叶晟,吴晓朝. 基于网格划分和 LLE 的高维数据离群点自适应检测方法[J]. 湖南科技大学学报(自然科学版), 2023, 38(1):85-91. doi:10.13582/j.cnki.1672-9102.2023.01.011

YE S, WU X C. An Adaptive Outlier Detection Method for High Dimensional Data Based on Grid Partition and LLE [J]. Journal of Hunan University of Science and Technology (Natural Science Edition), 2023, 38(1):85-91. doi:10.13582/j.cnki.1672-9102.2023.01.011

基于网格划分和 LLE 的高维数据离群点自适应检测方法

叶晟,吴晓朝*

(广州商学院 信息技术与工程学院,广东 广州 511363)

摘 要:针对目前高维数据量急剧增加,离群点检测技术精准度低、所需内存大、检测时间长等问题,提出了基于网格划分和局部线性嵌入方法(Locally Linear Embedding, LLE)的高维数据离群点自适应检测方法.根据高维数据的空间维度进行网格划分,设定单元格邻近单元数量,降低运行开销,减少计算时间.采用局部线性嵌入方法(LLE),分析不同组合数据点的局部特性,准确描述高维数据结构,完成高维数据集预处理.采集高维数据集合中小部分重要信息,保证采集结果的准确性,利用 MapReduce 编程模型,将大任务划分为多个不同的小任务,展开分布式处理.通过网格密度计算离群度,提升检测效率,优先过滤空白网格单元,降低空间开销,减小所需内存,从而实现高维数据离群点自适应检测.实验结果表明:所提方法在不同数据集大小测试中,执行时间更短,检测精确度更加稳定;在维度测试中,所需内存更少.证明所提方法能够有效降低执行时间和内存,提升检测结果的精确度.

关键词:网格过滤;LLE;高维数据;离群点;自适应检测;预处理

中图分类号:TP391 **文献标志码**:A **文章编号**:1672-9102(2023)01-0085-07

An Adaptive Outlier Detection Method for High Dimensional Data Based on Grid Partition and LLE

YE Sheng, WU Xiaochao

(School of Information Technology & Engineering, Guangzhou College of Commerce, Guangzhou 511363, China)

Abstract: In view of the rapid increase of high-dimensional data, low accuracy of outlier detection technology, large memory and long detection time, an adaptive detection method of high-dimensional data outliers based on grid partition and LLE (Locally Linear Embedding) is proposed. The grid is divided according to the spatial dimension of high-dimensional data, and the number of adjacent cells is set to reduce the running cost and computing time. LLE method is used to analyze the local characteristics of the same combined data points, accurately describe the high-dimensional data structure, and complete the preprocessing of high-dimensional data sets. It collects important information from small parts of high-dimensional data sets to ensure the accuracy of collection results. Using MapReduce programming model, the large task is divided into several different small tasks for distributed processing. The detection efficiency is improved by calculating the degree of outliers through

收稿日期:2022-09-07

基金项目:2021 年度广东省重点建设学科科研能力提升项目资助(2021ZDJS120)

* **通信作者**,E-mail: wuxiaochao@gcc.edu.cn

the grid density. The blank grid cells are filtered first to reduce the space overhead and the required memory, so as to realize the adaptive detection of outliers in high-dimensional data. Experimental results show that the proposed method has shorter execution time and more stable detection accuracy in different data set sizes. In dimension tests, few memory is required. It is proved that the proposed method can effectively reduce the execution time and memory, and improved the accuracy of detection results.

Keywords: grid filtering; LLE; high-dimensional data; outliers; adaptive detection; preprocessing

随着计算机技术的快速发展,信息的采集和存储等相关技术逐渐进步,各个研究领域产生的数据也随之增加,不仅仅是数据量的增加,而且增加的这些数据中还包含了各种高维数据以及不同种类的异构数据.为此,如何在这些数据中获取有效信息,需要展开数据挖掘处理.离群点检测技术是数据挖掘的重要分支,对异常数据检测具有十分重要的意义[1-2].离群点检测的主要任务是采用数据挖掘技术发现与正常数据存在偏差比较大的数据点,从而达到噪音消除以及异常排除的目的.由于导致离群点产生的原因有很多,而且随着大数据技术的飞速发展,数据维度的增加也提升了数据结构的复杂程度,因此如何有效地提高检测精准度,降低检测时间,对高维数据离群点检测具有重要的意义.

为了解决高维数据中离群点检测问题,在最短的时间内提高检测精准度,国内相关专家针对高维数据离群点检测方面的内容展开了大量研究.其中,杜旭升等[3]首先计算数据集的全部邻居节点数量,然后计算邻域系统密度取值,将两者展开对比分析,判断目标对象和邻居趋向是否在相同的簇内,最终输出的即为离群点对象,实现数据检测,但是,随着数据集的增多,这样的检测精准度会不高;杨晓玲等[4]将对象区域密度和邻近对象两者有效结合,得到两者间相对距离,以最大化边分割方法,对应最小生成树结构,快速分割离群簇和离群点,最终实现离群点检测,但当数据维数增多时,需要的内存更大;邱华等[5]对海量历史数据展开预处理,使用极限学习机,对预处理的历史数据进行训练,得到预测后的局部离群因子的阈值,并采用 WLOF 阈值对数据聚类处理,最终实现离群点检测.但该方法的实验时间偏长,可行性较低.

针对上述问题,赵向兵等[6]提出一种基于相关子空间的离群数据检测算法,该算法首先根据数据局部密度分布特征得出稀疏度矩阵,然后计算各属性维度中数据稀疏度相似因子,确定子空间向量及相关子空间后,结合数据稀疏度和维度权值得出数据对象的离群因子,并最终确定离群数据.受这种检测方法的启发,本文提出了一种基于网格划分和 LLE 的高维数据离群点自适应检测方法.该方法对高维数据进行网格划分,采用局部线性嵌入(LLE)方法对高维数据集进行预处理,通过高维数据集中重要信息的采集,检测出高维数据离群点.经实验测试结果表明:所提方法可以大幅度降低执行时间以及内存使用,同时还能够获取高精度的检测结果.

1 高维数据离群点自适应检测方法

在本次研究中,主要通过网格过滤法对高维数据进行检测,可有效收集数据集的重要信息,提升检测效率.根据高维数据的空间维度,进行网格划分处理;采用 LLE 方法,得到高维数据集预处理结果;根据数据集的重要信息输出高维离群点采集结果,最终检测出高维数据离群点.

1.1 网格划分

为了有效地实现高维数据离群点自适应检测,首先需要根据高维数据的空间维度,进行网格划分处理.

设定属性集为 $x_1,x_2,\cdots,x_d$, 对应的 d 维欧式距离空间可以表示为式(1)的形式.

$$\boldsymbol{R} = x_1,x_2,\cdots,x_d. \tag{1}$$

式中:$\boldsymbol{R}$ 为欧式距离空间.相邻 2 个网格中会有 1 个公共面或者边.在已知数据维数的空间内,为了降低运行开销,减少计算时间,需要对单元格邻近单元的数量进行设定.

给定数据集,设定空间维数,将全部连通的稠密网格单元的最大集合设定为聚类区域.当数据集经过网格划分处理后,需要对网格划分的数据集展开详细的分析和研究,由此可以定义聚类区域.分析网格的相关定义可知:在网格划分的数据结构中,全部网格单元都是相等的,即各个网格单元格的大小以及体积也是完全一样的.

在设定维数空间内对网格划分的过程中[7-8],设定维度的距离为 l,经过划分后获取的网格单元为 l^d,具体的表达式为

$$l^d = \frac{\boldsymbol{R}}{|\lambda - s_i|}. \tag{2}$$

式中:λ 为相邻单元数量;s_i 为第 i 个网格单元.

其中,网格划分的详细步骤如下所示:

1)随机选择数据子空间中没有访问的点,通过事先设定的维度间隔距离计算所选择数据点所属于的网络单元;

2)组建哈希函数,将获取的网格单元信息映射到对应的哈希表中,计算网格单元所包含的数据点总数,根据事先设定的取值判断网格单元的类型;

3)遍历哈希表中全部标记为稠密的网格单元,同时以邻近网格单元特征为判断依据,如果判断其为边界网格单元,则需对网格标记;

4)重复上述操作步骤,直至完成全部网格的划分.

1.2　高维数据集预处理

在离群点检测过程中,高维数据集预处理是核心操作步骤.为了准确描述高维数据结构,采用 LLE 方法[9-10]对高维数据集进行预处理,同时分析不同组合数据点的局部特性,经过计算得到高维数据对应的全局结构.高维数据集预处理的详细操作流程如图 1 所示.

1)优先输入高维数据集,在高维数据集中采用欧式距离获取 k 个最近邻点;

2)通过最小化公式重构误差获取各个高维数据点以及邻近数据点的重构权值,如式(3)所示.

$$D(w) = \sum_{k=1} \left| \bar{x}_i - \sum_{i=1} \omega_{i,j} \bar{x}_j \right|^2. \tag{3}$$

式中:$D(w)$ 为重构权值;$\bar{x}_i$ 和 $\bar{x}_j$ 为近邻点对应的 2 个限制条件;$\omega_{i,j}$ 为缩放过程中产生的重构权值.

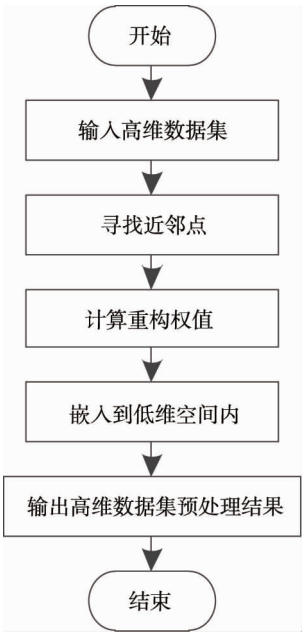

图 1　高维数据集预处理

为了确保重构误差最小化,可以将式(3)转换为

$$D(w) = \begin{cases} \left| \bar{x}_i - \sum_{i=1} \omega_{i,j} \bar{x}_j \right|^2; \\ \left| \sum_{i=1} \omega_{i,j} \bar{x}_i - \sum_{i=1} \omega_{i,j} \bar{x}_j \right|^2. \end{cases} \tag{4}$$

通过式(5)展开计算可以获取数据点 a 对应的最佳权重取值 ω_a.

$$\omega_a = \frac{\sum_{i=1} G_{i,j}}{\sum_{i=1} G_{i,j}^{-k}}. \tag{5}$$

式中：$G_{i,j}$ 为重构误差的取值范围.

假设设定的近邻点数量大于输入高维数据集的维度,为了确保计算结果的唯一性,需要对重构误差的取值范围实行以下形式的改动,如公式(6)所示.

$$G_{i,j} = \sum_{i=1} \sum_{j=1} \left(\partial_k \left(\frac{\omega_{i,j}}{D(w)} \right) \right). \tag{6}$$

式中：∂_k 为近邻点总数[11-12].

3)获取低维空间的向量 $\varphi(y)$,需将高维数据集嵌入到低维空间中,进而通过式(7)重构误差.

$$\varphi(y) = \sum_{i=1} \sum_{j=1} \left(\frac{\partial_k - \omega_{i,j}}{G_{i,j}} \right). \tag{7}$$

4)输出高维数据集预处理结果.

1.3 采集高维数据离群点

在采集高维数据离群点的过程中,高维数据集合中只有小部分数据可以提供相对重要的信息,剩余部分信息则会在数据挖掘过程产生干扰,影响采集结果的准确性.

上下文信息是离群点的重要组成部分,不仅可以体现离群数据和其他数据之间的一致,同时还可以提供有利用价值的信息,例如离群数据的差异性以及相关含义等.

子空间是全空间的一部分,因此,可以获取离群数据在相关子空间的离散程度,如式(8)所示.

$$F(o) = \max\left(\frac{p_{\mathrm{rs}}(o)}{\sqrt{2}(e_{\mathrm{obj}})\ [p_{\mathrm{rs}}(o)\ ]^2} \right). \tag{8}$$

式中：$F(o)$ 为相关子空间的离散程度；o 为离散程度；p_{rs} 为局部差异度阈值；e_{obj} 为属性维集的离散程度.

MapReduce 编程模型被广泛应用于不同的研究领域,通过 MapReduce 编程模型采集高维数据离群点,将大任务划分为多个不同的小任务,展开分布式处理.MapReduce 编程模型在运行过程中主要包含以下几个操作步骤：

1)数据分割.根据高维数据处理需求将数据分割处理,得到对应的输入数据；

2)负载均衡.通过集群计算节点的计算效率,根据计算结果完成集群资源的调度处理；

3)容错处理.分别计算节点以及数据,对其展开错误判断处理,收集全部节点的错误或者告警信息；

4)相互通信.管理高维数据集中全部需要展开通信的节点,确保节点之间通信正常,同时还需要确保通信的安全性.

综上所述,可以总结得到采集高维数据离群点的操作流程：

1)输入经过预处理的高维数据集；

2)展开并行运算处理,同时输出相关子空间的离散程度；

3)通过并行计算获取不同数据对象对应的维度信息,进而形成对应的离散程度；

4)通过离散程度组建稀疏度矩阵；

5)根据离散程度确定产生异常的局部因子,将离群因子按照从大到小的顺序排列,最终输出高维离群点采集结果.

1.4 检测高维数据离群点

由于数据对象的各个属性维度形成规则不同,所以各个属性之间的分布也存在十分明显的差异.在分布相对稀疏的高维空间中,数据集中随机 2 个点之间的最大和最小距离差值均接近于 0,通过式(9)计算对应的差值 σ.

$$\sigma = \frac{\mathrm{dist}_{\max} - \mathrm{dist}_{\min}}{\mathrm{dist}_{\min}}. \tag{9}$$

式中: $\mathrm{dist}_{\max}$, $\mathrm{dist}_{\min}$ 为最大和最小距离.

由于距离度量的失效,传统离群点通过欧式距离展开相似度度量不再具有任何意义[13-14],所以,采用基于角度方差的度量方式完成高维数据离群点检测.角度方差自身只可以描述不同数据之间的方向关系,但是无法描述点和点之间的距离位置关系.

为了提升检测效率,通过网格密度描述数据对象的近邻分布,采用公式(10)表示.

$$\mathrm{apprABOF}(\vec{A}) = \omega_{i,j}\mathrm{ABOF}(\vec{A})\left(\frac{(AB)\ (AC)}{(BC)}\right). \tag{10}$$

式中: $\mathrm{apprABOF}(\vec{A})$ 为近邻分布结果; $\mathrm{ABOF}(\vec{A})$ 为角度离群因子; AB, AC, BC 为不同数据集.

如果当高维数据集中离群数据点数量明显低于正常数据数量,在维数比较低的相关子空间内又存在分布相对密集的区域,即在子空间相关属性选择过程中,我们需要通过局部密度分布描述差异因子的取值.因此,对比网格集合中全部网格的体积,我们可以选取体积最大的网格设定为标准网络.其中,数据对象和对应邻近点之间的属性维度距离可以组成网格属性维度半径向量,根据以上属性对网格划分处理,这样可以有效地避免过度稀疏情况的产生.但由于高维数据集分布十分稀疏,因此,网格在划分过程中可能会含有部分不包含任何数据的空白网格单元.所以,在判断网格类型前期,我们需要优先过滤掉空白网格单元,从而全面降低空间开销,同时还能够减少时间复杂度.

另外,网格的存储结构也会对算法的复杂程度产生很大的影响,对于高维大数据的稀疏特性,为了全面减少系统的存储空间,同时降低查询以及遍历次数,我们可以借助 Hash 表完成网格单元的存储工作.这样,通过网格的相邻关系,采用哈希表存储网格单元信息,我们将划分处理后的各个子空间进行映射处理,从而形成一张表.

通过上述分析,可以获取基于角度方差的高维数据离群点检测步骤,如图 2 所示.

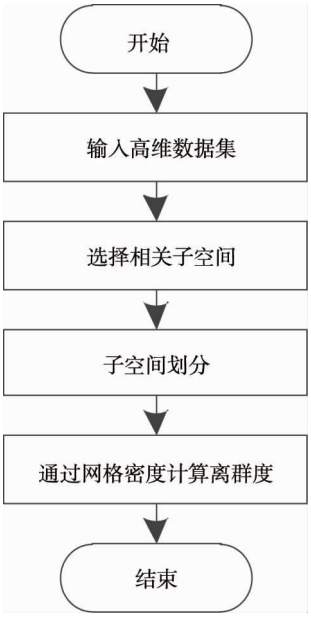

图 2　高维数据离群点检测

1)相关子空间选择.根据局部密度分布矩阵展开相关属性选择处理,进而构建相关子空间,完成高维数据集预处理.

2)子空间划分处理.采用局部密度分布矩阵选择标准网络,确定子空间内网格划分的属性维度半径.针对已经完成划分处理的网格而言,需要删除空白网格,同时将其存入到哈希表中遍历处理.将初始阶段选择的标准网格设定为中心网络,以此为依据对相邻网络展开拓展操作处理,同时通过相对密度选择网格,经过修剪处理得到运行正常的网络.

3)通过网格密度计算离群度[15-16].对候选网格中全部数据的影响角度方差因子进行计算,设定网格密度为局部相对重要指标,分别对比不同子空间内数据对象角度方差因子,对比离群度,将排名比较高的数据对象作为离群点输出,进而检测出高维数据离群点.

2　仿真实验

为了验证所提基于网格划分和 LLE 的高维数据离群点自适应检测方法的有效性,需要展开实验测试.

2.1　测试环境和数据集

实验使用的计算机系统为 Windows 10,配置 i7 处理器和 256 GB 运行内存,UCI 数据集为机器学习标准测试常用数据集,本文选取 UCI 数据集中的典型高维数据集 German, Innosphere 和 Segment 作为测试数据集,其中样本数据具体描述如表 1 所示.并将 MATLAB 仿真软件作为测试平台.

2.2　测试结果

实验分别对所提的网格划分检测方法、文献[3]的密度检测方法和文献[4]的距离检测方法展开离群点检测测试,根据实验需求,设定数据空间维数,分析不同密度阈值下 3 种方法的执行时间变化情况,如表

2 所示.

表 1 实验数据集描述

名称	样本数量	维数	类别
German	1 000	24	2
Innosphere	351	34	2
Segment	2 310	19	7

表 2 不同密度阈值下 3 种方法的执行时间测试结果对比

密度阈值	执行时间/s		
	网格划分法	密度检测法	距离检测法
200	2 463	5 214	5 325
190	2 341	5 263	5 425
180	2 274	5 412	5 387
170	2 145	5 236	5 374
160	2 056	5 102	5 366
150	1 966	5 036	5 347
140	1 824	5 011	5 388
130	1 745	5 124	5 302
120	1 625	5 269	5 344
110	1 541	5 374	5 321
100	1 420	5 286	5 345
90	1 362	5 212	5 333
80	1 102	5 425	5 310
70	9 54	5 222	5 347
60	745	5 412	5 363
50	651	5 315	5 247
40	543	5 247	5 352
30	442	5 236	5 428
20	362	5 111	5 366
10	251	5 284	5 274

从表 1 中的实验数据可知:在密度阈值不断下降的情况下,3 种方法的执行时间发生了明显的变化.本文所提方法的执行时间会随着密度阈值的降低而降低,而另外 2 种方法的执行时间基本处于相对稳定的水平.由此可知,对于所提方法而言,密度阈值越小,执行时间越短.

实验进一步测试分析对于不同大小的数据集 3 种方法的执行时间变化情况,如图 3 所示.

从图 3 中的实验数据可知:3 种方法的执行时间均会随着数据集的增加而增加.但是相比另外 2 种方法,本文所提方法的执行时间明显更短一些,充分验证了该方法的时效性较高.

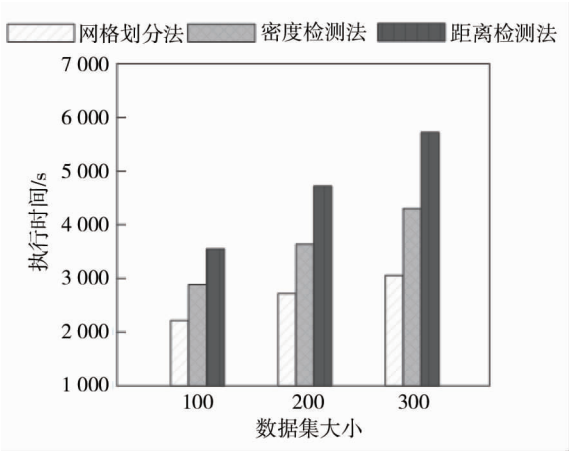

图 3 不同大小数据集 3 种方法的执行时间测试结果对比

为了验证本文所提方法的检测能力,选取检测精确度作为测试指标,检测精确度结果越高,则说明检测结果的准确性越高,具体的实验测试结果如图 4 所示.

从图 4 中的实验数据可知:本文所提方法的检测精确度一直处于比较稳定的状态,均值在 90%以上.而另外 2 种方法的检测精确度明显偏低一些,同时还不同程度受到数据集大小的影响.由此可知,在 3 种方法中,本文所提方法可以有效地提升检测准确性.

在此基础上,进一步分析不同维数下 3 种方法的内存使用情况,如图 5 所示.

由图 5 中的实验数据可知:3 种方法的内存会随着维数的增加而增加.在 3 种检测方法中,本文所提方法所使用的内存一直较低,而另外 2 种方法则相对较高一些.当维数为 19 时,所提方法的内存为 365 kB,而另外 2 种方法的内存分别为 418 和 440 kB.由此可知,所提方法能够有效降低内存.

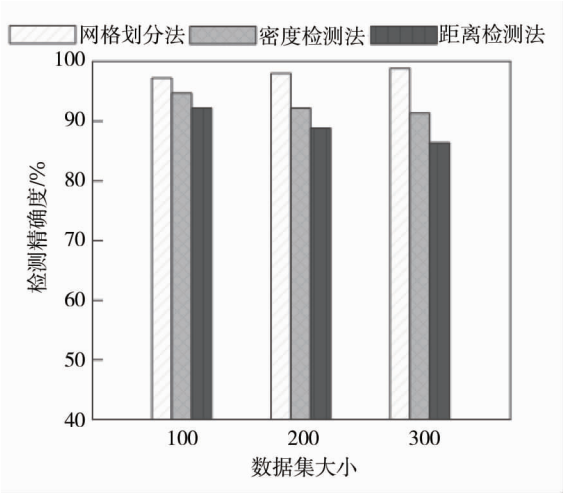

图 4　不同大小数据集 3 种方法的检测精确度测试结果对比

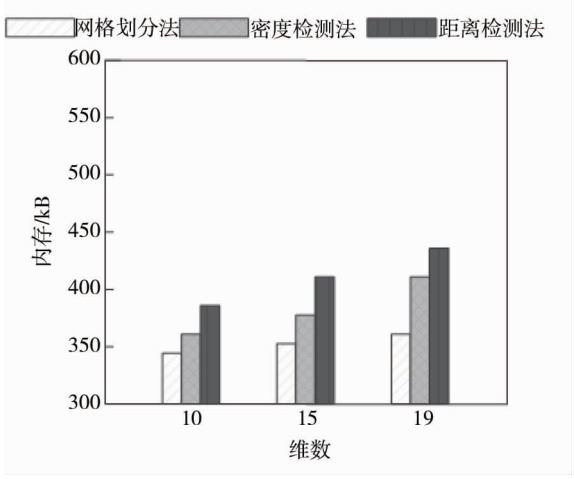

图 5　不同维数下 3 种方法的内存使用情况测试结果对比

3　结论

1)所提的高维数据离群点自适应检测方法执行时间较短,占用内存较小并且检测精确度较为稳定.

2)通过对高维数据空间维度的网格划分和采用局部线性嵌入方法(LLE),并结合相关的编程模型,可以在保证检测精度的前提下极大地提高高维数据离群点的检测效率.这对于某些需要对海量数据进行分析和检测的应用(比如商业软件后台的大数据分析)在进行软件设计时具有良好的借鉴意义.

参考文献:

[1] 毛亚琼,田立勤,王艳,等.引入局部向量点积密度的数据流离群点快速检测算法[J].计算机工程,2020,46(11):132-138.

[2] 周玉,朱文豪,房倩,等.基于聚类的离群点检测方法研究综述[J]. 计算机工程与应用,2021,57(12):37-45.

[3] 杜旭升,于炯,陈嘉颖,等.一种基于邻域系统密度差异度量的离群点检测算法[J].计算机应用研究,2020,37(7):1969-1973.

[4] 杨晓玲,冯山,袁钟.基于相对距离的反 k 近邻树离群点检测[J].电子学报,2020,48(5):937-945.

[5] 邱华,乔涵哲,虞董平,等.基于极限学习机的密度聚类离群点检测研究[J].控制工程,2021,28(8):1676-1682.

[6] 赵向兵,张天刚.基于相关子空间的高维离群数据检测算法[J].计算技术与自动化,2022,41(1):82-86

[7] 王瑞,胡志平,任翔,等.2.5D 有限元建模关键问题——边界条件、网格划分及计算域选取[J].振动工程学报,2021,34(1):80-88.

[8] 苏海东,付志,颉志强.基于任意网格划分的二维自动计算[J].长江科学院院报,2020,37(7):160-166.

[9] 蓝雯飞,汪敦志,张盛兰.一种新的降维算法 PCA_LLE 在图像识别中的应用[J].中南民族大学学报(自然科学版),2020,39(1):85-90.

[10] 刘玉敏,梁晓莹,赵哲耘,等.基于 LLE-SVDD 的高维非线性轮廓数据实时监控方法[J].统计与决策,2020,36(19):20-24.

[11] 徐国天.网络入侵检测中 K 近邻高速匹配算法研究[J].信息网络安全,2020,20(8):71-80.

[12] 樊瑞宣,姜高霞,王文剑.一种个性化 k 近邻的离群点检测算法[J].小型微型计算机系统,2020,41(4):752-757.

[13] 梅林,张凤荔,王瑞锦,等.基于网格划分加权的分布式离群点检测算法[J].电子科技大学学报,2020,49(6):860-866.

[14] 陈旺虎,田真,张礼智,等.基于插值的高维稀疏数据离群点检测方法[J].计算机工程与科学,2020,42(6):966-972.

[15] 唐宇坤,邓松,许梦雅,等.基于几何特征的学生评教数据离群点检测算法[J].江西师范大学学报(自然科学版),2021,45(3):292-298.

[16] 林雪.海量不确定数据集中离群点快速检测方法仿真[J].计算机仿真,2021,38(6):378-382.