doi:10.11887/j.cn.202406011 http://journal.nudt.edu.cn

面向混合特征数据的粒子群填补方法

刘 艺,秦 伟,李庚松,刘 坤,王 强,郑奇斌*,任小广
(军事科学院,北京 100091)

摘 要:针对传统数据填补方法难以有效利用标签信息和缺失数据的随机信息的不足,提出面向混合型特征的粒子群优化填补算法。将连续型特征取值建模为高斯分布,均值和标准差作为优化参数。将离散型特征的取值概率作为参数进行优化。使用分类正确率作为优化目标,充分利用标签信息和缺失数据的随机信息。采用4种基于统计的方法和2种基于演化算法的填补方法作为对比,在6个典型的分类数据集上进行实验。结果表明,提出的方法在分类正确率指标上显著优于其他对比算法,同时具有较优的时间开销,能够有效解决混合特征数据缺失的问题。

关键词:缺失数据;数据填补;粒子群优化;混合特征;分类
中图分类号:TP391 **文献标志码**:A **文章编号**:1001-2486(2024)06-107-06

论文拓展

Particle swarm optimization based data imputation method for mixed features

LIU Yi, QIN Wei, LI Gengsong, LIU Kun, WANG Qiang, ZHENG Qibin, REN Xiaoguang*
(Academy of Military Sciences, Beijing 100091, China)

Abstract: Aiming at the deficiency of traditional data imputation methods in effectively using the label information and random characteristics of missing data, a particle swarm optimization based imputation method for mixed features was proposed. The value of continuous feature was modeled as Gaussian distribution, and the mean and standard deviation were used as optimization parameters. The value probability of categorical features was optimized as a parameter. The classification accuracy rate was used as the optimization target to make full use of random information of label information and missing data. Four statistical methods and two evolutionary algorithm based imputation methods were used to compare the results on six typical classification datasets. The results show that the proposed method significantly outperforms other comparison algorithms in terms of classification accuracy indicator, and has better time overhead at the same time, which can effectively solve the data missing problems of mixed features.

Keywords: missing data; data imputation; particle swarm optimization; mixed features; classification

软件系统和应用程序中经常面临特征数据缺失的情况[1],如物联网数据、医疗数据、材料数据等[2-4],数据缺失可能导致学习算法或程序性能下降甚至不可用。导致数据缺失的原因较多,如调研项目无回应、意外丢失或传输错误等[5-7]。为了解决数据缺失问题,研究人员提出了一些有效的填补方法,按照采用技术的不同,可以分为基于统计学的方法和基于学习的方法[8]。基于统计学的方法利用已有数据的距离、频率、均值等信息,填补缺失数据;基于学习的方法通过训练学习模型,预测并填补缺失数据。

然而,传统方法的缺点导致其性能难以得到进一步的提升[9]:基于统计学的典型方法包括均值填充、期望最大填充、多重填补等,它们通常采用已有实例的统计信息进行填补,无法有效利用缺失数据的实例标签信息[10];基于学习的典型方法包括基于支持向量机的方法、基于决策树的方法以及基于聚类的方法等,它们通常训练特定的学习模型参数,当参数确定时,会输出确定的填补值,未能在一定程度上考虑缺失数据的随

收稿日期:2022-07-15
基金项目:国家自然科学基金资助项目(91948303);国家自然科学基金青年科学基金资助项目(61802426)
第一作者:刘艺(1990—),男,安徽蚌埠人,助理研究员,博士,E-mail:albertliu20th@163.com
*通信作者:郑奇斌(1990—),男,甘肃兰州人,助理研究员,博士,E-mail:zhengqibin1990@163.com
引用格式:刘艺,秦伟,李庚松,等. 面向混合特征数据的粒子群填补方法[J]. 国防科技大学学报,2024,46(6):107-112.
Citation: LIU Y, QIN W, LI G S, et al. Particle swarm optimization based data imputation method for mixed features[J]. Journal of National University of Defense Technology, 2024, 46(6): 107-112.

机性[11]。

粒子群优化(particle swarm optimization, PSO)是一种寻优能力较强的演化算法,它具有参数设置方便、易于适配、部署实现简单等特点,在众多领域得到了广泛的应用[12-14]。

为了提升数据缺失情况下分类系统的性能,提出针对混合变量数据缺失问题的粒子群数据填补(data imputation based on particle swarm optimization, DIPSO)算法,将发生数据缺失的特征分为连续型特征和离散型特征;针对连续型特征,将其取值建模为高斯分布,均值和标准差作为参数;针对离散型特征,将不同取值的概率作为参数;采用粒子群优化算法对这些参数进行优化,通过高斯分布和概率值生成具有一定随机性的填补数据,充分利用标签信息和缺失数据的随机性,提升分类系统的精度和可用性。

1 针对混合特征的粒子群数据填补方法

DIPSO 的算法流程如图 1 所示。

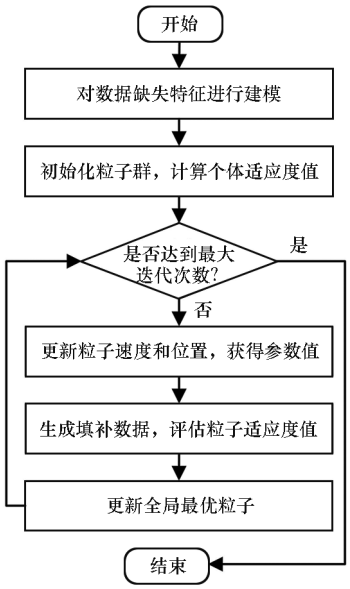

图 1 DIPSO 流程图

Fig. 1 Flowchart of DIPSO

首先对数据集实例中发生数据缺失的特征进行记录,并根据数据源及已有数据的分布统计情况,通过人工的方式将其划分为连续型或离散型:若已有数据不重复且呈现出可通过曲线拟合的情况,则可以将其判定为连续型特征;若已有数据仅在几个值之间重复,可以将其判定为离散型特征。针对连续型特征,将其取值建模为高斯分布,均值和标准差作为参数;针对离散型特征,确定其不重复的值,将这些值出现的概率作为参数,确定参数的个数和对应的值域。

然后根据参数的个数(即粒子的维度)和对应的值域,初始化粒子群,通过粒子初始值与对应特征的填补方法,填补数据集并进行分类测试,采用分类指标计算粒子个体的适应度值。

进入粒子群优化迭代过程后,更新粒子速度,并移动粒子的位置,获得参数值;根据参数值和对应特征类型,采用相应的方法生成数据填补数据集,通过完整的数据集进行分类,使用分类指标评估当前粒子的适应度值;根据粒子的适应度值更新全局最优粒子;当达到最大循环次数后,停止迭代过程,输出最优粒子个体,即最优参数取值。

下面详细介绍连续型和离散型特征的填补方法,描述粒子群优化算法的主要步骤,给出 DIPSO 算法的伪代码,分析其时间复杂度。

1.1 连续型特征填补方法

众多实际系统和应用产生的数据均近似地服从高斯分布[15]。因此,针对连续型特征缺失数据的情况,将其取值的分布建模为高斯分布。假定数据集在连续型特征 i 上缺失了 C_i 个数据,可以将特征 i 的取值概率密度函数建模为:

$$f(x_i)=\frac{1}{\delta_i\sqrt{2\pi}}\mathrm{e}^{-\frac{1}{2}\left(\frac{x-\mu_i}{\delta_i}\right)^2} \tag{1}$$

式中,μ_i 为均值,δ_i 为标准差。显然,当均值和标准差变化时,特征 i 的取值分布会发生显著的改变。DIPSO 将 μ_i 和 δ_i 作为优化参数编码成粒子,通过迭代产生较好的取值,根据 μ_i 和 δ_i 的值生成 C_i 个数据,将这些数据填补至数据集中。在优化迭代的过程中,需要设置优化参数的值域范围,即特征 i 对应的 μ_i 和 δ_i 在粒子中的上下界,值域范围的设置可以通过特征 i 在数据集中出现的值来确定,采用特征 i 出现的最大值和最小值作为 μ_i 和 δ_i 的上界和下界。

1.2 离散型特征填补方法

针对离散型特征缺失数据的情况,假设在特征 j 上缺失了 D_j 个值,统计该特征不重复的取值,将这些值出现的概率作为参数进行编码并优化。基于优化产生的概率值,经过归一化后,采用轮盘赌策略生成 D_j 个值进行填补。显然,每个参数的阈值范围为[0,1]。

下面分析参数的个数,即粒子的维度。假设 C 为发生数据缺失的连续型特征个数,D 为发生数据缺失的离散型特征个数,则待优化的参数个数为 $2C+\sum_{p=1}^{D}D_p$,其中 D_p 为在特征 p 上出现的不重复值个数。

1.3　粒子群优化算法

PSO 是一种受鸟类群体觅食行为启发的演化算法,能够从个体和群体信息中进行学习,其主要步骤包括粒子速度的更新以及位置的移动[16]。在 t 时刻,粒子个体 i 的速度更新为:

$$\boldsymbol{v}_i^t = w\boldsymbol{v}_i^{t-1} + \alpha r_1(\boldsymbol{g}^* - \boldsymbol{x}_i^{t-1}) + \beta r_2(\boldsymbol{x}_i^* - \boldsymbol{x}_i^{t-1}) \tag{2}$$

式中,$\boldsymbol{v}_i^{t-1}$ 表示粒子 i 在 $t-1$ 时刻的速度;w 为惯性权重,控制粒子在空间中的搜索范围;α 和 β 是加速因子,控制粒子在群体和个体信息上的学习程度;$\boldsymbol{g}^*$ 表示全局最优解,$\boldsymbol{x}_i^*$ 是粒子 i 的历史最优解;$\boldsymbol{x}_i^{t-1}$ 是粒子 i 在 $t-1$ 时刻的位置;r_1 和 r_2 是满足均匀分布的随机数,取值范围为(0,1)。

粒子速度更新后,需要通过新的速度移动粒子的位置,即搜索新的解,粒子的位置移动方法为:

$$\boldsymbol{x}_i^t = \boldsymbol{x}_i^{t-1} + \boldsymbol{v}_i^t \tag{3}$$

1.4　算法整体流程与复杂度分析

综上,DIPSO 算法的伪代码如算法 1 所示。

算法 1　DIPSO 伪代码

Alg. 1　Pseudo-code of DIPSO

```
1. begin
2. 初始化个体,根据优化目标计算适应度值
3. while 未达到最大迭代次数
4.    for 每个粒子
5.        通过公式更新速度
6.        通过公式当前速度移动到新的位置
7.        基于粒子产生的参数,通过高斯分布和轮盘
      赌策略产生填补值填补数据
8.        通过完整数据和优化目标评估分类性能
9.        更新个体和全局最优解
10.   end for
11. end while
12. end
```

现对 DIPSO 算法的时间复杂度做分析。假设 T 为算法的最大迭代次数,N 为粒子的个数,L 为粒子的维度,填补算法的复杂度为 $O(L)$,算法的时间复杂度为 $O(T\times L\times N)$。

2　实验与分析

本节通过实验对 DIPSO 算法的性能进行比较分析。为了综合评估方法的性能,实验使用不同特征类型的 6 个完整数据集,在实验过程中通过控制数据集的缺失率来测试算法的填补能力。数据集来源于 UCI 机器学习网站(https://archive.ics.uci.edu/ml/datasets),数据集的有关基本信息如表 1 所示。BCC 是乳腺癌数据集,包括 116 个实例和 9 个连续型特征,预测类别为是否患癌;Park 是帕金森病患者的语音数据集,包括 197 个实例和 22 个连续型特征,预测类别为是否患有帕金森病;Lymp 是淋巴数据集,包括 148 个实例、18 个离散型特征和 4 个预测类别;MONK 是僧侣问题数据集,包括 432 个实例、7 个离散型特征和 2 个预测类别;Zoo 是动物数据集,包括 101 个实例、17 个混合型特征和 7 个预测类别;StH 是心脏病数据集,包括 270 个实例、13 个混合型特征和 2 个预测类别。

表 1　实验数据集属性

Tab. 1　Characteristics of experiment datasets

数据集	实例	特征	类别	类型
BCC	116	9	2	连续
Park	197	22	2	连续
Lymp	148	18	4	离散
MONK	432	7	2	离散
Zoo	101	17	7	混合
StH	270	13	2	混合

为了说明方法的有效性和优越性,选择 4 种广泛使用的统计学填补算法和 2 种基于学习的填补算法进行对比。4 种统计学填补方法采用实际中应用广泛的均值填补(mean imputation method,MI)算法[17]、K 近邻(K nearest neighbor,KNN)算法[18]、正则期望最大化(regularized expectation maximization,REM)算法[19]以及链式方程多元回归(multivariate imputation by chained equations,MICE)算法[20]。2 种基于学习的算法包括近年来提出的性能优异的极限学习机 - 粒子群优化 - 模糊 C 均值(extreme learning machine-particle swarm optimization-fuzzy C means,EPF)算法[21]以及多目标遗传数据填补算法(multi-objective genetic algorithm for data imputation,MOGAImp)[22],EPF 使用粒子群算法优化模糊 C 均值填补方法的参数,MOGAImp 通过多目标遗传方法优化填补值出现的位置。

采用 Ubuntu20.04LTS 操作系统、Intel Xeon E5 - 2630v4@2.2 GHz 处理器、64 GB 内存的实验平台。KNN 算法的 K 值设置为 5;REM 算法采用文献[23]提供的开源代码,参数为默认值;

MICE 算法使用 R 语言的开源软件包,参数为默认值[24];EPF 算法的迭代次数设置为 250,种群个数为 100,其他参数与文献[17]设置一致;MOGAImp 的迭代次数设置为 250,种群个数为 100,交叉率为 0.7,变异率为 0.02,由于该算法为多目标算法,搜索得到的帕累托解无优劣之分,因此从帕累托档案中随机选择一个帕累托解作为输出;DIPSO 算法的迭代次数为 250,种群个数为 100,惯性权重 $w=0.5$,加速因子 $\alpha=\beta=1$。MICE 算法使用 RStudio 软件运行,其他算法使用 MATLAB2018a 软件运行。

实验使用 K 近邻分类器,K 值设置为 5。缺失率设置为 5%、10%、20%、30%、40% 和 50%,采用分类正确率作为评估指标以及 DIPSO 的优化目标,其计算方法为:

$$A=\frac{P}{S} \tag{4}$$

式中,S 表示样本总数,P 表示正确分类的样本数。在每个测试条件下,算法独立运行 20 次,取分类正确率的平均值作为算法运行结果。

表 2 是 7 种方法在 6 个测试数据集上不同缺失率条件下的分类正确率实验结果。

表 2 分类正确率实验结果

Tab. 2 Experimental results of accuracy indicator

%

数据集	缺失率	MI	KNN	REM	MICE	EPF	MOGAImp	DIPSO
BCC	5	49.78	51.58	49.58	48.57	**66.74**	47.61	60.49
	10	48.66	47.92	53.13	48.00	63.33	49.96	**65.82**
	20	53.30	55.00	42.17	48.57	**71.88**	48.26	68.41
	30	46.12	54.31	54.08	49.71	**73.29**	55.94	66.09
	40	51.49	50.56	45.04	49.71	68.35	54.89	**71.85**
	50	52.75	46.47	51.20	48.57	**76.44**	56.68	66.38
Park	5	84.10	82.56	83.85	82.71	**85.90**	85.13	84.36
	10	80.00	84.36	82.82	84.75	**87.82**	84.36	87.18
	20	80.77	86.92	82.31	84.07	**89.10**	80.51	87.95
	30	78.97	84.87	84.10	77.97	89.62	88.46	**90.26**
	40	77.95	80.77	82.05	79.32	88.72	80.77	**89.23**
	50	81.28	81.79	80.00	80.68	87.82	83.33	**90.00**
Lymp	5	77.03	73.53	75.29	75.91	79.03	77.52	**87.43**
	10	74.95	78.36	72.72	74.55	82.63	80.33	**87.83**
	20	73.82	73.74	70.95	73.64	79.69	77.49	**88.20**
	30	73.55	72.59	70.23	74.55	86.02	78.00	**93.60**
	40	65.41	69.83	74.68	64.09	83.71	79.39	**92.57**
	50	67.24	66.82	67.63	65.91	85.34	82.95	**92.87**
MONK	5	31.53	28.32	32.81	35.54	34.13	33.85	**44.74**
	10	35.10	35.78	32.32	36.31	41.77	48.09	**49.42**
	20	38.44	37.81	36.76	42.77	47.83	54.70	**56.61**
	30	40.95	42.32	37.77	44.15	50.95	**61.38**	61.09
	40	44.09	44.39	42.37	46.46	53.28	63.46	**63.58**
	50	45.13	46.36	44.08	49.54	54.66	59.93	**63.77**
Zoo	5	81.00	80.71	78.60	85.33	88.33	86.21	**93.10**
	10	81.79	86.60	83.14	80.00	89.90	87.21	**97.00**
	20	80.64	86.76	86.24	84.67	89.38	84.14	**97.52**
	30	80.52	80.62	85.21	83.33	89.52	86.05	**96.07**
	40	77.33	78.00	82.50	84.67	88.68	85.55	**98.52**
	50	78.31	84.60	81.62	54.67	88.06	86.74	**96.07**

续表

数据集	缺失率	MI	KNN	REM	MICE	EPF	MOGAImp	DIPSO
StH	5	64.44	63.15	61.30	64.69	**70.37**	61.30	**70.37**
	10	64.26	66.48	64.07	60.74	74.07	68.70	**74.81**
	20	63.52	62.96	66.11	67.16	77.22	72.78	**77.96**
	30	62.59	63.15	65.37	61.73	**76.85**	73.70	75.19
	40	62.96	56.30	65.93	63.70	76.48	75.00	**77.41**
	50	62.22	57.96	60.37	61.23	74.35	73.33	**77.78**

从较好解统计的角度,对表 2 提供的结果进行分析。DIPSO 在多数情况下均提供了较好的结果,经过统计,DIPSO 一共提供了 27 个较好解,EPF 一共提供了 9 个较好解,MOGAImp 提供了 1 个较好解。进一步,在 BCC 上,当缺失率为 5%、20%、30% 和 50% 时,EPF 提供了较优值,DIPSO 在缺失率为 10% 和 40% 时表现较好;在 Park 上,EPF 的表现与 DIPSO 相当,在缺失率为 30%、40% 和 50% 时,DIPSO 提供了较好解;在 Lymp、MONK 和 Zoo 上,DIPSO 好于其他对比算法,除了在 MONK 数据集上,当缺失率为 30% 时,MOGAImp 提供了较优值;在 StH 上,DIPSO 也优于其他对比算法,除了在缺失率为 30% 时,EPF 提供了较优解。

从算法性能的提升程度上看,DIPSO 在 BCC 数据集上的分类正确率比其他算法提升了约 26.5%,在 Park 上提升了约 5.9%,在 Lymp 上提升了约 21.3%,在 MONK 上提升了约 33.4%,在 Zoo 上提升了约 16.6%,在 StH 上提升了约 14.8%。从整体上看,DIPSO 的性能相比其他算法提升了约 19.8%。

下面从两个角度对结果进行深入分析。从技术角度看,较好解均是由基于学习的方法提供,这是由于基于统计的方法无法有效利用标签信息或标签与特征之间的关系,而基于学习的方法能够同时利用标签和特征的信息。从方法的角度看,DIPSO 优于 EPF 和 MOGAImp,这是由于 EPF 主要采用模糊 C 均值的方法填补数据,MOGAImp 则是采用已有的数据填补缺失值,它们都无法有效利用缺失数据的随机性。这表明了 DIPSO 能够有效利用标签信息和缺失数据的随机性,验证了方法的有效性和优越性。

分析实验结果中产生的一个现象,很多方法在测试数据集上的指标值并没有因为缺失率的提高而降低,尤其在 Lymp 和 MONK 数据集上,DIPOS 的指标值随着缺失率的提高反而出现了一定程度的提升。造成这种情况的原因可能是,特征之间具有较强的冗余性,在一定范围内的随机数据缺失并不会导致学习难度的提高,移除缺失数据或者采用有效的填补策略可能使分类超平面更加容易学习[25]。

下面通过实验分析 DIPSO 算法的时间复杂度。显然,基于统计方法的时间复杂度要小于基于学习的方法,因此这里仅对 MOGAImp、EPF 和 DIPSO 做详细的对比,3 种方法在不同缺失率和数据集上的时间开销堆积直方图如图 2 所示。

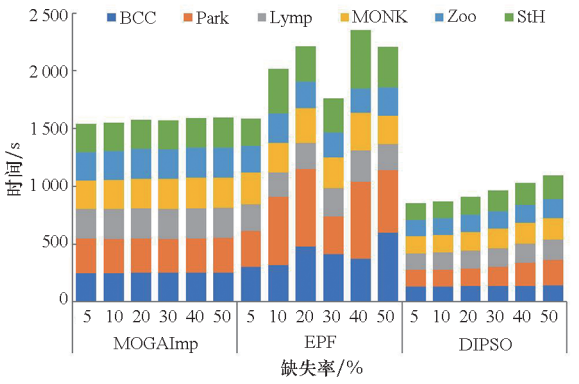

图 2　时间开销堆积直方图

Fig. 2　Stacked histogram of time costs

从图 2 可以看出,在同一个数据集上,DIPSO 的时间开销并没有因为缺失率的增加而发生显著的变化,这是由于 DIPSO 的粒子长度仅与连续型特征数以及离散型特征中的不重复值个数相关,与缺失数据的数量无关。

DIPSO 的时间开销要低于 EPF 和 MOGAImp。这是由于 DIPSO 仅对填补模型的参数进行优化,填补时通过参数生成填补值即可,时间开销较低;而 EPF 在迭代中需要使用模糊 C 均值操作,MOGAImp 需要维护帕累托档案,导致增加了时间耗费。

上述实验表明,DIPSO 具有较好填补性能的同时,也具有较优的时间开销,能够适用于实际业务系统,具有较好的实用性。

3 结论

数据缺失是系统常见的问题,可能导致算法性能下降或不可用。为了解决分类系统中的数据缺失问题,提出基于粒子群优化的数据填补方法DIPSO。将发生缺失数据的特征分为连续型特征和离散型特征,将连续型特征数据取值建模为高斯分布,均值和标准差作为参数,将离散型特征取值建模为概率参数,以分类正确率作为目标通过粒子群算法对参数进行优化。在6个数据集上与4种基于统计的方法和2种基于演化算法的方法进行对比,结果验证了DIPSO方法的有效性和优越性,同时具有较好的时间开销。

但是DIPSO仍然存在一些缺点:首先,目前仍然是通过人工的方式判定发生数据缺失的特征类型,缺少智能的判别方式;然后,没有考虑连续型特征取值不满足高斯分布的情况下,如何对数据分布进行建模;最后,没有深入探究如何通过缺失值附近的数据点来提升填补的准确性。下一步,将从这几个方面入手,进一步提升DIPSO方法的综合性能及其可扩展性。

参考文献(References)

[1] MIRZAEI A, CARTER S R, PATANWALA A E, et al. Missing data in surveys: key concepts, approaches, and applications[J]. Research in Social and Administrative Pharmacy, 2022, 18(2): 2308-2316.

[2] LEE M, AN J, LEE Y. Missing-value imputation of continuous missing based on deep imputation network using correlations among multiple IoT data streams in a smart space[J]. IEICE Transactions on Information and Systems, 2019, E102-D(2): 289-298.

[3] LUO Y. Evaluating the state of the art in missing data imputation for clinical data[J]. Briefings in Bioinformatics, 2022, 23(1): bbab489.

[4] LYNGDOH G A, ZAKI M, ANOOP KRISHNAN N M, et al. Prediction of concrete strengths enabled by missing data imputation and interpretable machine learning[J]. Cement and Concrete Composites, 2022, 128: 104414.

[5] CHEN S X, HAZIZA D. Recent developments in dealing with item non-response in surveys: a critical review[J]. International Statistical Review, 2019, 87(S1): S192-S218.

[6] MIAO X Y, GAO Y J, GUO S, et al. Incomplete data management: a survey[J]. Frontiers of Computer Science, 2018, 12(1): 4-25.

[7] CHAN R K C, LIM J M Y, PARTHIBAN R. A neural network approach for traffic prediction and routing with missing data imputation for intelligent transportation system[J]. Expert Systems with Applications, 2021, 171: 114573.

[8] OSMAN M S, ABU-MAHFOUZ A M, PAGE P R. A survey on data imputation techniques: water distribution system as a use case[J]. IEEE Access, 2018, 6: 63279-63291.

[9] JADHAV A, PRAMOD D, RAMANATHAN K. Comparison of performance of data imputation methods for numeric dataset[J]. Applied Artificial Intelligence, 2019, 33(10): 913-933.

[10] MIAO X Y, WU Y Y, CHEN L, et al. An experimental survey of missing data imputation algorithms[J]. IEEE Transactions on Knowledge and Data Engineering, 2023, 35(7): 6630-6650.

[11] EMMANUEL T, MAUPONG T, MPOELENG D, et al. A survey on missing data in machine learning[J]. Journal of Big Data, 2021, 8(1): 140.

[12] KAMEYAMA K. Particle swarm optimization: a survey[J]. IEICE Transactions on Information and Systems, 2009, 92(7): 1354-1361.

[13] HOUSSEIN E H, GAD A G, HUSSAIN K, et al. Major advances in particle swarm optimization: theory, analysis, and application[J]. Swarm and Evolutionary Computation, 2021, 63: 100868.

[14] SHAMI T M, EL-SALEH A A, ALSWAITTI M, et al. Particle swarm optimization: a comprehensive survey[J]. IEEE Access, 2022, 10: 10031-10061.

[15] JEON Y, HWANG G. Bayesian mixture of Gaussian processes for data association problem[J]. Pattern Recognition, 2022, 127: 108592.

[16] WANG D S, TAN D P, LIU L. Particle swarm optimization algorithm: an overview[J]. Soft Computing, 2018, 22(2): 387-408.

[17] KABIR G, TESFAMARIAM S, HEMSING J, et al. Handling incomplete and missing data in water network database using imputation methods[J]. Sustainable and Resilient Infrastructure, 2020, 5(6): 365-377.

[18] TROYANSKAYA O, CANTOR M, SHERLOCK G, et al. Missing value estimation methods for DNA microarrays[J]. Bioinformatics, 2001, 17(6): 520-525.

[19] SCHNEIDER T. Analysis of incomplete climate data: estimation of mean values and covariance matrices and imputation of missing values[J]. Journal of Climate, 2001, 14(5): 853-871.

[20] VAN BUUREN S, GROOTHUIS-OUDSHOORN K. Mice: multivariate imputation by chained equations in R[J]. Journal of Statistical Software, 2011, 45(3): 1-67.

[21] SUN X Y, WANG Z, HU J T. ELM-PSO-FCM based missing values imputation for byproduct gas flow data analysis[C]// Proceedings of the IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), 2019: 56-59.

[22] LOBATO F, SALES C, ARAUJO I, et al. Multi-objective genetic algorithm for missing data imputation[J]. Pattern Recognition Letters, 2015, 68: 126-131.

[23] SCHNEIDER T. RegEM: regularized expectation maximization[EB/OL]. (2017-7-13)[2022-06-01]. https://github.com/tapios/RegEM.

[24] VAN BUUREN S, GROOTHUIS-OUDSHOORN K, VINK G, et al. Mice: multivariate imputation by chained equations[EB/OL]. (2023-06-05)[2023-12-01]. https://cran.r-project.org/web/packages/mice/.

[25] ZURADA J. Does removing/replacing missing values improve the models' classification performances?[J]. International Journal of Management & Information Systems (IJMIS), 2012, 16(3): 215-220.