智能学院 成都 611756)
(chenxueying@stu.scu.edu.cn)

Federated Evolutionary Feature Selection Algorithm Based on Rough Hypercuboid

Chen Xueying[1], Luo Chuan[1], Li Tianrui[2], and Chen Hongmei[2]
[1](*College of Computer Science*, *Sichuan University*, *Chengdu* 610065)
[2](*School of Computing and Artificial Intelligence*, *Southwest Jiaotong University*, *Chengdu* 611756)

Abstract Feature selection is an effective technique of dimensionality reduction in the field of machine learning. In the era of big data, data security has become an issue of great concern nowadays, and how to perform the feature selection task under the premise of privacy protection is a challenging scientific problem that needs to be solved urgently. Rough hypercuboid is an uncertainty approximation computational model combining rough set theory and hypercuboid learning, which provides an efficient feature selection method for numerical approximate classification problems by introducing supervised information granulation technique and multiple feature evaluation criteria. In this paper, we propose a novel multi-party federated feature selection algorithm under privacy protection, based on the rough hypercuboid model and particle swarm optimization algorithm. Firstly, a centralized (client/server) federated feature selection architecture for multi-party participation is established. Based on the architecture, the rough hypercuboid model and the particle swarm optimization algorithm are used to search the optimal feature subset on the client, and a novel global feature subset evaluation strategy for multiple participants is proposed on the server. Then, the ability of the proposed algorithm to select features in collaboration with multiple participants is improved by designing a particle initialization strategy in a federated environment. Finally, experimental results on the twelve UCI benchmark datasets show that compared with the other six traditional feature selection algorithms, the subset of features selected by the proposed algorithm has higher classification performance on each participant under the premise of satisfying the data privacy protection.

Key words feature selection; rough hypercuboid; federated learning; particle swarm optimization; privacy protection

摘 要 特征选择是机器学习领域中数据降维的有效手段.在大数据时代,数据安全成为了当今社会中备受关注的问题,如何在隐私保护的前提下完成特征选择任务是亟需解决的一个挑战性科学问题.粗糙超立方是一种结合粗糙集理论和超立方学习的不确定性近似计算模型,通过引入有监督的信息粒化技术和多重特征评估准则,为数值型近似分类问题提供了一种高效的特征选择方法.将粗糙集超立方模型和粒子群优化算法相结合,提出了一种新颖的隐私保护下多方参与的联邦特征选择算法.首先,该算法建立了一种适用于多方参与的集中式(客户端/服务器)联邦特征选择架构.在客户端上利用粗糙集超立方模型

收稿日期: 2024-06-03; **修回日期**: 2025-02-17
基金项目: 国家自然科学基金项目(62476182, 62076171, 62376230)
This work was supported by the National Natural Science Foundation of China (62476182, 62076171, 62376230).
通信作者: 罗川(cluo@scu.edu.cn)

和粒子群优化算法搜索本地最优特征子集，同时在服务器端给出了一种适应多参与方的全局特征子集评估策略. 然后，通过设计联邦环境下的粒子初始化策略提高了算法在多参与方下协同特征选择能力. 最后，在 12 组 UCI 基准数据集上的实验结果表明，相比于其他 6 种传统特征选择算法，在满足各参与方数据隐私保护的前提下，算法所选择出的特征子集在各参与方上具有更好的分类性能表现.

关键词 特征选择；粗糙超立方；联邦学习；粒子群优化；隐私保护

中图法分类号 TP391

DOI: 10.7544/issn1000-1239.202440422 **CSTR:** 32373.14.issn1000-1239.202440422

大数据时代，社会各个领域都在不断地产生着体量大、维度高和动态演化的数据，人们利用这些数据进行模型训练和知识挖掘. 但数据中所蕴含的许多不相关、冗余的特征，将造成额外的内存需求和数据处理时间的浪费，同时也会给所训练模型的学习性能带来负面影响. 为了能够高效地利用高维数据，需要对其进行冗余特征的剔除和相关特征的保留. 特征选择是一项至关重要的数据预处理技术，其能够从高维数据中获取不包含冗余信息的特征子集，有效地降低了数据的特征维度，带来更好的学习性能和更高的计算效率[1]. 近年来，特征选择技术已经广泛应用于众多领域，如文本挖掘[2]、信息检索[3]和图像处理[4]等领域.

高维数据中广泛存在着的冗余信息和不相关特征会对决策结果产生负面影响，也给从数据中获取的知识增添了不确定性[5]. 粗糙集是一种处理不精确、不一致和不完备信息的不确定性数据分析方法[6]，该方法最大的优势在于不需要提供除数据本身以外的任何先验知识来对知识不确定性进行分析和推理，并能够借此从数据中挖掘到潜在的有价值知识，其在数据挖掘、知识发现等领域受到了广泛关注. 经典粗糙集理论通常仅能处理名义型特征描述的离散数据. 然而，现实世界中的数据具有多样性，将其进行扩展是非常有必要的. 模糊粗糙集[7]、邻域粗糙集[8]和粗糙超立方[9]等是在经典粗糙集理论的基础上进行扩展后的模型，这些方法被相继运用于数值型数据的特征选择任务中. 获取最优特征子集最直接的策略是生成所有可能的特征子集，并选择具有最优准则的子集，显然，这是一个 NP-难问题[10]，这类方法只适用于非常简单的小规模数据集. 因此，针对大规模复杂数据集的全局特征寻优，可以考虑引入群体智能优化算法，例如：粒子群优化（particle swarm optimization, PSO）算法[11]、蚁群算法[12]、基因算法[13]等. PSO 算法是一种流行的元启发式进化优化技术，由 Kennedy 等人[14]于 1995 年提出，其灵感来自鱼群和鸟群的集群行为. 作为一种群体智能进化算法，PSO 算法以其收敛速度快、参数较少和易于实现等优势，在特征选择问题中受到了广泛关注[15]. 近年来，越来越多的研究工作考虑到将粗糙集理论和粒子群优化算法相结合，并将其用于特征选择任务中.

高维度的大规模数据为机器学习算法模型提供了丰富多样的学习样本，提高了模型的训练效果. 然而，随着大数据技术的快速发展，人们在利用大数据信息时越来越关注数据的隐私安全问题. 这也导致了在实际运用中，能够有效使用的数据规模往往是有限的，如不同银行存储的客户数据、不同医院收集的典型病例数据等. 受限于用户隐私和数据安全保护的需求，这些机构无法互相共享隐私数据，这也造成了大数据背景下典型的数据孤岛现象，虽然拥有海量的数据，但由于隐私安全问题，数据不能被连接，从而导致数据被孤立到不同的组织和机构中. 为了应对传统集中式学习的应用受限，联邦学习的概念被提出，这是一种新兴的分布式机器学习框架，满足隐私保护的要求，不涉及原始数据集的共享，并具有安全的加密技术[16]. 在联邦学习中，不同参与方协作训练机器学习模型，同时确保每个参与方的数据集不被共享使用，即每个参与方的私有数据集不会被集中存储. 同时，可以对中央聚合服务器和本地数据持有者之间的通信进行加密，以防止信息泄露. 传统场景下，即数据可以集中存储共享的场景，特征选择问题的相关研究已经存在很多，然而，这类研究不满足现阶段公众所要求的数据隐私保护问题. 隐私保护前提下的联邦特征选择技术已成为了近年来该领域的一个研究热点.

本文针对横向联邦学习场景提出了一种基于粗糙超立方的联邦进化特征选择算法（federated evolutionary feature selection algorithm based on rough hypercuboid, FERHFS）. 针对数据独立分布存储于客户端上且互不共享的场景，该算法提出了一种集中式（客户端/服务器）联邦特征选择架构. 基于该架构，训练

过程中可通过服务器端交换有效信息以带来更高的算法性能. 同时, 该方法引入了 FERHFS, 并在服务器端设置了基于粗糙超立方的特征子集评估策略以指导算法的逐层迭代选择.

总体来说, 本文的主要贡献归纳为 3 个方面:

1)提出了一种集中式(客户端/服务器)联邦特征选择架构, 其私有数据只能由客户端单独拥有, 算法训练过程中可通过服务器端交换有效信息;

2)基于所提出的集中式(客户端/服务器)联邦特征选择架构, 结合粗糙超立方模型和粒子群优化技术提出了一种新颖的联邦进化特征选择算法, 并通过改进粒子群算法的适应度函数和初始化策略带来更高的算法性能;

3)在服务器端设置基于粗糙超立方的特征子集评估策略, 其目的在于选择一个适应多参与方的全局最优特征子集, 为算法下一次迭代提供先验指导, 提高多参与方协同合作的能力.

1 相关工作

特征选择方法旨在通过删除冗余特征, 从高维数据集中挑选到一个最优特征子集, 在减少数据特征维度的同时保证算法的精度, 是一项重要的数据预处理技术. 传统的特征选择方法根据特征评估准则的不同, 常常可以分为过滤式和封装式[17]. 过滤式特征选择算法往往通过评估每个特征与决策属性之间的关系或特征子集内部的关系来选择相关特征, 这类特征选择算法通常更为高效且更适用于多数实际应用领域. 封装式特征选择算法在选择特征子集的过程中, 采用学习算法的性能作为特征子集的评估准则. 此外, 一些学者还提出了嵌入式特征选择算法, 这类算法倾向于将特征选择过程与学习算法融合. 另外, 特征选择过程的本质即为搜索最优特征子集, 因此, 按照搜索策略的不同, 可以将特征选择算法分为完全搜索、启发式搜索和随机搜索 3 类[18]. 完全搜索算法往往通过评估所有可能特征子集的价值来寻找最优解, 所以该类算法消耗的时间成本较高, 不适用于高维数据. 启发式搜索算法在特征搜索过程中每次都选择一个使得评价函数的取值达到最优的特征加入, 本质上是一种贪心算法, 该类算法由于缺乏全局性, 搜索能力有限, 容易陷入局部最优. 随机搜索算法通常随机产生一个特征子集, 然后在该子集的基础上融入随机步骤迭代选择下一代特征子集, 由于采用随机策略, 这类方法得到的结果通常具有较高的不确定性. 近年来, 为进一步提高特征选择算法的全局搜索性和泛化性, 学者们将进化优化技术和特征选择算法结合, 提出了许多进化特征选择算法, 遗传算法(genetic algorithm, GA)是最开始被应用于特征选择任务的一种进化算法[19], 随后, 粒子群算法由于其收敛速度快、参数较少和易于实现等优势, 逐渐被广泛运用于特征选择任务中. 例如, Chen 等人[20]结合多任务优化, 提出了一种基于 PSO 算法的进化特征选择算法, 通过数据集生成的 2 个相关任务之间的信息共享来解决高维分类问题. Shafqat 等人[21]提出了一种基于集成学习和粒子群优化的特征选择方法, 设计了一种基于 PSO 算法的微调超参数机制以解决特征选择问题. 此外, Song 等人[22]针对数据降维问题, 结合相关引导聚类和 PSO 算法, 提出了一种基于对称不确定性的特征删除策略. 粗糙集理论是一种处理不精确、不一致和不完备信息的数据分析理论, 该理论最大的优势在于可以提供来自数据集本身的先验知识对知识不确定性进行分析和推理, 并能够借此从数据中挖掘到潜在的有价值的知识. 基于粒子群优化算法在特征选择任务上的有效性, 很多学者提出了结合粗糙集理论和粒子群算法的特征选择技术. 例如, Kumar 等人[23]结合邻域粗糙集模型和粒子群算法, 提出了一种用于多类别动作想象分类的特征选择方法. Huda 等人[24]提出了一种基于粒子群算法和模糊粗糙适应度函数的特征选择算法, 用于从具有大规模特征的数据集中获得最优特征子集.

虽然现有的一些特征选择方法结合了粒子群算法和粗糙集理论的优势, 提高了特征选择技术的性能, 但上述方法都是针对传统场景下, 即数据集中存储场景下的特征选择问题. 面对现代数字社会日益重要的数据隐私保护需求, 这类方法无法有效处理数据隐私保护下的特征选择问题. 联邦学习是一种隐私限制下的新型机器学习范式, 其在满足分散的多个机构用户隐私和数据安全保护前提下, 使多参与方协同完成机器学习建模的框架. 这种机制可以在不暴露原始数据的同时, 协助多参与方上的算法模型训练, 实现学习模型的共同提升[25]. 在联邦学习的场景下, 由于数据的分散性, 若在各参与方上直接使用传统的特征选择方法, 各参与方之间缺乏有效的协作机制, 挑选的最优特征子集通常不具备全局最优性. 因此, 联邦学习场景下的特征选择算法, 需要考虑设计各参与方之间的信息共享机制, 并且保证在特征选择模型的训练过程中, 原始数据不会在

各参与方之间传递.

近年来，由于隐私保护和数据安全意识的提高，在完成特征选择等数据预处理步骤时，学者们常常使用到联邦学习的思路. 例如，Qin 等人[26]提出了一种基于联邦特征选择技术的入侵检测系统，用于优化资源受限环境下的入侵检测，但在该系统中，使用贪婪方法搜索特征子集，不利于获取全局最优解. Feng 等人[27]为分布式垂直联邦场景设计了一种基于多视图学习的多参与方特征选择算法，然而，该方法需要在参与方之间共享标签，对大规模数据通信的效率会严重下降. Zhang 等人[28]提出了一种用于物联网中横向联邦学习场景下的无监督联合特征选择算法，该算法设计了特征相关性分层聚类策略用于重叠特征选择. Hu 等人[29]提出了一种基于粒子群优化的联邦特征选择算法，实现了隐私保护，然而，该算法需要在数据私有方之间交换部分数据信息，增加了时间消耗和隐私信息泄露的风险.

2 预备知识

在本节中，我们将介绍粗糙超立方方法[9]、联邦学习[16]和粒子群优化算法[14]的相关知识.

2.1 粗糙超立方方法

假设$S=(U, C\cup D)$是一个知识表达系统，其中$U=\{u_1,u_2,\cdots,u_n\}$是一个包含 n 个对象的非空有限集合，$C=\{a_1,a_2,\cdots,a_m\}$和 D 分别是 U 中的非空有限条件特征集和决策特征集. 由决策特征集 D 生成的 U 的等价关系记为$U/D=\{\beta_1,\beta_2,\cdots,\beta_l\}$，$l$ 表示 U 相对于决策特征集 D 有 l 个类别. 条件特征 $a_k\in C$ 相对于第 i 个等价类 β_i 的值域表示为区间$[L_i,U_i]$，该区间用于表示由决策类信息监督的数据粒化. 同样，条件特征集 C 生成的 U 的等价类记为$U/C=\{\sigma_1,\sigma_2,\cdots,\sigma_l\}$. 因此，条件特征 a_k 的粗糙超立方等价划分矩阵定义为

$$\boldsymbol{H}(a_k)=\begin{pmatrix} h_{11}(a_k) & h_{12}(a_k) & \cdots & h_{1n}(a_k) \\ h_{21}(a_k) & h_{22}(a_k) & \cdots & h_{2n}(a_k) \\ \vdots & \vdots & & \vdots \\ h_{l1}(a_k) & h_{l2}(a_k) & \cdots & h_{ln}(a_k) \end{pmatrix}, \tag{1}$$

$$h_{ij}(ak)=\begin{cases}1, L_i\leqslant x_j(ak)\leqslant U_i,\\ 0, \text{其他},\end{cases} \tag{2}$$

其中 $x_j(a_k)$ 是对象 x_j 的第 k 个条件特征值，$h_{ij}(a_k)$ 表示特征 a_k 下对象 x_j 相对于等价类 β_i 的隶属度，并满足：

$$1\leqslant\sum_{j=1}^{n}h_{ij}(a_k)\leqslant n,\forall i;1\leqslant\sum_{i=1}^{c}h_{ij}(a_k)\leqslant c,\forall j. \tag{3}$$

不同特征的值域区间互相交叉，形成隐式超立方体，通过寻找误分类对象的隐式超立方体来评估决策特征相对于条件特征的依赖程度. 误分类对象可以用混淆矩阵表示为

$$\boldsymbol{V}(a_k)=(v_1(a_k),v_2(a_k),\cdots,v_n(a_k)), \tag{4}$$

其中

$$v_j(a_k)=\min\left\{1,\sum_{i=1}^{l}h_{ij}(a_k)-1\right\}. \tag{5}$$

给定等价类$\beta_i\in U/D$，β_i在特征a_k下的粗糙近似集可由a_k的超立方体等价划分矩阵和混淆向量表示为

$$\underline{A}(\beta_i)=\left\{u_j\middle|h_{ij}(a_k)=1 \text{ and } v_j(a_k)=0\right\}, \tag{6}$$

$$\overline{A}(\beta_i)=\left\{u_j\middle|h_{ij}(a_k)=1\right\}, \tag{7}$$

其中 A 为由条件特征a_k得到的等价关系.

第 i 个超立方等价划分即为 $l\times n$ 超立方等价划分矩阵 $\boldsymbol{H}(a_k)$ 的第 i 行，可被表示为

$$\beta_i=\{h_{i1}(a_k)/u_1+h_{i2}(a_k)/u_2+\cdots+h_{in}(a_k)/u_n\}, \tag{8}$$

其中“+”表示 2 个集合之间的并集运算符. 等价类 β_i 的上下近似集的基数可被计算为

$$|\underline{A}(\beta_i)|=\sum_{j=1}^{n}h_{ij}(a_k)\wedge\left[1-v_j(a_k)\right], \tag{9}$$

$$\left|\overline{A}(\beta_i)\right|=\sum_{j=1}^{n}h_{ij}(a_k). \tag{10}$$

因此，等价类 β_i 的边界域可表示为

$$BND_A(\beta_i)=\left\{u_j|h_{ij}(a_k)=1 \text{ and } v_j(a_k)=1\right\}, \tag{11}$$

同时，等价类 β_i 边界域的基数可被计算为

$$|BND_A(\beta_i)|=\sum_{j=1}^{n}h_{ij}(a_k)\wedge v_j(a_k). \tag{12}$$

给定一个决策特征集 D，决策划分 U/D 的正、负和边界域定义为：

$$PSO_A(D)=\bigcup_{\beta_i\in U/D}\underline{A}(\beta_i), \tag{13}$$

$$NEG_A(D)=U-\bigcup_{\beta_i\in U/D}\overline{A}(\beta_i), \tag{14}$$

$$BND_A(D)=\bigcup_{\beta_i\in U/D}\overline{A}(\beta_i)-\bigcup_{\beta_i\in U/D}\underline{A}(\beta_i). \tag{15}$$

根据超立方体等价划分矩阵和 a_k 的混淆向量，U/D 的决策正域的基数可以表示为

$$|PSO_A(D)|=\sum_{i=1}^{l}\sum_{j=1}^{n}h_{ij}(a_k)\wedge\left(1-v_j(a_k)\right). \tag{16}$$

因此, 条件属性 a_k 和决策特征集 D 的依赖度定义为

$$\gamma_{a_k}(D)=\frac{1}{n}\sum_{i=1}^{l}\sum_{j=1}^{n}h_{ij}(a_k)\wedge\left(1-v_j(a_k)\right)=1-\frac{1}{n}\sum_{j=1}^{n}v_j(a_k),\tag{17}$$

其中$0\leqslant\gamma_{a_k}(D)\leqslant 1$.

同样, 给定 2 个条件特征 a_k 和 a_m, 特征子集 $\{a_k, a_m\}$的超立方体等价划分矩阵可以计算为 $\boldsymbol{H}(\{a_k, a_m\})=\boldsymbol{H}(a_k)\wedge\boldsymbol{H}(a_m)$, 其中 $h_{ij}(\{a_k,a_m\})=h_{ij}(a_k)\wedge h_{ij}(a_m)$. 因此特征 a_k 相对于特征子集 $\{a_k,a_m\}$ 的重要度为

$$\sigma_{\{a_k,a_m\}}(D,a_k)=\frac{1}{n}\sum_{j=1}^{n}[v_j(\{a_m\})-v_j(\{a_k,a_m\})],\tag{18}$$

其中$0\leqslant\sigma_{\{a_k,a_m\}}(D,a_k)\leqslant 1$.

根据上述定义, 可以定义所选特征子集相对于决策特征集的相关度、依赖度和重要度. 已选特征子集 $S(S\subseteq C)$ 与决策特征集 D 的平均相关度为

$$J_{\text{relev}}=\frac{1}{|S|}\sum_{a_i\in S}\gamma_{a_i}(D).\tag{19}$$

决策特征集 D 对特征子集 S 上的依赖度为

$$J_{\text{depen}}=\gamma_S(D).\tag{20}$$

特征子集 S 的平均重要度为

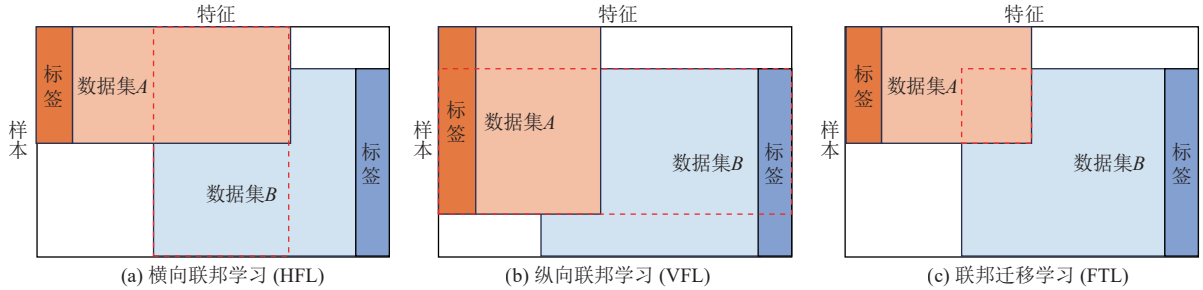
$$J_{\text{signf}}=\frac{\sum_{a_i\neq a_j\in S}\{\sigma_{\{a_i,a_j\}}(D,a_i)+\sigma_{\{a_i,a_j\}}(D,a_j)\}}{|S|(|S|-1)}.\tag{21}$$

因此, 我们可以利用上述定义的式(21)来构造特征选择优化函数, 并以此选择最优条件特征子集[9].

2.2 联邦学习

联邦学习的目的是为多站点参与的机器学习提供一个通用模型, 模型训练过程中个体间可以交换有利信息来提高算法的性能, 且这种操作不会导致各自数据的交换, 训练好的模型可以在各站点之间共享. 在这种联邦机制的背景下, 既保证了每个数据拥有者的数据隐私安全, 又有效地实现了各参与方的“共同富裕”, 即实现机器学习模型的共同提升, 避免了因单一参与方数据量不足而导致的模型性能较差和泛化能力不足的问题. 作为一种分布式的机器学习框架, 根据框架结构可以将联邦学习分为点对点的联邦学习框架和集中式的联邦学习框架 2 类[30]. 点对点的联邦学习框架是指所提出的学习框架是去中心化的, 不存在中央服务器; 集中式的联邦学习框架是指所提出的学习框架是基于中央服务器的, 其拥有多个客户端设备和 1 个服务器设备. 相比于点对点的联邦学习框架, 集中式的联邦学习框架更适应数据分布复杂的场景, 可以完成更多样的学习任务.

根据数据样本不同空间的划分, 联邦学习又可以分为横向联邦学习(horizontal federated learning, HFL)、纵向联邦学习(vertical federated learning, VFL)和联邦迁移学习(federated transfer learning, FTL)[31], 具体表示如图 1 所示. HFL 适用于特征空间部分重叠较多及样本空间部分重叠较少的数据集. 例如, 针对不同城市的银行数据建模场景, 数据集在用户空间上重叠较少, 但特征空间数据类似. 此外, 以不同医院间的典型疾病诊断数据集为例, 对于同种典型疾病, 数据特征空间相同, 显然, 不同医院所提供的数据集在样本空间重叠是较少的. HFL 为上述隐私敏感计算场景的研究带来了更多的可能性. VFL 适用于具有相同样本空间的数据集, 但特征空间不同的场景. 例如, 同一地区的银行与运营商之间的联邦建模, 又比如同一地区的新闻推荐系统和视频推荐系统之间的联邦建模, 这类场景都有同样的特质, 即参与方的数据集用户空间交集较大, 但特征空间重叠较少. FTL 适用于数据集的样本和特征都重叠较少的情况, 例如不同地区的银行和电商公司间的联邦建模, 参与方所提供数据集的样本和特征空间重叠都较少, 利用联邦迁移技术可以有效地帮助标签数据少的数据集在保证数据安全前提下更好地进行机器



(a) 横向联邦学习 (HFL) (b) 纵向联邦学习 (VFL) (c) 联邦迁移学习 (FTL)

Fig. 1 Three categories of federated learning

图 1 联邦学习的 3 种分类

学习研究. 表 1 对比了 3 种联邦学习框架的区别. 基于 HFL 在扩展样本空间上的有效性，联邦学习领域内的大多数研究都集中在 HFL 场景，本文重点研究 HFL 下的特征选择任务.

Table 1 Comparison of Data Distribution Patterns in Different Federated Learning Paradigms

表 1 不同联邦学习范式的数据分布模式对比

联邦学习	样本重叠度	特征重叠度	可用标签
HFL	高	低	全部
VFL	低	高	部分
FTL	低	低	部分

2.3 粒子群优化算法

粒子群优化算法是通过研究鱼群和鸟群觅食行为所提出的一种群体智能进化算法，其中，粒子是候选解，群是指 1 组候选解，优化即指选择全局最佳粒子，该算法的目的在于从候选解中找到最优解. 在算法中，粒子群在一开始就被初始化，根据每个粒子的速度和位置在多维空间搜索，从而获得全局最优解[11]. 在粒子群优化算法中，假设在一个 D 维目标搜索空间，每个粒子都是空间中的 1 个点，总共有 M 个粒子，其中，第 i 个粒子的位置向量被表示为 $\boldsymbol{X}_i=(x_{i1},x_{i2},\cdots,x_{iD})$，第 i 个粒子的速度向量被表示为 $\boldsymbol{V}_i=(v_{i1},v_{i2},\cdots,v_{iD})$. 在迭代过程中，每个粒子通过个体最佳位置 $pbest$ 和全局最佳位置 $gbest$ 更新其当前位置. 为了找到全局最优解，首先对所有粒子进行随机初始化，并同时更新粒子的速度和位置. 在每次迭代的过程中，第 i 个粒子会依据式 (22) (23) 对其当前位置和速度进行更新，随机移动以寻找全局最优的解决方案.

$$v_i^{D(t+1)}=w\times v_i^D(t)+c_1\times r_1\times(pbest_i^D(t)-x_i^D(t))+c_2\times r_2\times(gbest_i^D(t)-x_i^D(t)),\tag{22}$$

$$x_i^D(t+1)=x_i^D(t)+v_i^D(t+1),\tag{23}$$

其中 D 为粒子的维数，t 为搜索过程中的迭代次数，ω 为惯性权重，c_1 和 c_2 为加速度常数，r_1 和 r_2 是在区间 [0,1] 中随机选取的整数.

3 联邦进化特征选择算法

本文基于 HFL 框架，结合粗糙超立方模型和离散粒子群算法提出一种新的联邦特征选择方法. 下面将详细阐述本文算法的关键步骤.

3.1 问题定义

假设有 N 个本地设备 $F_i(i=1,2,\cdots,N)$，每个本地设备拥有各自的数据集，并且所有数据集在特征空间上有大部分重叠且样本空间重叠较少. 本文研究的问题可以描述为：在满足每个本地设备隐私保护需求下，在 D 维数据集中选择适用于所有设备的全局最优特征子集，同时，希望所得到的全局最优特征子集具有较大的泛化性能.

3.2 联邦特征选择架构

为了满足多参与方隐私保护的需要，受联邦学习思想的启发，针对 HFL 的情景，提出了一个集中式联邦(客户端/服务器)特征性选择架构. 客户端指多个本地设备，分别存储各自的隐私数据；算法训练过程中，服务器作为交换有效信息的中央平台. 图 2 给出了联邦特征选择架构的示意图.

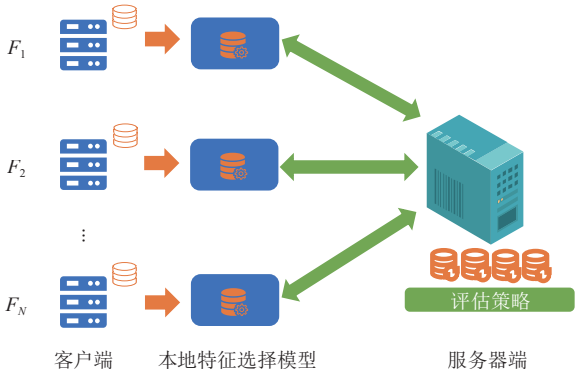


Fig. 2 Federated feature selection architecture

图 2 联邦特征选择架构

在所提出的联邦特征选择架构中，首先，在每个本地设备上，根据改进后的粒子群算法训练得到本地特征选择模型，并获取本地最优子集. 其次，迭代过程中，本地设备将各自的本地最优子集传送到服务器上，在服务器上设置一种基于粗糙超立方的特征评估策略，从而建立一个全局最优的特征选择模型. 最后，通过服务器交换有效信息，即服务器将全局最优子集传送给每个本地设备以指导下一轮次算法学习. 特别说明的是有效信息指每个本地设备算法训练中的过程矩阵值和本地最优特征子集的索引，这并不会导致数据泄露. FERHFS 可分为 2 个阶段：一个阶段是在客户端上运行的粒子群优化算法；另一个阶段是在服务器端运行的特征子集评估策略. 同时，整个算法分内循环和外循环，内循环指客户端的粒子群优化算法，将内循环次数称为迭代次数 M，外循环次数指包含客户端和服务器端的整个算法循环次数，将其称为轮回次数 T，内循环算法提供了算法训练基础，外循环算法给内循环算法提供指导经验. 算法 1 给出了 FERHFS 算法的伪代码.

算法 1. FERHFS 算法.

输入: N 个本地设备使用的数据集;

输出: 全局最优特征子集向量 $\boldsymbol{X}^*$.

① while 外循环终止条件未满足 do

/* N 个本地设备同时执行如下操作 */

② 根据式(19)~(21)计算特征相关度、依赖度和重要度矩阵 $\boldsymbol{Sig}$;

③ 将相关度、依赖度和重要度矩阵传送给服务器;

④ 接收来自服务器的全局最优子集向量 $\boldsymbol{X}^*$;

⑤ while 内循环终止条件未满足 do

⑥ 采用 3.5 节提出的粒子初始化策略初始化种群;

⑦ 离散粒子群算法更新粒子位置, 产生本地最优子集;

⑧ 将本地最优子集传送给服务器;

⑨ end while

/* 中央服务器执行如下操作 */

⑩ 接收来自 N 个客户端的相关度、依赖度和重要度矩阵;

⑪ 接收来自 N 个客户端的本地最优特征子集向量 $\boldsymbol{X}_i, i=1,2,\cdots,N$;

⑫ 由 3.4 节评估策略选择全局最优子集向量 $\boldsymbol{X}^*$;

⑬ 将 $\boldsymbol{X}^*$ 发送给 N 个客户端;

⑭ end while

3.3 适应度函数优化

在联邦环境下, 每个本地设备选择离散粒子群优化(binary particle swarm optimization, BPSO)算法训练特征选择模型. BPSO 算法是对 PSO 算法应用于特征选择问题的一种扩展算法, 其中, 第 i 个粒子的位置向量 $\boldsymbol{X}_i=(x_{i1},x_{i2},\cdots,x_{iD})$可以用二进制变量表示, 即 x_{ij}=1 或 x_{ij}=0, 是由 Sigmoid 函数将速度向量的连续值映射到 [0,1].

$$s(v_{ij})=\frac{1}{1+\mathrm{e}^{-v_{ij}}}. \tag{24}$$

粒子位置更新公式为

$$x_i^D(t+1)=\begin{cases}1, & rand<s(v_t^D(t)).\\ 0, & \text{其他}.\end{cases} \tag{25}$$

在基于 BPSO 的特征选择问题中, 粒子的位置向量的每一维即代表 1 个特征, 当其值为 1 时代表该特征被选择. 因此, 当粒子位置向量中的某一维对应的速度越大时, 该特征被选择的概率也相应越高.

在离散粒子群算法中, 寻找全局最优解是通过评估当前解的适应度函数值, 适应度函数的优劣会显著影响所选择最优特征子集的质量. 在以往的基于 BPSO 的特征选择算法中, 通常采用某个分类器所获得的精确度作为适应度值, 这种做法得到的最优特征子集往往不具备泛化性, 且在迭代过程中计算分类器精确度所消耗的时间成本也较大. 因此, 我们结合粗糙超立方模型的优势来改进 BPSO 中的适应度函数. 基于粗糙超立方模型, 由式(19)~(21)可以计算得到特征的相关度、依赖度和重要度. 利用这 3 方面的评价标准构造 BPSO 算法的适应度函数, 可以帮助特征选择模型更好地获取全局最优解. 因此, 结合粗糙超立方构造适应度函数, 得到的最优特征子集更具备泛化性.

结合粗糙超立方构造适应度函数, 由式(19)~(21), 将 BPSO 算法的适应度函数定义为:

$$Fitness(\boldsymbol{X})=\mu\frac{J_{\mathrm{relev}}}{\max\limits_{a_i\in S}\{\gamma_{a_i}(D)\}}+\nu(1-\mu)J_{\mathrm{depen}}+(1-\mu)(1-\nu)\frac{2\times J_{\mathrm{signf}}}{\max\{\boldsymbol{Sig}\}}, \tag{26}$$

其中 μ 和 ν 是 2 个权重参数, $\boldsymbol{Sig}=(sig_{ij})$是特征重要度矩阵.

3.4 服务器的评估策略

粒子群初始化的质量会直接影响算法的性能. 在所提出的联邦特征选择算法中, 我们期望在每一轮的计算中, 中央服务器都可以向本地设备传递有利信息, 从而指导本地设备下一轮的粒子初始化. 显然, 原始数据集不能被其他设备共享也不能泄露到中央服务器. 每一轮算法中, N 个本地设备使用私有数据训练模型, 生成 N 个本地最优特征子集, 并将其传输到中央服务器. 同时, 为了完成后续的评估策略, 每个本地设备将各自的相关度、依赖度和重要度矩阵传输到中央服务器. 因此, 服务器上应存储 N 个本地最优特征子集和 $3\times N$ 个矩阵.

对于来自第 i 个设备的本地最优特征子集向量 $\boldsymbol{X}_i$, 首先, 基于中央服务器上 N 个参与方的相关度、依赖度和重要度矩阵值, 在中心服务器上计算 N 个适应度值, 即 $Fitness(\boldsymbol{X}_i)_j, j=1,2,\cdots,N$. 然后根据这 N 个适应度值计算该本地最优特征子集向量 $\boldsymbol{X}_i$的平均适应度值并将其作为 $\boldsymbol{X}_i$的得分, 比较 N 个本地最优特征子集的得分, 将得分最大的子集返回给 N 个本地设备作为全局最优子集向量 $\boldsymbol{X}^*$, 指导本地设备进行下一轮算法学习, 具体如下所示:

$$\boldsymbol{X}^*=\max\left\{\boldsymbol{X}_i\middle|\frac{1}{N}\sum_{j=1}^{N}Fitness(\boldsymbol{X}_i)\right\}, \tag{27}$$

其中 $\boldsymbol{X}_i$表示来自第 i 个本地设备上特征选择算法产生的本地最优特征子集向量.

3.5 联邦环境下的粒子初始化策略

在本文所提出的算法中，粒子初始化阶段可以引入从粗糙超立方模型学习到的先验知识，同时，来自服务器端的有效信息也可作为粒子初始化的先验知识. 鉴于此，本地设备可以产生更高质量的粒子种群，从而更好地获得最优解. 以特征维度为 D 的数据集为例，在某个设备上的第 i 个粒子向量 $\boldsymbol{X}_i=(x_{i1}, x_{i2}, \cdots, x_{iD})$，$x_{ij}\in[0,1]$为例，其中，$x_{ij}$ 表示第 j 个特征被选中的概率. 假设在某一轮的算法结束时，经中央服务器评估给出的全局最优特征子集向量 $\boldsymbol{X}^*$ 为来自第 k 个本地设备 F_k 所提供的本地最优子集向量，即 $\boldsymbol{X}^*=\boldsymbol{X}_k$.

首先，在第 1 轮 BPSO 算法中，即 T=1 时，粒子初始化可以结合基于粗糙超立方模型的先验知识，指导生成高质量的粒子，其定义为

$$x_{ij}=\begin{cases}1, rand\leqslant 1-\dfrac{rank(\gamma_{a_i}(D))}{m},\\ 0, \text{其他},\end{cases} \tag{28}$$

其中 $rank(\gamma_{a_i}(D))$ 指特征 a_i 在所有特征相关度降序排列后的序值，m 为所有特征的总数目，$rand$ 为介于 [0, 1] 间的随机数.

其次，当 $T\neq 1$时，考虑第 k 个本地设备 F_k，在当前轮次的算法中，全局最优特征子集向量 $\boldsymbol{X}^*$ 将指导粒子初始化，具体表现在：$\boldsymbol{X}^*$ 中被选中的特征，很大程度上表明该特征是优秀的，应该增加该特征被选择的概率；同理，在 $\boldsymbol{X}^*$ 中没有被选中的特征，则说明该特征不相关的概率很高，该特征被选择的概率应该降低. 所以，第 i 个粒子的初始化可以定义为：

$$x_{ij}=(1-\sigma)\times rand+\sigma\times x_j^*, j=1,2,\cdots,m, \tag{29}$$

其中 x_j^* 代表在 $\boldsymbol{X}^*$ 中第 j 个特征的值，σ 是 [0, 1] 范围内的参数，其目的为控制 $\boldsymbol{X}^*$ 对粒子初始化的影响，$rand$ 为介于 [0, 1] 间的随机数.

最后，当 T≠1，即在除第 1 轮次的后续迭代中，考虑除第 k 个本地设备 F_k 以外的其他本地设备 F_l，假设 $\boldsymbol{X}^*$ 选中了第 j 个特征，说明该特征在很大程度上是优秀的，应该增加其被选择的概率；同理，假设在 $\boldsymbol{X}^*$ 和 $\boldsymbol{X}_l$ 中都没有选择该特征，则该特征较大的概率为不相关特征，若 $\boldsymbol{X}^*$ 未选中该特征而 $\boldsymbol{X}_l$ 中选择了该特征，则无法判断该特征的重要性，因此引入粗糙超立方模型的先验知识来初始化该特征. 具体为：

$$x_{ij}=\begin{cases}1-\dfrac{rank(\gamma_{a_i}(D))}{m}, x_j^*=0 \text{ 且 } x_{lj}=1,\\ rand, x_j^*=0 \text{ 且 } x_{lj}=0,\\ (1-\sigma)\times rand+\sigma\times x_j^*, \text{其他}.\end{cases} \tag{30}$$

其中 $rank(\gamma_{a_i}(D))$ 为特征 a_i 在所有特征相关度降序排列后的序值；m 为所有特征的总数目；σ 为 [0, 1] 范围内的参数，其目的为控制 $\boldsymbol{X}^*$ 对粒子初始化的影响；$rand$ 为介于 [0, 1] 间的随机数.

4 实验与结果

本节通过实验结果分析，验证了本文提出方法的有效性和高效性. 为了分析 FERHFS 的性能表现，实验分为 3 个部分：第 1 部分讨论算法中提到的参数设置对算法性能的影响；第 2 部分讨论了联邦学习策略的有效性；第 3 部分通过与其他相关典型特征选择方法的实验比较，验证了 FERHFS 的高效性.

4.1 实验设置

数据集来自 UCI 机器学习库，表 2 描述了实验中使用的 12 组数据集的相关信息. 数据集被随机划分成 N 份并随机分配给 N 个本地设备，实验均采用 5 折交叉验证法，利用从 C4.5 和 SVM 分类器中获得的平均分类准确率（Mean-Acc）评估该算法的性能表现. 设置 FERHFS 算法学习的轮次 T=4，每个设备上特征选择算法的迭代次数 M=20，粒子规模 I=20，学习因子 c_1=c_2=2.

Table 2 Description of Datasets

表 2 数据集描述

序号	数据集	样本数	特征数	类别数
1	Ionosphere	351	34	2
2	Liver	345	6	2
3	Sonar	208	60	2
4	SpectfHeart	267	44	2
5	Vehicle	846	18	4
6	Wine	178	13	3
7	Glass	214	10	7
8	Texture	5 500	40	11
9	Segment	2 310	18	7
10	Isolet	7 797	617	26
11	Ring	7 400	20	2
12	LetterRecognition	20 000	16	26

4.2 粒子初始化参数 σ 的取值分析

粒子初始化参数 σ 是为了控制上一轮算法迭代中产生的全局最优子集向量 $\boldsymbol{X}^*$ 对当前粒子初始化的影响，通过分析不同参数值对 FERHFS 性能的影响，选择了最优参数值进行后续实验. 首先，$\sigma\in\{0.1,0.2,\cdots,0.9\}$，图 3 为不同 σ 值下 FERHFS 算法的平均分类准确率.

可以清楚地看到，在超过一半的数据集上，当

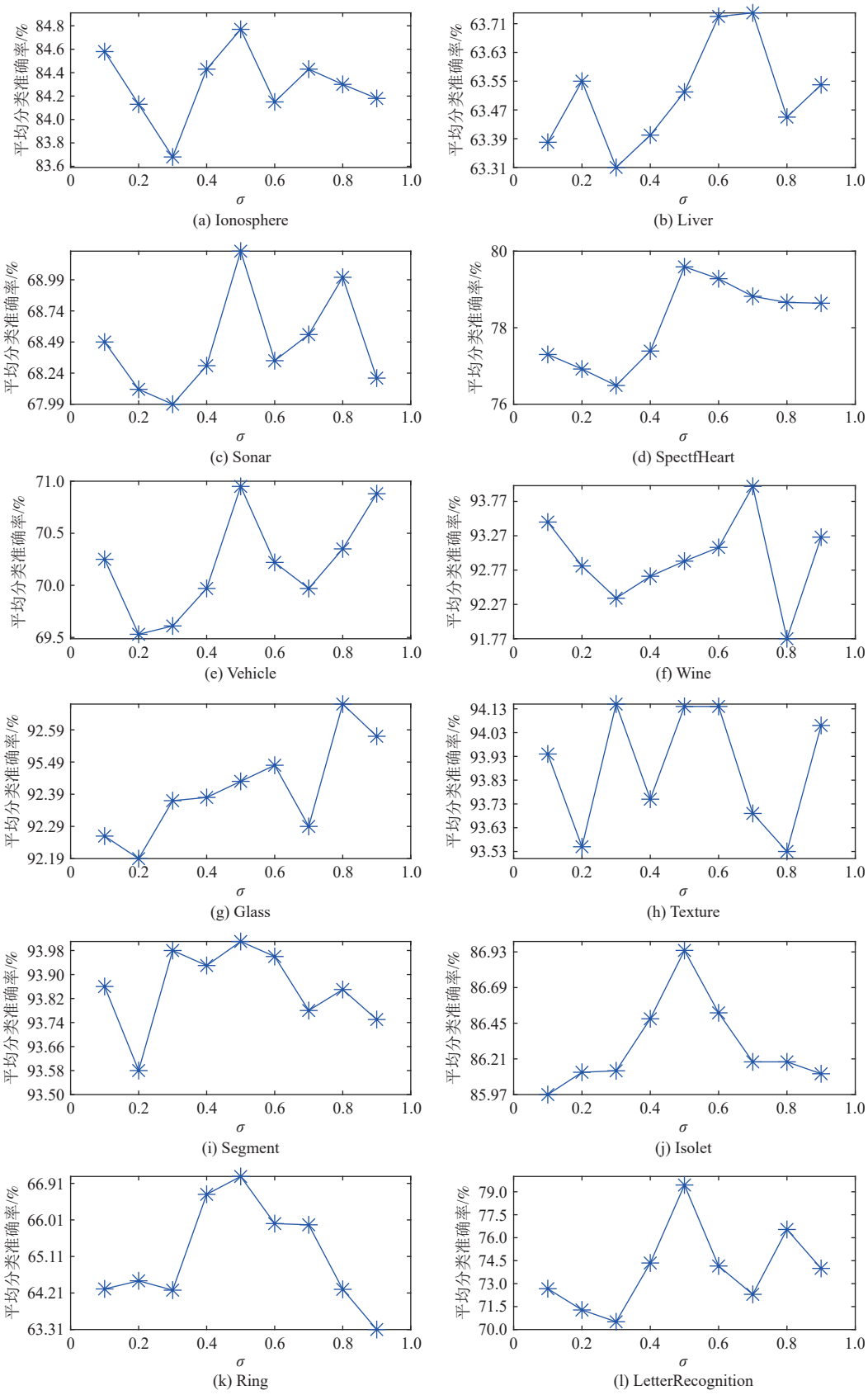


Fig. 3 Average classification accuracy of twelve datasets under different parameter σ values

图 3 不同参数 σ 值下 12 个数据集的平均分类准确率

σ=0.5 时，FERHFS 算法拥有最高的平均分类准确率，当$\sigma \in [0.4, 0.8]$时，所有数据集都获得了较高的平均分类准确率. 同时，可以观察到，当$\sigma \in [0, 0.3]$时，算法往往具有较低的平均分类准确率；当 σ 的取值接近于 1 时，算法的平均性能也较低. 通过式(29)分析可知，σ 是用来控制算法过程中随机性和由全局最优特征子集给出的学习经验对粒子初始化过程的影响. 在全局最优子集中被选择的特征，通常被认为更重要，应该增加其被选择的概率，在这种思路下，即认为将参数 σ 的值增大，需要通过初始化参数 σ 实现这个需求. 因此，当$\sigma \in [0, 0.3]$，即 σ 的取值过小时，粒子初始化过程依赖于随机性，忽略了从全局最优特征子集学习到的有效经验，在这种情况下，算法在大部分数据集下都表现出较差的性能. 当过度增大 σ 的值，特别是令参数 σ 的值接近于 1 时，粒子初始化过程缺少随机性，算法很容易陷入局部最优. 所以，应该将参数 σ 设置为一个合适的值，既能利用到全局最优特征子集带来的有效经验，也能通过初始化过程中的随机性避免算法陷入局部最优. 基于实验结果，在后续的实验任务中，参数 σ 的值被设置为 0.5.

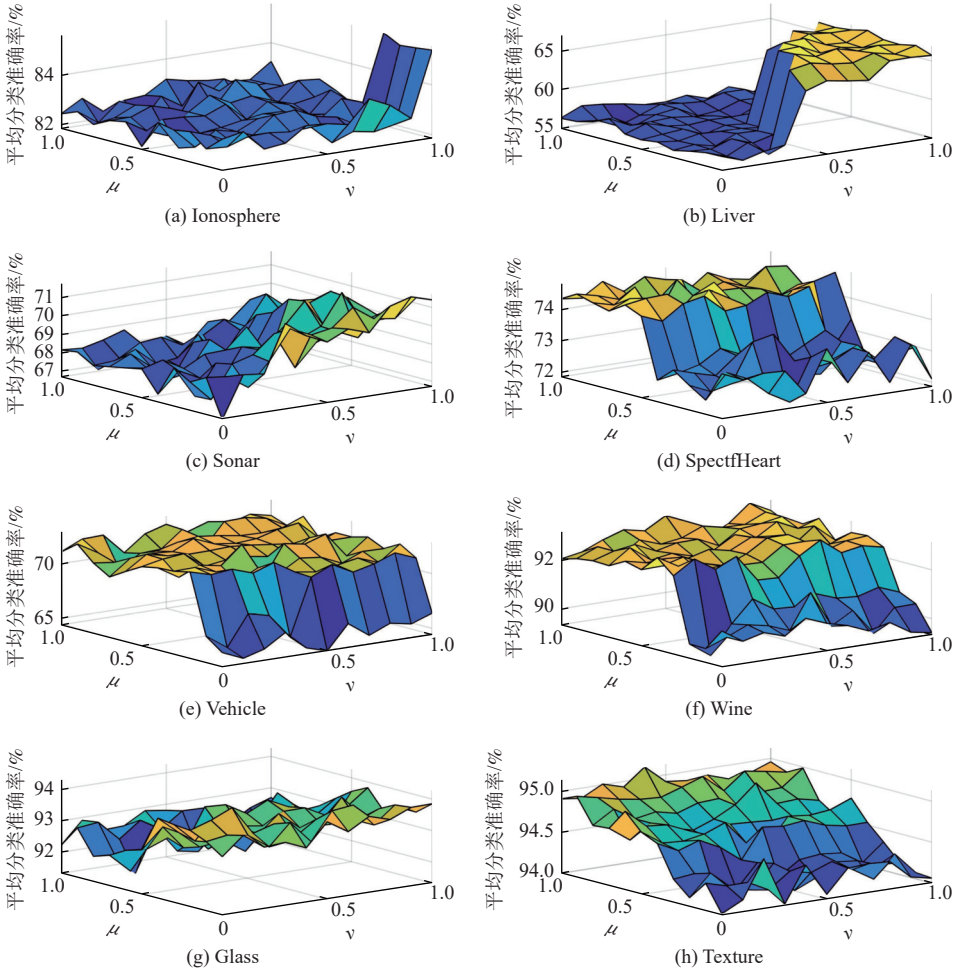
4.3 权重参数μ和ν的取值分析

μ和 ν 的值作为 BPSO 算法的适应度函数的权值参数，对算法的质量有很大的影响. 为了给权重参数 μ和 ν 选择最合适的值，将实验中μ和 ν 均设置为 0.0~1.0，步长为 0.1. 图 4 描述了在不同参数值组合下的平均分类准确率.

显然，从图 4 中可以看出，随着参数 ν 的增加和参数μ的减少，平均分类准确率总体上呈下降趋势. 可以观察到，当 FERHFS 算法得到最大的平均分类准确率时参数μ和 ν 的值并不相同，但当参数$\nu \in [0.5, 1]$，$\mu \in [0, 0.5]$时，大部分数据集有较好的平均分类准确率. 根据实验结果，在后续的实验中，将 12 个数据集上的参数$\{\mu, \nu\}$依次设置为 {0.3, 1}, {0.6, 0.3}, {0.4, 0.9}, {0.5, 0.6}, {0.4, 0.6}, {0.5, 0.6}, {0.4, 1}, {0.5, 1}, {0.5, 0.8}, {0.3, 0.9}, {0.3, 0.5}, {0.1, 0.5}.

4.4 本地设备参数数目 N 的取值分析

实验讨论了本地设备数目 N 对 FERHFS 算法性能的影响，本地设备的数量分别设置为 2, 3, 5. 图 5 说明了不同 N 值下，FERHFS 算法所获得的平均分类准确率. 如图 5 所示，对于参与实验的 12 组数据集中的大多



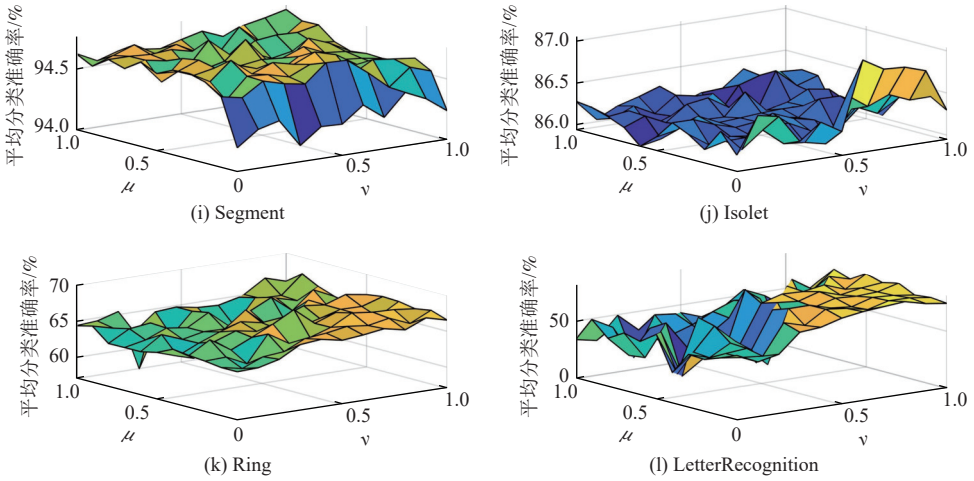


Fig. 4 Average classification accuracy of twelve datasets with the different combinations of parameters μ and v

图 4 不同参数组合$\{\mu, v\}$下 12 个数据集的平均分类准确率

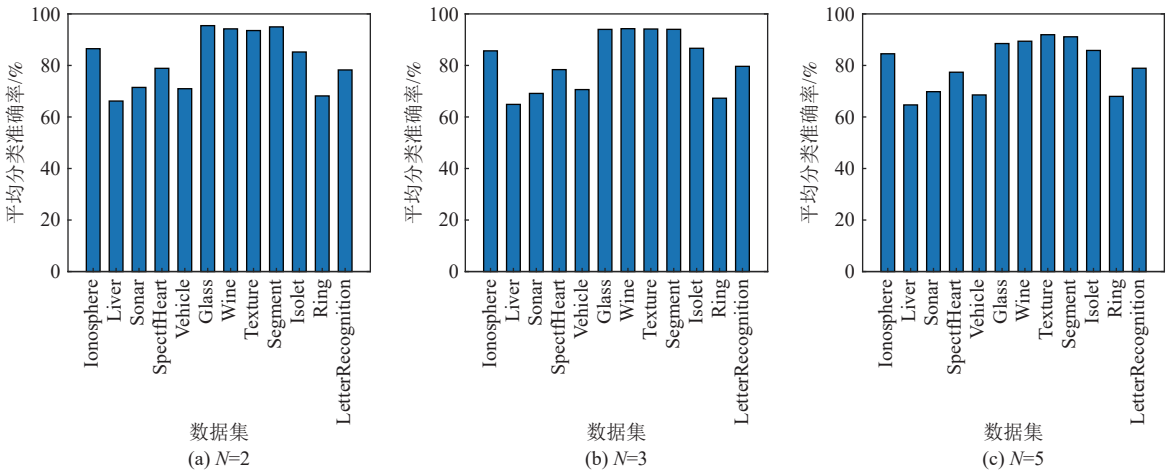


Fig. 5 Average classification accuracy of twelve datasets with different N values of local device

图 5 不同本地设备 N 值下 12 个数据集的平均分类准确率

数, 当 N=2 时 FERHFS 所获得的平均分类准确率最高, 当 N=5 时 FERHFS 所获得的平均分类精准确率最低.

从实验结果可知, 在大部分小规模数据集中, 当 N=2 时每个本地设备拥有的样本数量最高, 当 N=5 时每个本地设备拥有的样本数量最低. 出现这种实验现象的原因在于当本地设备数量 N 越大时, 在对数据集进行随机划分的过程中, 随机分配给每个本地设备的样本量就会随之减少. 对于一些总样本数量较少的数据集, 如 Glass 数据集和 Wine 数据集, 样本数量分别为 214 和 178, 当将其随机划分给 5 个本地设备时, 每个本地设备分配到的样本数量相比于 N=2 时会大幅度下降. 一般来说, 当客户端都有足够数量的样本时, 算法构建的模型性能更好, 当参与训练的样本数量不足时, 每个本地设备难以构建准确的特征选择模型, 算法的性能表现就会相对较差. 因此, 对于这 2 个数据集, 当 N=5 时特征子集的平均分类准确率与 N=2 时特征子集的平均分类准确率相差较大. 同时, 对于规模较大的数据集, 如 Ring 数据集和 LetterRecognition 数据集, 样本数量分别为 7 400 和 20 000, 当 N 分别设置为 2, 3, 5 时, 每个参与训练的本地设备都可以分配到足够模型训练的样本量. 表现在实验结果上, 即本地设备数量 N 的设置对于算法的性能影响较小. 以 LetterRecognition 数据集为例, N 分别设置为 2, 3, 5 时, 算法所获得的平均分类准确率分别为 78.252 5%, 79.644 9%, 78.912 5%, 当每个本地设备拥有足够数量的样本后, 继续增加样本量并不会提高算法性能.

4.5 联邦策略的分析

为了验证联邦学习框架的有效性, 将 FERHFS 和不含联邦策略的 FERHFS\FL 进行比较, FERHFS\FL

表示每个本地设备独立执行 FERHFS 算法得到本地最优子集，然后直接输出平均分类准确率. 将本地设备的数量设置为 3. 表 3 列出了每个本地设备的分类准确率和总体平均分类准确率. 从表 3 中发现，对于所有数据集，在联邦学习策略的帮助下，FERHFS 的分类准确率明显高于 FERHFS\FL. 以 Sonar 和 SpectfHeart 数据集为例，FERHFS 所挑选出的全局最优特征子集在分类器上的平均分类准确率分别为 70.377 3% 和 78.545 4%，而 FERHFS\FL 的平均分类准确率则分别为 65.405 4% 和 74.588 2%，对比 2 种情况下 3 个本地设备的平均分类准确率，可以明显地看出在联邦策略的帮助下，所挑选出的全局最优特征子集在平均分类准确率有更好的表现，对于其他数据集也有类似的结果. 同时，以数据集 Liver 为例，对于 3 个本地设备，FERHFS 的分类准确率分别为 62.724 6%，70.376 8%，61.014 4%；FERHFS\FL 的分类准确率分别为 55.507 2%，61.826 1%，58.724 6%. 同理，从实验结果对比可知，所挑选出的全局最优特征子集在每个本地设备下都具有更好的表现，对于其他数据集也有类似的结果.

在 FERHFS 算法中，设计了联邦环境下的粒子初始化策略和特征子集评估策略，每个参与方在本地训练各自独有的特征选择模型，还可以通过所提出联邦框架的中央服务器端交换学习有效信息. 由于这种独特的联邦策略，使算法具有良好的协同训练能力，不仅能够利用本地私有的数据集进行学习，而且充分吸取了共同参与方所给出的有效学习经验，使最终选择的全局最优特征子集适应于所有参与方的模型需求. 由于 FERHFS\F 算法不含联邦策略，在训练过程中只能利用本地私有数据集孤立地进行学习，选择的最优特征子集没有充分全面考虑到每个本地设备的性能需求，表现在实验结果上即为平均分类准确率远远低于 FERHFS 的平均分类准确率. 并且，对于 FERHFS 算法，联邦策略在提高协同训练能力的同时，没有忽略数据隐私保护这个前提，在联邦训练的全过程中，保证了本地设备数据集的隐私性. 总的来说，本文提出的联邦学习策略不仅解决了多设备下特征选择任务和用户隐私保护的问题，而且提高了算法在数据隐私保护条件下的性能.

4.6 与现有特征选择算法的比较

本节比较了 FERHFS 算法与现有 6 种特征选择算法在满足隐私保护条件下特征选择结果上的差异. 这 6 种算法包括典型的过滤式集成特征选择算法(FAFS)[29]、典型的封装式特征选择算法(SBS)[32]、一种新颖的乌鸦搜索特征选择算法(NOCSA)[33]、基于离散粒子群优化的特征选择算法(BPSO)[34]、结合粗糙超立方模型和粒子群优化算法的(RHBPSO)[35]算法和一种新颖的基于粒子群优化的联邦特征选择算法(FPSO)[29]. 从表 4 的结果来看，对于所有数据集，FERHFS 算法的平均分类准确率都优于其他 6 种特征选择算法. 特别是对于 Sonar, Vehicle, Wine 数据集，FERHFS 算法的效果更为明显. 例如，对于 Sonar 数据集，FERHFS 的平均分类准确率为 70.377 3%，其他 6 种算法所得到的值分别为 57.507 9%，65.706 9%，58.965 8%，65.909 6%，65.711 8%，64.811 9%；对于 Wine 数据集，FERHFS 的平均分类准确率为 94.688 5%，其

Table 3 Average Classification Accuracy Obtained by FERHFS and FERHFS\FL

表 3 FERHFS 和 FERHFS\FL 得到的平均分类准确率 %

数据集	FERHFS\FL 得到的平均分类准确率				FERHFS 得到的平均分类准确率			
	F_1	F_2	F_3	平均值	F_1	F_2	F_3	平均值
Ionosphere	84.810 4	83.751 2	79.042 2	82.534 6	86.013 3	86.066 4	85.068 8	85.716 2
Liver	55.507 2	61.826 1	58.724 6	58.685 9	62.724 6	70.376 8	61.014 4	64.705 3
Sonar	66.725 3	61.109 8	68.380 9	65.405 4	71.717 9	66.747 3	72.666 7	70.377 3
SpectfHeart	78.350 7	75.884 5	69.529 4	74.588 2	83.106 7	76.949 9	75.579 5	78.545 4
Vehicle	63.389 5	66.027 8	67.008 7	65.475 3	69.664 3	71.691 7	74.031 1	71.795 7
Wine	90.454 5	94.191 9	80.277 6	88.308 1	95.469 7	95.818 2	92.777 8	94.688 6
Glass	87.428 6	89.847 6	88.342 8	88.539 7	90.641 2	94.692 1	94.104 8	93.146 0
Texture	87.823 4	88.685 2	88.464 2	88.324 3	93.728 0	94.384 4	94.093 2	94.068 5
Segment	93.290 0	91.857 1	91.987 0	92.378 1	95.000 0	94.372 3	94.090 9	94.487 7
Isolet	86.086 8	85.630 4	85.081 6	85.599 6	87.264 3	86.802 4	85.917 7	86.661 5
Ring	63.847 5	63.463 5	64.189 2	63.833 4	69.491 1	65.018 6	67.328 6	67.279 5
LetterRecognition	60.596 1	61.366 1	60.622 6	60.861 6	79.322 8	80.153 1	79.459 1	79.645 0

Table 4 Average Classification Accuracy Obtained from FERHFS and Comparative Algorithms

表 4 FERHFS 和对比算法得到的平均分类准确率 %

数据集	FAFS	SBS	NOCSA	BPSO	RHBPSO	FPSO	FERHFS (本文)
Ionosphere	82.860 3	76.126 8	81.292 2	82.785 8	83.061 6	82.468 2	85.716 2
Liver	57.120 8	58.637 6	56.425 1	59.169 1	60.038 6	60.531 4	64.705 3
Sonar	58.965 8	57.507 9	65.909 6	65.706 9	65.711 8	64.811 9	70.377 2
SpectfHeart	74.239 6	76.527 9	74.647 7	73.300 7	74.080 6	72.472 7	78.545 3
Vehicle	56.258 7	59.393 3	63.587 4	65.383 2	66.825 4	67.421 1	71.795 7
Wine	88.355 2	82.291 2	89.107 7	88.774 4	89.575 8	86.260 9	94.688 5
Glass	91.651 8	79.622 2	91.862 4	87.821 1	91.772 5	91.790 5	93.146 0
Texture	82.239 8	91.387 7	88.866 8	92.794 1	92.897 2	92.806 6	94.068 5
Segment	84.555 6	88.370 8	88.802 3	91.555 6	90.461 7	92.525 3	94.487 7
Isolet	64.108 5	55.479 5	84.558 9	85.699 5	83.815 2	84.759 4	86.943 5
Ring	63.086 1	61.452 7	56.741 9	62.455 2	61.374 5	63.319 3	67.279 5
LetterRecognition	61.511 2	70.939 7	64.726 5	74.801 6	75.229 2	73.297 4	79.645 0

他 6 种算法的值分别为 88.355 2%, 82.291 2%, 89.107 7%, 88.774 4%, 89.575 8%, 86.260 9%; 对于其他所有数据集, 情况也类似. 此外, 相比于其他 6 种特征选择算法, FERHFS 算法的平均分类准确率分别提高了 8.629 4 个百分点、4.471 1 个百分点、7.920 3 个百分点、5.225 4 个百分点、3.678 4 个百分点、4.049 1 个百分点.

出现这样的实验结果的原因如下: 首先, 对 SBS 和 FAFS 算法, 由于这 2 种算法缺少全局搜索机制, 因此很容易陷入局部最优, 在这种情况下所选择出的特征子集往往不具备很好的性能表现. 其次, 对于 BPSO, NOCSA, FERHFS 这 3 种算法, 在算法训练过程中, 本地设备间没有设置有效信息交换策略, 在各个本地设备上, 算法只能利用本地设备所拥有的数据集孤立地进行模型训练, 这样显然是不利于获取全局最优特征子集. 同时, 由于隐私保护, 每个本地设备的样本量较小, 这容易出现由“高维小样本”数据引起的过拟合问题. 最后, FPSO 算法虽然在模型训练的过程中设置了有效信息共享机制, 但在搜索最佳特征子集时缺乏有效的先验知识和更好的适应度函数, 这也使得其性能比 FERHFS 的更差. 综上所述, FERHFS 算法不仅满足了数据隐私保护需求, 并且在联邦学习架构的基础上所提出的粒子初始化策略和特征子集评估策略有效提高了性能, 对比近年来所提出的典型相关特征选择算法, 经 FERHFS 得到的特征子集具有更高的分类质量.

5 总 结

由于隐私保护的需求, 每个客户端参与训练的样本量较少, 传统的特征选择方法无法适应这类情况. 为了解决多方参与下的特征选择问题, 本文提出了一种集中式(客户端/服务器)的联邦特征选择架构. 基于该架构, 结合粗糙超立方模型和改进后的离散粒子群算法完成特征选择任务, 同时, 设计了粒子初始化策略和基于中央服务器的特征子集评估策略共享有效信息以指导搜索全局最优特征子集. 在 9 个 UCI 基准数据集上的实验结果表明, 相比于其他 6 种传统特征选择算法(FAFS, SBS, NOCSA, BPSO, RHBPSO, FPSO), 在满足各参与方数据隐私保护前提下, FERHFS 所选特征子集在各参与方上具有更高的分类性能. 此外, 实验结果表明, 基于本文所提出的联邦策略, 各个参与方上的搜索性能被显著提高.

作者贡献声明: 陈雪颖完成了算法思路设计、实验验证和论文撰写; 罗川负责理论分析与方法建模以及论文撰写; 李天瑞提出指导意见以及审阅与修订论文; 陈红梅指导方案设计以及审阅与修订论文.

参 考 文 献

[1] Ren Jiadong, Zhang Yafei, Zhang Bing, et al. Classification method of industrial Internet intrusion detection based on feature selection[J]. Journal of Computer Research and Development, 2022, 59(5): 1148−1159 (in Chinese)
(任家东, 张亚飞, 张炳, 等. 基于特征选择的工业互联网入侵检测分类方法[J]. 计算机研究与发展, 2022, 59(5): 1148−1159)

[2] Ashokkumar P, Shankar S G, Srivastava G, et al. A two-stage text feature selection algorithm for improving text classification[J]. ACM Transactions on Asian and Low-Resource Language Information

Processing, 2021, 20(3): 1−19

[3] Ji Xiaonan, Shen Hanwei, Ritter A, et al. Visual exploration of neural document embedding in information retrieval: Semantics and feature selection[J]. IEEE Transactions on Visualization and Computer Graphics, 2019, 25(6): 2181−2192

[4] Zhang Xiaohong, Li Mengyuan, Shao Songtao, et al. (I, O)-fuzzy rough sets based on overlap functions with their applications to feature selection and image edge extraction[J]. IEEE Transactions on Fuzzy Systems, 2023, 32(4): 1796−1809

[5] Yuan Zhong, Chen Hongmei, Wang Zhihong, et al. Exploiting hybrid kernel-based fuzzy complementary mutual information for selecting features[J]. Journal of Computer Research and Development, 2023, 60(5): 1111−1120 (in Chinese)
(袁钟, 陈红梅, 王志红, 等. 利用混杂核模糊补互信息选择特征[J]. 计算机研究与发展, 2023, 60(5): 1111−1120)

[6] Pawlak Z. Rough sets[J]. International Journal of Computer & Information Sciences, 1982, 11(5): 341−356

[7] Jensen R, Shen Qiang. Fuzzy-rough sets assisted attribute selection [J]. IEEE Transactions on Fuzzy Systems, 2007, 15(1): 73−89

[8] Duan Jie, Hu Qinghua, Zhang Lingjun, et al. Feature selection for multi-label classification based on neighborhood rough sets[J]. Journal of Computer Research and Development, 2015, 52(1): 56−65 (in Chinese)
(段洁, 胡清华, 张灵均, 等. 基于邻域粗糙集的多标记分类特征选择算法[J]. 计算机研究与发展, 2015, 52(1): 56−65)

[9] Maji P. A rough hypercuboid approach for feature selection in approximation spaces[J]. IEEE Transactions on Knowledge and Data Engineering, 2012, 26(1): 16−29

[10] Wang Changzhong, Qi Yali, Shao Mingwen, et al. A fitting model for feature selection with fuzzy rough sets[J]. IEEE Transactions on Fuzzy Systems, 2016, 25(4): 741−753

[11] Shami T M, El-Saleh A A, Alswaitti M, et al. Particle swarm optimization: A comprehensive survey[J]. IEEE Access, 2022, 10: 10031−10061

[12] Kabir M M, Shahjahan M, Murase K. A new hybrid ant colony optimization algorithm for feature selection[J]. Expert Systems with Applications, 2012, 39(3): 3747−3763

[13] Oh I S, Lee J S, Moon B R. Hybrid genetic algorithms for feature selection[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2004, 26(11): 1424−1437

[14] Kennedy J, Eberhart R. Particle swarm optimization[C] //Proc of IEEE Int Conf on Neural Networks. Piscataway, NJ: IEEE, 1995: 1942−1948

[15] Mistry K, Zhang Li, Neoh S C, et al. A micro-GA embedded PSO feature selection approach to intelligent facial emotion recognition[J]. IEEE Transactions on Cybernetics, 2016, 47(6): 1496−1509

[16] Kairouz P, McMahan H B, Avent B, et al. Advances and open problems in federated learning[J]. Foundations and Trends in Machine Learning, 2021, 14(1/2): 1−210

[17] Li Jundong, Cheng Kewei, Wang Suhang, et al. Feature selection: A data perspective[J]. ACM Computing Surveys, 2017, 50(6): 1−45

[18] Wei Wei, Ouyang Dantong, Lü Shuai, et al. An approach combining incremental search and heuristic search for solving multiobjective problems[J]. Journal of Computer Research and Development, 2010, 47(11): 1954−1961 (in Chinese)
(魏唯, 欧阳丹彤, 吕帅, 等. 结合增量与启发式搜索的多目标问题处理方法[J]. 计算机研究与发展, 2010, 47(11): 1954−1961)

[19] Xu Zhaozhao, Shen Derong, Nie Tiezheng, et al. Hybrid feature selection algorithm combining information gain ratio and genetic algorithm[J]. Journal of Software, 2022, 33(3): 1128−1140 (in Chinese)
(许召召, 申德荣, 聂铁铮, 等. 融合信息增益比和遗传算法的混合式特征选择算法[J]. 软件学报, 2022, 33(3): 1128−1140)

[20] Chen Ke, Xue Bing, Zhang Mengjie, et al. Evolutionary multitasking for feature selection in high-dimensional classification via particle swarm optimization[J]. IEEE Transactions on Evolutionary Computation, 2021, 26(3): 446−460

[21] Shafqat W, Malik S, Lee K T, et al. PSO based optimized ensemble learning and feature selection approach for efficient energy forecast[J]. Electronics, 2021, 10(18): 2188

[22] Song Xianfang, Zhang Yong, Gong Dunwei, et al. A fast hybrid feature selection based on correlation-guided clustering and particle swarm optimization for high-dimensional data[J]. IEEE Transactions on Cybernetics, 2021, 52(9): 9573−9586

[23] Kumar S U, Inbarani H H. PSO-based feature selection and neighborhood rough set-based classification for BCI multiclass motor imagery task[J]. Neural Computing and Applications, 2017, 28(11): 3239−3258

[24] Huda R K, Banka H. Efficient feature selection methods using PSO with fuzzy rough set as fitness function[J]. Soft Computing, 2022, 26(5): 2501−2521

[25] Zhou Wei, Wang Chao, Xu Jian, et al. Privacy-preserving and decentralized federated learning model based on the blockchain[J]. Journal of Computer Research and Development, 2022, 59(11): 2423−2436 (in Chinese)
(周炜, 王超, 徐剑, 等. 基于区块链的隐私保护去中心化联邦学习模型[J]. 计算机研究与发展, 2022, 59(11): 2423−2436)

[26] Qin Yang, Kondo M. Federated learning-based network intrusion detection with a feature selection approach[C/OL] //Proc of the 3rd Annual Int Conf on Electrical, Communication, and Computer Engineering. Piscataway, NJ: IEEE, 2021[2024-04-16]. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=-&arnumber=9514222

[27] Feng Siwei, Yu Han, Zhu Yuebing. MMVFL: A simple vertical federated learning framework for multi-class multi-participant scenarios[J]. Sensors, 2024, 24(2): 619

[28] Zhang Xunzheng, Mavromatis A, Vafeas A, et al. Federated feature selection for horizontal federated learning in IoT networks[J]. IEEE Internet of Things Journal, 2023, 10(11): 10095−10112

[29] Hu Ying, Zhang Yong, Gong Dunwei, et al. Multi-participant federated feature selection algorithm with particle swarm optimizaiton for imbalanced data under privacy protection[J]. IEEE Transactions on Artificial Intelligence, 2022, 4(5): 1002−1016

[30] Rieke N, Hancox J, Li W, et al. The future of digital health with federated learning[J]. NPJ Digital Medicine, 2020, 3(1): 1−7

[31] Li Qinbin, He Bingsheng, Song Dawn. Model-contrastive federated learning[C] //Proc of the 39th Annual IEEE/CVF Conf on Computer Vision and Pattern Recognition (CVPR). Piscataway, NJ: IEEE, 2021: 10708−10717

[32] Haq A U, Li Jianping, Memon M H, et al. Heart disease prediction system using model of machine learning and sequential backward

selection algorithm for features selection[C/OL] //Proc of the 5th Annual IEEE Int Conf for Convergence in Technology. Piscataway, NJ: IEEE, 2019[2024-04-20]. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9033683

[33] Sayed G I, Hassanien A E, Azar A T. Feature selection via a novel chaotic crow search algorithm[J]. Neural Computing and Applications, 2019, 31(1): 171−188

[34] Chantar H K, Corne D W. Feature subset selection for Arabic document categorization using BPSO-KNN[C] //Proc of the 3rd Annual World Congress on Nature and Biologically Inspired Computing. Piscataway, NJ: IEEE, 2011: 546−551

[35] Luo Chuan, Wang Sizhao, Li Tianrui et al. Large-scale meta-heuristic feature selection based on BPSO assisted rough hypercuboid approach[J]. IEEE Transactions on Neural Networks and Learning Systems, 2023, 34(12): 10889−10903



Chen Xueying, born in 2000. Master candidate. Student member of CCF. Her main research interests include feature selection and federated learning.

陈雪颖, 2000 年生. 硕士研究生. CCF 学生会员. 主要研究方向为特征选择、联邦学习.



Luo Chuan, born in 1987. PhD, associate professor, PhD supervisor. Member of CCF. His main research interests include data mining and granular computing.

罗 川, 1987 年生. 博士, 副教授, 博士生导师. CCF 会员. 主要研究方向为数据挖掘、粒计算.



Li Tianrui, born in 1969. PhD, professor, PhD supervisor. His main research interests include data mining and granular computing.

李天瑞, 1969 年生. 博士, 教授, 博士生导师. 主要研究方向为数据挖掘、粒计算.



Chen Hongmei, born in 1971. PhD, professor, PhD supervisor. Her main research interests include data mining and granular computing.

陈红梅, 1971 年生. 博士, 教授, 博士生导师. 主要研究方向为数据挖掘、粒计算.