计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.201000156

基于邻域一致性的异常检测序列集成方法

刘 意 毛莺池 程杨堃 高 建 王龙宝
河海大学计算机与信息学院 南京 211100
水利部水利大数据重点实验室 南京 211100
(1175476508@qq.com)

摘 要 异常检测已广泛应用于多个应用领域,如网络入侵检测、信用卡欺诈检测等。数据维度的增加导致出现许多不相关和冗余的特征,这些特征会掩盖相关特征,出现假阳性结果。由于高维数据具有稀疏性和距离聚集效应,传统的基于密度、距离等的异常检测算法不再适用。大部分基于机器学习的异常检测研究都关注单一模型,而单一模型在抗过拟合能力上存在一定的不足。集成学习模型有着良好的泛化能力,而且在实际应用中展现出比单一模型更好的预测准确性。文中提出了基于邻域一致性的异常检测序列集成方法(Locality and Consistency Based Sequential Ensemble Method for Outlier Detection,LCSE)。首先基于多样性构造异常检测基本模型,其次根据全局集成一致性筛选出异常候选点,最后考虑数据局部邻域相关性选择并组合基本模型结果。通过实验验证,LCSE 相比传统方法异常检测的准确率平均提升了 20.7%,与集成算法 LSCP_AOM 和 iForest 相比,性能(AUC)平均提升了 3.6%,因此其性能优于其他集成方法和神经网络方法。

关键词:高维数据;异常检测;集成多样性;集成一致性;领域相关性
中图法分类号 TP391.4

Locality and Consistency Based Sequential Ensemble Method for Outlier Detection

LIU Yi,MAO Ying-chi,CHENG Yang-kun,GAO Jian and WANG Long-bao
College of Computer and Information,Hohai University,Nanjing 211100,China
Key Laboratory of Water Big Data Technology of Ministry of Water Resources,Nanjing 211100,China

Abstract Outlier detection has been widely used in many fields,such as network intrusion detection,credit card fraud detection,etc. The increase in data dimensions leads to many irrelevant and redundant features,which will obscure the relevant features and result in false positive results. Due to the sparseness and distance aggregation effects of high-dimensional data,the traditional outlier detection algorithms based on density and distance are no longer applicable. Most of the outlier detection research based on machine learning focuses on a single model,which has certain deficiencies in anti-overfitting ability. The ensemble learning model has good generalization ability,and in actual application shows better prediction accuracy than the single model. This paper proposes an outlier detection sequence integration method LCSE based on neighborhood consistency (locality and consistency based sequential ensemble method for outlier detection). Firstly,it constructs a basic model of outlier detection based on diversity,secondly,selects the abnormal candidate points according to the global integration consistency,and finally considers the local neighborhood correlation of the data to select and combine the basic model results. Experiments verify that LCSE has an average outlier detection accuracy increase of 20.7% compared with traditional methods. Compared with the ensemble methods LSCP_AOM and iForest,the performance is increased by 3.6% on average. Therefore,it is better than other ensemble methods and neural network methods.

Keywords High-dimensional data,Outlier detection,Ensemble diversity,Ensemble consistency,Neighborhood correlation

1 引言

计算能力和存储技术的快速发展丰富了各个应用领域的数据采集手段,所采集到的数据具有规模大、维度高的特点。例如在基因 DNA 序列、股票证券交易、网络社交媒体等应用领域,数据维度已达到百维甚至千维[1]。异常检测致力于发现不同于期望(正常)数据模式的异常数据,被广泛应用于多个领域,如网络入侵检测、信用卡欺诈检测、健康诊断等。不

到稿日期:2020-10-27 返修日期:2020-12-08 本文已加入开放科学计划(OSID),请扫描上方二维码获取补充信息。
项目基金:国家重点研发课题(2018YFC0407105);国家自然科学基金重点项目(61832005);华能集团重点研发课题(HNKJ17-21)
This work was supported by the National Key Research Program of China(2018YFC0407105),Key Program of the National Natural Science Foundation of China(61832005) and Key Research Program of China Huaneng(HNKJ17-21).
通信作者:毛莺池(yingchimao@hhu.edu.cn)

同于预测、分类等问题，异常检测的数据集通常缺少标签样本，因此通常采用无监督的检测方法。近年来，研究者们使用集成方法进行无监督异常检测，因为集成方法能有效地提高模型的准确性和稳定性。集成方法的主要工作为：一方面根据多样性组合不同基础检测器和不同数据样本的集成单元，另一方面对集成单元的结果进行选择和组合，得到最终的检测结果。

集成学习能有效提升异常检测的性能，但是现有方法存在以下问题：1）大多数方法不经过筛选，组合所有的基础检测模型。这类方法会导致集成方法的准确性降低，因为对于不同的数据集，基础检测单元的性能也不同，不经选择直接组合模型会导致表现较差的基础模型降低整体性能。2）在选择和组合基础检测模型时，通常会忽略数据的局部相关性，导致输出结果不是最优解。3）选择基础检测模型时，通常会考虑多样性和一致性，没有对二者进行平衡。

“贪婪集成”使用加权皮尔逊相关系数度量得分向量的相似性[2]，该方法致力于结合多样性，选择相似性差异大的集成单元。但仅仅追求多样性，将差异大的结果进行组合，很可能会对结果造成误导。如果基础检测模型中存在误判，其必然与其他基础模型有差异，此时会损失异常检测结果的准确度。为了解决这类问题，Campos 等[3]提出 BoostSelect 方法，计算结果的一致性，并通过 Boosting 的方法选取相关性大的基础检测模型，能有效提升异常检测的准确率，但是该方法仅仅考虑了模型结果的全局一致性，并没有考虑局部邻域相关性。Zhao 等[4]提出平行集成方法 LSCP_AOM，关注数据邻域相关性，利用数据的邻域构造伪标签，用于选择和组合基础模型。LSCP_AOM 能有效提升异常检测的准确率，但未考虑数据的全局信息，导致其在一些数据集上表现不佳。

基于学习的方法被逐渐应用于异常检测，该类方法构造深度神经网络来捕获数据关系。许多神经网络方法，如小波神经网络[5]、长短记忆网络（LSTM）、循环神经网络（RNN）、自编码器（Auto-Encoder）[6-7]等，能有效捕获数据的时序关系，提升时序数据异常检测的性能。在处理有标签的数据时，基于学习的方法表现出优越的性能，但是在处理异常检测这类无监督问题时表现受限[8]。基于学习的方法通常需要大量的数据样本，通过不断迭代学习计算结果，在处理高维数据时计算消耗巨大。许多研究者将集成学习应用到异常检测中，其在处理高维数据时表现稳定。集成学习将来自不同模型的结果组合起来，能有效减小模型的偏差和方差。近年来，基于集成学习的异常检测方法不断涌现，如传统的 Boosting 算法、Bagging 算法[9-10]、平行集成的孤立森林算法（Isolation Forest，iForest）[11]和序列集成算法[12]。基于集成的方法构造多样性的基本异常检测模型，通过一致性选择并组合基本模型。现有的许多集成方法考虑了模型结果的全局一致性以进行基本模型选择，然而一些研究表明，通过捕获数据局部邻域关系能更好地识别出异常[4]。

为了解决以上问题，本文同时考虑全局和局部关系，提出了基于邻域相关性和集成一致性的选择集成方法 LCSE。

LCSE 集成框架如图 1 所示，其首先根据多样性构造基础模型，然后考虑全局集成一致性筛选出异常候选点，最后考虑邻域相关性组合异常得分。本文的主要贡献如下。

（1）构造伪标签（pseudo ground truth）：由于异常检测为无监督问题，序列迭代需要构造伪标签。伪标签初始化时组合异常集成单元池的异常得分，后续每次迭代使用上一次迭代得到的异常得分向量作为伪标签。

（2）基于集成一致性筛选异常候选点（outlier candidates）：根据异常集成单元异常得分的全局一致性，筛选出异常候选点。

（3）基于邻域相关性的异常得分组合：对每个异常候选点构造本邻域，根据其邻域内点的相关性，组合集成单元得到最终异常得分。

实验结果表明：结合了异常集成单元构造多样性、全局集成一致性和局部邻域相关性的 LCSE，有效减小了基础模型的偏差，提高了异常检测模型的准确率和稳定性。

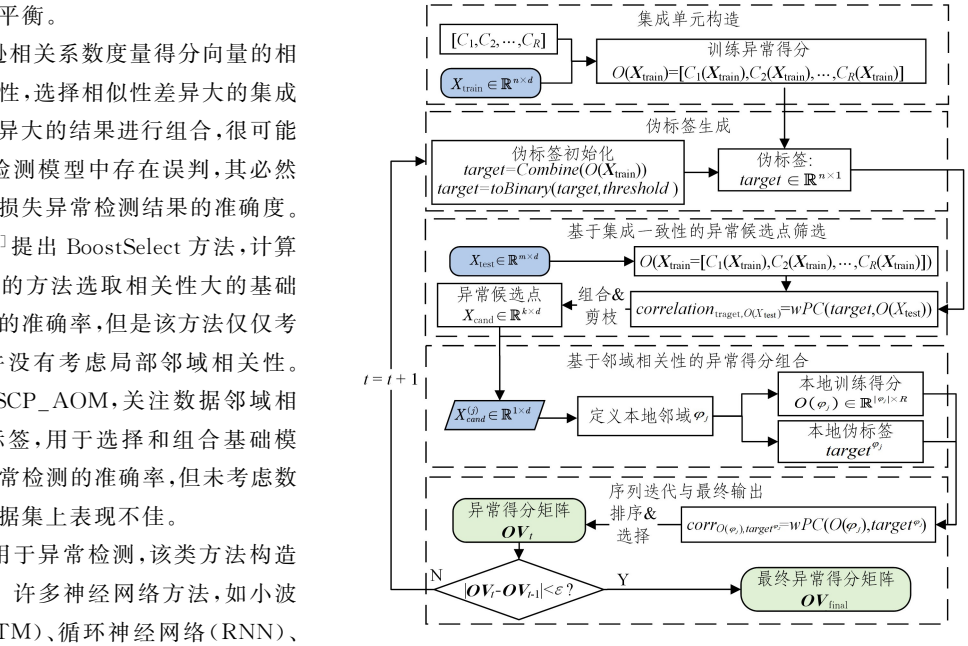

图 1　基于邻域相关性和集成一致性的选择集成方法 LCSE 框架图

Fig. 1　LCSE framework for selecting integration method based on neighborhood correlation and integration consistency

2　相关工作

2.1　集成多样性

集成异常检测主要面临两个挑战，即异常检测基本模型的构造和组合。集成方法在构造基本模型时考虑了多样性，尽可能多地组合各种情况，并通过一致性将多种异常结果组合起来，获得更为鲁棒的异常检测结果。集成多样性体现在数据样本构造多样性、基本模型类型多样性和模型超参数多样性[13]等。

Lazarevic 等[9]最早提出使用集成方法来提升异常检测的性能，其利用了特征套袋法（Feature Bagging，FB）处理大量高维数据集。该方法组合了多个异常检测模型的输出，每个模型从原始数据集随机选择多个特征子集，并由此得到异常得分。每个异常得分代表该数据点异常的概率，从不同模型组合的异常得分来获取更高质量的异常点。Zimek 等[14]提出一种随机子采样技术来评估数据的最近邻域和局部密度，

通常在给定数据集上使用子采样技术，获得不重复的训练样本。该算法通过子采样提升样本的多样性，能有效提升异常检测的性能。其他集成方法也可通过子采样技术生成不同的结果集，达到更高的检测率。Zimek 等[15]从另一个视角提升多样性，提出了一种数据扰乱技术。该方法的主要思想是在欧氏距离空间中利用距离和密度估计概率，通过增加随机噪声来改变每个数据点的属性值，获取更加一致的概率估计。所有扰乱后的数据集在所选特征集上进行计算，可以记录每个数据点的特征并聚合异常得分，得到更稳定、可靠的异常得分。Pasillas-Díaz 等[16]结合随机子采样和 FB 技术，选择不同的数据子集组合多样性基本模型，并在每次迭代时获得不同的结果，充分满足集成多样性。但是，子采样的数据子集大小对异常检测结果的影响较大。

2.2 集成一致性

在构造多样性的异常检测基本模型后，通常依据一致性来选择和组合不同模型。同一样本在不同基本异常检测模型的异常得分多样，将这些异常得分组合起来是一个挑战。Schubert 等[2]使用相似性度量来比较异常得分排序，并使用贪婪集成技术剔除相似性低的异常得分序列，得到最终异常得分结果。Nguyen 等[17]研究了高维数据集上集成异常检测的难度，并提出了一个通用框架，用以整合不同异常检测算法中不相关的模型。不同于每次使用相同的异常检测模型确定异常得分，该集成框架使用多样的异常检测模型来拟合异常得分。他们提出随机子空间异构检测器集成方法（Heterogeneous Detector Ensemble on Random Subspaces，HeDES），通过对多样异常检测模型的组合来解决异构性问题。不同于 Lazarevic 等[9]使用同类型的异常检测基础模型的方法，HeDES 使用不同技术生成不同类型的异常得分，如真实异常得分值和异常标签值。

Campos 等[18]使用 Boosting 的方式组合异常检测基本模型，组合所有基本模型结果作为伪标签，并在每次迭代时计算伪标签与所有基本模型输出的加权皮尔逊相关性，通过相关性来选择基本模型。

2.3 邻域相关性

异常检测中捕获数据对象间的关系相当重要，于是现有的异常检测方法可以分为全局相关和局部相关两类。前者在训练和推理中考虑全部对象，后者仅考虑对象的局部选择，二者的适用性取决于数据的结构。当异常值是差异较大的数据分布时，全局异常点检测算法能获得优越的性能[10]，但往往不能识别高维数据在局部邻域的异常表现[19]。全局模型要处理混合分布情况的数据，同时全局特征也无法有效地表示数据对象在局部区域的分布情况。为了解决这类问题，许多面向局部邻域的算法被提出，如局部异常因子算法 LOF[20]、局部异常概率算法（Local Outlier Probabilities，LoOP）[21]和使用全局邻居的本地子空间异常检测算法（Global-Local Outliers in Sub-Spaces，GLOSS）[19]。Zhao 等[4]提出了一个基于无监督异常检测框架的得分动态组合集成算法（Dynamic Combination of Detector Scores for Outlier Ensemble，DCSO），致力于解决在缺失真值标签的情况下选择和组合不同基本模型异常得分的挑战。DCSO 根据数据的局部相关性选择最合适的基本检测器，利用测试样本附近的 k 个最近样本来初始化其本地邻域，并选择在本地邻域内表现最好的基本检测器进行组合。但是，这些局部数据关系的异常检测算法没有在数据集成上考虑邻域性质，为此 Zhao 等[4]提出平行异常集成中的局部选择组合算法（Locally Selective Combination in Parallel Outlier Ensembles，LSCP_AOM），在训练基本检测器时充分考虑了全局相关性，同时依据数据对象的局部邻域相关性选择并组合检测器，能达到较好的异常检测效果。

3 基于多样性的集成单元的构建

集成方法就是结合多个有差异的集成单元，学习数据中不同的特征，以得到比单一方法更具鲁棒性的结果。其中，集成单元的构建尤为关键，需要在保证性能的前提下尽可能地提升多样性。

定义 1（异常集成单元） 异常集成方法中的最小单元由异常检测模型 C 和数据集 X 组成，表示为二元组 $\{C, X\}$。

异常检测模型差异通常体现在两方面：一是模型类型差异，可以同时选择多种不同类型的模型，如基于距离的 KNN 模型、基于密度的 LOF 模型；二是模型超参数差异，可以选择具有多个不同超参数的同类型模型。

数据集通常有两种构造方法：一是样本差异，进行子采样来获取不同的样本域，通常采用的方法是随机子采样法；二是特征差异，在全特征空间下使用子空间搜索方法构造不同的特征子空间。针对高维数据集，不是所有特征都对异常检测有用，通常采用特征提取方法剔除冗余特征，提高异常检测的性能，降低数据维度。LCSE 框架采用模型差异和数据差异构建集成单元。首先根据不同的超参数，生成模型池 $\boldsymbol{C}=\{C_1, C_2, \ldots, C_R\}$，其中 R 表示模型池中的模型个数。训练数据由 n 个 d 维数据组成，表示为 $\boldsymbol{X}_{\text{train}} \in \mathbb{R}^{n\times d}$；测试数据集大小为 m，表示为 $\boldsymbol{X}_{\text{test}} \in \mathbb{R}^{m\times d}$。所有的基础检测模型都由 $\boldsymbol{X}_{\text{train}}$ 的下采样数据子集训练，然后在相同的数据集上得到异常得分。所有模型输出的结果被组合成一个异常得分矩阵 $O(\boldsymbol{X}_{\text{train}})$，如式(1)所示：

$$O(\boldsymbol{X}_{\text{train}})=[C_1(\boldsymbol{X}_{\text{train}}), C_2(\boldsymbol{X}_{\text{train}}), \cdots, C_R(\boldsymbol{X}_{\text{train}})] \tag{1}$$

4 基于集成一致性和邻域相关性的异常检测

4.1 基于集成一致性的异常候选点的筛选

4.1.1 伪真值标签构造

集成方法是训练多个基础模型以解决相同的问题，并将它们结合起来以获得更好的结果的方法，通常分为平行式集成、序列式集成、堆叠式集成。平行式集成在异常检测集成中最为常用，该方法通常考虑的是同类型基础模型。平行式集成相互独立地并行学习这些基础模型，并按照某种确定性的平均过程将它们组合起来；序列式集成通常考虑的也是同类型基础模型，它以一种高度自适应的方法顺序地学习这些基础模型，并按照某种确定性策略将它们组合起来；堆叠式集成通常考虑的是不同类型的基础模型，并行地学习它们，并通过训练一个“元模型”将它们组合起来。

由于序列式集成、堆叠式集成方法需要真值标签作为训练指标，而异常检测中的真值标签较少，故异常检测集成方法

通常采用平行集成。LCSE 是混合模型，在基选择器的选择和组合上使用了平行式集成，在整个生成异常结果的迭代上使用了序列集成，因此需要生成伪真值标签。

LCSE 框架作为异常检测集成框架，由于集成单元没有真值标签(ground truth)，所以需要根据 $\boldsymbol{X}_{\text{test}}$ 构造伪真值标签 *target*。LCSE 是序列集成模型，经过多次迭代直至模型收敛，在第 t 次迭代时 $target_t$ 的值由 $t-1$ 次迭代的模型输出结果 OV_{t-1} 表示，如式(2)所示：

$$target_t = OV_{t-1} \tag{2}$$

第一次迭代时，LCSE 通过将集成单元得分组合，构造集成单元的伪真值标签的初始值 $target_0$，如式(3)所示：

$$target_0 = Combine(O(\boldsymbol{X}_{\text{train}})) \tag{3}$$

其中，*Combine* 代表组合函数，通常取得分组合的最大值或平均值。

4.1.2　异常候选点筛选

异常集成时通常考虑提升多样性和一致性，多样性指基本集成单元的多样性，可以在集成时达到互补的效果；一致性指基本集成单元评价结果的一致性，满足一致性通常可以提升检测准确率。异常集成单元具有多样性，有的表现较好，有的表现较差，在组合这些单元时，直接选择所有单元会造成集成结果性能不好。为了提高集成结果的鲁棒性，集成选择很有必要，Campos 等[3]充分考虑了集成单元的一致性，通过比较集成单元间的相关性，选择相关性高的单元进行组合。

加权皮尔逊相关性(Weighted Pearson Correlation，wPC)广泛应用于异常得分的相似性度量。LCSE 使用加权皮尔逊相关性作为集成单元选择的依据，依据 Schubert 等[2]的方法初始化相关性的权重 W，异常点的权重设置为 $1/2K$，正常点的权重设置为 $1/2(n-K)$，其中 K 为异常点的数量。权重 W 会根据 Boosting 过程更新。

LCSE 根据集成一致性从测试数据 $\boldsymbol{X}_{\text{test}} \in \mathbb{R}^{m\times d}$ 中筛选出异常候选点 $\boldsymbol{X}_{\text{cand}} \in \mathbb{R}^{k\times d}$，将 $\boldsymbol{X}_{\text{test}}$ 输入到训练好的集成单元中，得到异常得分矩阵 $O(\boldsymbol{X}_{\text{test}})$，如式(4)所示：

$$O(\boldsymbol{X}_{\text{test}}) = [C_1(\boldsymbol{X}_{\text{test}}), C_2(\boldsymbol{X}_{\text{test}}), \cdots, C_R(\boldsymbol{X}_{\text{test}})] \tag{4}$$

LCSE 根据皮尔逊相关性选择出最优异常得分矩阵集 $\boldsymbol{O}_{\text{opt}}$，作为输出异常候选点的依据。$O(\boldsymbol{X}_{\text{test}})$ 表示 R 个集成单元异常得分的集合，将每个异常得分向量 $C_i(\boldsymbol{X}_{\text{test}})$ 与 ***target*** 向量的加权皮尔逊相关性排序，与 ***target*** 向量最相似的异常得分向量被选为最优异常得分矩阵 $\boldsymbol{O}_{\text{opt}}$ 中的第一个向量。剩余的异常得分向量根据它们与当前最优异常得分矩阵输出结果的相关性进行排序，迭代地选择出其他优秀的异常得分向量，组成一个互补稳定的最优异常得分矩阵 $\boldsymbol{O}_{\text{opt}}$。当一个异常得分向量 $C_i(\boldsymbol{X}_{\text{test}})$ 能提升最优异常得分矩阵 $\boldsymbol{O}_{\text{opt}}$ 与 ***target*** 的相似性时，该向量被选择，否则该向量被剔除，如式(5)和式(6)所示：

$$Cor_{\text{target},O(\boldsymbol{X}_{\text{test}})} = wPC(\boldsymbol{target}, O(\boldsymbol{X}_{\text{test}})) \tag{5}$$

$$\boldsymbol{O}_{\text{opt}} = \boldsymbol{O}_{\text{opt}} \cup f, \text{if } wpc(\boldsymbol{target}, Combine(\boldsymbol{O}_{\text{opt}}, C_i)) > wpc(\boldsymbol{target}, \boldsymbol{O}_{\text{opt}}) \tag{6}$$

每次迭代有新成员加入最优异常得分矩阵 $\boldsymbol{O}_{\text{opt}}$ 时，执行 Boosting 步骤更新权重 W。具体步骤如下：1)将满足条件的异常得分向量 $C_i(\boldsymbol{X}_{\text{test}})$ 转换为 0－1 标签 *outliers*；2)遍历数组 *outliers*，当迭代到第 j 次，如果满足条件 $target(j)==1 \& outliers(j)==1$，那么 $W(j)=W(j)*drop\_rate$。

迭代完成后，得到最优异常得分矩阵集 $\boldsymbol{O}_{\text{opt}}$，组合异常得分矩阵集得到最优异常得分矩阵，根据异常得分排名筛选出候选异常点 $\boldsymbol{X}_{\text{cand}}$，如式(7)所示：

$$\boldsymbol{X}_{\text{cand}} = prune(Combine(\boldsymbol{X}_{\text{opt}}), \boldsymbol{X}_{\text{test}}) \tag{7}$$

基于集成一致性的异常候选点筛选方法如算法 1 所示。1－3 行表示相关性权重 W 初始化。4－7 行表示最优异常得分矩阵 $\boldsymbol{O}_{\text{opt}}$ 的初始化。8－22 行迭代筛选相关性高的基础集成单元加入最优得分矩阵 $\boldsymbol{O}_{\text{opt}}$，其中 13－15 行计算基础集成单元与最优得分矩阵 $\boldsymbol{O}_{\text{opt}}$ 的加权皮尔逊相关性，满足相关性条件时将该集成单元加入最优得分矩阵 $\boldsymbol{O}_{\text{opt}}$，16－20 行调整相关性权重 W。最后，23－24 行组合最优得分矩阵 $\boldsymbol{O}_{\text{opt}}$ 获得统一异常得分排序 *cur*，根据剔除率 $drop\_rate$ 剪枝得分排序靠后的数据，得到异常候选点。其中 $drop\_rate=90\%$，与数据异常率相对应；在算法 1 中的二分异常得分阈值，根据数据异常率，取异常得分的 10%中位数。

算法 1　基于集成一致性的异常候选点筛选

输入：$\mathbf{X}_{\text{test}}$：＝训练数据集，drop_rate：＝候选点剔除比例，Combine：＝组合函数，C：＝基本模型集合，target：＝伪标签，t：＝异常得分二分阈值，convertBinary：＝异常得分二值化函数

输出：$\mathbf{X}_{\text{cand}}$：＝$\mathbf{X}_{\text{test}}$ 筛选后的异常候选点

1. $O(\mathbf{X}_{\text{test}}) = [C_1(\mathbf{X}_{\text{test}}), C_2(\mathbf{X}_{\text{test}}), \cdots, C_R(\mathbf{X}_{\text{test}})]$
2. W：＝[n]，O_{opt}：＝∅
3. $W = [\text{out} = \frac{1}{2K}, \text{in} = \frac{1}{2(n-K)}]$ // 初始化权重向量 W
4. $Corr_{\text{target},O(\mathbf{X}_{\text{test}})} = wPC(\text{target}, O(\mathbf{X}_{\text{test}}))$ // 计算 target 与每一个 $C(\mathbf{X}_{\text{test}})$ 的相关性
5. $f = \arg\max_{C_i(\mathbf{X}_{\text{test}})} Corr_{\text{target},C_i(\mathbf{X}_{\text{test}})}$ //取相关性最大的 $C(\mathbf{X}_{\text{test}})$
6. $O(\mathbf{X}_{\text{test}})$.pop(f) // 剔除该基础模型 f
7. $\mathbf{X}_{\text{opt}} = \mathbf{X}_{\text{opt}} \cup f$
8. WHILE $O(\mathbf{X}_{\text{test}}) \neq \emptyset$ DO
9. cur：＝Combine(E)
10. $Corr_{\text{cur},O(\mathbf{X}_{\text{test}})} = wPC(\text{curr}, O(\mathbf{X}_{\text{test}}))$ // 计算加权皮尔逊相关性
11. $f = \arg\max_{C_i(\mathbf{X}_{\text{test}})} Cor_{\text{cur},C_i(\mathbf{X}_{\text{test}})}$
12. $O(\mathbf{X}_{\text{test}})$.pop(f)
13. IF wPC(Combine(E∪f))＞wPC(curr，target)THEN
14. 　$\mathbf{O}_{\text{opt}} = \mathbf{O}_{\text{opt}} \cup f$
15. 　outliers＝converBinary(f，t) // 异常标签二值化
16. 　FOR i∈1：size(target) do // 动态调整 W 的权重
17. 　　if target(i)＝＝1 & outliers(i)＝＝1THEN
18. 　　W(i)＝W(i)＊drop_rate
19. 　　END IF
20. 　END FOR
21. END IF
22. END WHILE
23. cur＝Combine(E)
24. $\mathbf{X}_{\text{cand}}$＝prune(cur，drop_rate) // 根据 drop_rate 进行剪枝

4.2　基于邻域相关性的异常得分组合

4.2.1　本地邻域构造

一个候选数据点 X^j_{cand} 的本地邻域 φ_j 是由它 k 近邻的训练个体决定的，形式化表示为：

$$\varphi_j = \{x_i \mid x_i \in \boldsymbol{X}_{\text{train}}, x_i \in kNN^j_{\text{ens}}\} \tag{8}$$

其中，KNN_{ens}表示候选数据点的最近邻居点集。

这里采用类似于 Feature Bagging 的 kNN 的变体，能有效降低涉及 kNN 的维数诅咒。算法过程如下：1）t 组特征被选择出来构造新的特征空间，每组$[d/2,d]$个特征；2）使用欧氏距离度量每组 X^j_{cand}的 k 个最相近的训练数据点；3）出现超过 $t/2$ 次的训练数据点被加入 $O(\varphi_j)$，从而定义出了本地邻域。由于邻域在训练数据点选择标准上是独立的，因此邻域的大小并不是固定的。

本地邻域的参数 k 在这个过程中决定了最近邻居的数量，要避免选择极端的 k 值。k 值越小，模型越聚焦于局部关系，可能会造成不稳定性；k 值越大，模型更强调全局关系，也会造成更高的计算消耗。在存在真值标签的情况下，通常可以通过交叉验证获取最优 k 值，但是在异常检测这种无监督的情况下，该方法无法实现。为此，在多次实验下，推荐设置 $k=0.1n$，即训练样本的 10%，这样可以生成更好的结果。

4.2.2 基于本地邻域的集成单元选择

对于每个候选数据点 X^j_{cand}，它的本地伪真值标签 $target^{\varphi_j}$ 根据式(9)由本地邻域 φ_j 内的数据生成。

$$target^{\varphi_j}=\{target_{x_i}\mid x_i\in\varphi_j\} \tag{9}$$

类似地，本地训练异常得分 $O(\varphi_j)$可以由预计算的训练异常得分矩阵 $O(\boldsymbol{X}_{train})$得到，如式(10)所示：

$$O(\varphi_j)=[C_1(\varphi_j),\cdots,C_r(\varphi_j)]\in\mathbb{R}^{|\varphi_j|\times R} \tag{10}$$

其中，$|\varphi_j|$表示本地邻域 φ_j 的基。每个候选数据点需要计算本地邻域，但是本地异常得分和伪标签可以由第 3 节中集成单元构建的计算结果获取。

为了评估本地邻域中集成单元的性能，有监督学习用标签来度量基本分类器的准确性，而 LCSE 只能通过伪真值标签和集成单元得分的相似度来度量。这种区别是由于在无监督的异常值挖掘中缺乏真值标签。虽然将伪异常得分转为标签值是可行的，但是定义一个准确区分度的阈值具有极大的挑战性。此外，由于异常检测任务中通常存在不平衡的样本，使用相似性度量模型的绝对准确度更为稳定。因此，LCSE 使用伪真值标签 $target^{\varphi_j}$ 和本地集成单元得分 $C_r(X^{\varphi_j}_{train})$的皮尔逊相关系数来评估每个集成单元的局部性能。根据皮尔逊相关性选出模型组 C^*，作为候选数据点 X^j_{cand}的最优本地模型组，然后使用 AOM(Average of Maximum)或 MOA(Maximum of Average)的方式组合最优本地模型组，得到异常得分 $score_j$。最后，构造异常得分向量 $\boldsymbol{OV}$(Outlier Vector)。具体地，将异常候选点 $\boldsymbol{X}_{cand}$的异常得分归一化在区间$[0,1]$内，不属于异常候选集合的点的异常得分视为 0。数据 X_i 的异常得分 OV_i 的取值如式(11)所示：

$$OV_i=\begin{cases}normalize(score_i), & X_i\in\boldsymbol{X}_{cand}\\ 0, & X_i\notin\boldsymbol{X}_{cand}\end{cases} \tag{11}$$

其中，$normalize$ 为归一化函数。

基于邻域相关性的异常得分组合的具体算法如算法 2 所示。

算法 2 基于邻域相关性的异常得分组合

输入：$\mathbf{X}_{cand}$：=异常候选点集合，g：=最优模型组长度，target：=伪真值标签，AOM：=最大值平均函数

输出：scores：=异常得分向量

1. FOR x_j IN $\mathbf{X}_{cand}$
2. $\varphi_j=\{x_i\mid x_i\in\mathbf{X}_{train},x_i\in kNN^j_{ens}\}$ // 针对构造本地邻域
3. $target^{\varphi_j}=\{target_{x_i}\mid x_i\in\varphi_j\}$ // 计算邻域上的 target
4. $O(\varphi_j)=[C_1(\varphi_j),\cdots,C_r(\varphi_j)]\in\mathbb{R}^{|\varphi_j|\times R}$ // 计算邻域上 φ_j 的基本模型输出 O
5. $Corr_{target,O(\varphi_j)}=PC(target,O(\varphi_j))$ // 计算 target 和 O 的皮尔逊相关性
6. Sort $O(\varphi_j)$By Corr
7. $C^*(\varphi_j)=Top\_g(O(\varphi_j))$ // 选择相关性最高的 g 个 O 组成模型组
8. $score_j=AOM(C^*(\varphi_j))$ // 使用 AOM 组合模型组得分
9. END FOR

4.2.3 迭代计算

在 4.1.1 节构造伪真值标签时，$target$ 为初始化参数，最终异常得分向量 $\boldsymbol{OV}$ 受到 $target$ 影响。为了得到稳定的异常检测表现，LCSE 致力于生成趋于固定的异常得分向量。于是，LCSE 迭代计算异常标签。评判一个向量的收敛性通常采用向量范数。在第 t 次迭代，基于 p 次范数的稳定标准可以定义如下：

$$\lim_{t\to+\infty}\boldsymbol{OV}_{t+1}-\boldsymbol{OV}_t\leqslant\varepsilon \tag{12}$$

其中，$p\geqslant1$ 且 ε 是极小值。

迭代的步骤如下：1）在 t 次迭代时，得到异常得分向量 $\boldsymbol{OV}_t$。2）根据式(12)比较第 t 次和第 $t-1$ 次迭代的异常得分向量，判断是否达到稳定标准。3）如果达到稳定标准，则直接输出结果 $\boldsymbol{OV}_{final}$；如果未达到标准，将 $\boldsymbol{OV}_t$ 作为下一次迭代的伪真值标签，继续执行迭代，如式(13)所示：

$$target_{t+1}=\boldsymbol{OV}_t \tag{13}$$

至此，LCSE 所有步骤执行完毕。LCSE 是一个序列模型，先考虑全局一致性，筛选出异常候选点，再通过邻域相关性，进一步计算出异常点，最后迭代以上步骤直至收敛。

5 实验验证

5.1 实验准备

实验所用公开数据集共 6 个，来自 ODDS[22] 和 DAMI[23]，如表 1 所列。所选数据集均为异常检测领域的常用数据集，有一定的代表性。在所有实验中，将 60%的数据作为训练数据，剩余的 40%用作测试评估。

表 1 异常检测数据集

Table 1 Outlier Detection data set

数据集	数据量	维度	异常数	异常率/%
Internet-Ads	3264	1775	177	9.97
Speech	3686	400	61	1.65
Musk	3062	166	97	3.2
Mnist	7600	100	700	9.2
Cardio	1831	21	176	9.61
Stamps	340	9	31	9.12

本文提出的集成方法 LCSE 考虑了集成一致性和邻域相关性，组合单一模型进行实验。实验对比的基准方法分别选择传统方法和集成方法。

传统方法：1）基于距离的标志性异常检测算法 kNN；2）基于密度的常用算法 LOF[20]。

集成方法：1）常用的平行异常集成算法 iForest[11]；2）基于邻域的异常集成算法 LSCP_AOM[4]。

神经网络方法：无监督自编码器 Auto-Encoder[6]。

(1)实验设置

实验中为了保证集成方法异常评估的一致性，基础模型统一选用 50 个 LOF 模型。基础模型的多样性体现在超参数的不同，每个 LOF 的邻居数量(MinPts)在[5，200]范围内随机选取。$drop\_rate=90\%$，与数据异常率相对应。二分异常得分阈值 t，根据数据异常率，取异常得分的 10% 中位数；最优模型长度可选择[5，10]之间的整数，过小则对集成平均没有意义，过大则会造成计算资源的浪费。

(2)异常检测准确度

由于单次实验存在随机性，实验采用 10 次独立实验的平均接受者工作特征曲线下面积 AUC 作为评判依据。该指标广泛用于异常检测研究，*AUC* 值越高代表模型的准确率越高。

(3)抗噪性

选用 Musk 数据集，采用加入一定比例随机噪声的方式构造噪声数据集，同时观察不同方法在噪声情况下的表现。

5.2　实验结果分析

(1)异常检测性能分析

针对不同数据集的异常检测 *AUC* 的结果如表 2 所列，其中加粗显示的数值为算法中表现排名前两位的结果。

表 2　异常检测算法的 *AUC*

Table 2　*AUC* of outlier detection algorithms

数据集	算法					
	KNN	LOF	Auto-Encoder	iForest	LSCP_AOM	LCSE
Internet-Ads	**0.752**	**0.761**	0.701	0.626	0.720	0.727
Speech	0.507	**0.523**	0.505	0.502	0.517	**0.522**
Musk	0.624	0.639	**0.989**	0.987	0.988	**0.992**
Mnist	0.847	0.718	0.851	0.813	**0.862**	**0.869**
Cardio	0.727	0.582	**0.952**	0.924	0.926	**0.935**
Stamps	0.582	0.564	0.624	**0.652**	0.606	**0.675**
Average	0.673	0.631	0.770	0.751	0.769	**0.787**

从表 2 可以看出，LCSE 在多数数据集上表现都比较好，尤其在 Musk 数据集上表现最好，*AUC* 高达 0.992。相比传统方法 KNN 和 LOF，LCSE 的异常检测准确率平均提升了 20.7%，因为 LCSE 集成了多个基础模型，能有效减小模型的方差和偏差，得到的结果鲁棒性较好。

LCSE 相比其他集成方法 LSCP_AOM 和 iForest，性能平均提升了 3.6%。基于邻域相关性的平行集成方法 LSCP_AOM 在所有实验数据集上的性能均不如 LCSE。因为 LSCP_AOM 仅考虑了数据的邻域特性，在数据密度相差较大的数据集上表现较好，而 LCSE 同时考虑了集成一致性和邻域相关性，能得到更准确的结果。LCSE 与 iForest 在多数数据集上表现都很好，但是在稍高维的 Musk 和 Speech 数据集上，iForest 明显性能不足，LCSE 性能虽有下降，但表现更好。因为高维数据中含有较多噪声特征，iForest 容易产生过拟合。神经网络方法 Auto-Encoder 的整体性能较好，但是在大部分数据上性能不如 LCSE。因为神经网络在无监督问题上受到限制，且需要较大的样本量用于训练。

从表 2 也可以看出，虽然 LCSE 在多数中低维数据集中表现不错，但是在高维数据集上性能有明显降低，因为高维空间含有许多不相关和冗余的特征，LCSE 并没有对特征进行筛选，所以在高维空间表现不佳。

(2)抗噪性分析

本实验从稳定性方面考虑，通过加入一定比例的随机噪声，模拟噪声影响下异常检测的情况，各算法表现如图 2 所示。从实验结果可以看出，本文基于集成一致性和邻域相关性的集成异常检测方法 LCSE 具有较好的抗噪性，随着噪声比例的升高，性能较为稳定。传统的基于密度的算法 LOF 和基于距离的算法 KNN 初始表现明显不如集成算法，且随着噪声比例的上升，异常检测性能越来越差，这是因为集成算法通过数据采样等方法构造不同的数据空间，选择并结合基础模型输出一致性结果，能有效减小方差和偏差。基于邻域相关性的异常检测算法 LSCP_AOM 虽然初始性能较好，但是随着噪声比例的增大，性能下降极快，在噪声比例为 25% 以上时，性能基本与传统方法一致，因为 LSCP_AOM 仅关注局部邻域特征，没有进行全局筛选，导致噪声对邻域的影响被放大。LCSE，iForest 和 Auto-Encoder 都有较好的抗噪能力，随着噪声比例的上升，性能下降不明显，但是相比之下 LCSE 性能下降得更缓慢，因为 LCSE 在构造集成单元时为保证多样性对数据进行下采样，且依据全局集成一致性对异常点进行了初步筛选，最后依据邻域相关性进行评判，从而保证了算法的稳定性。

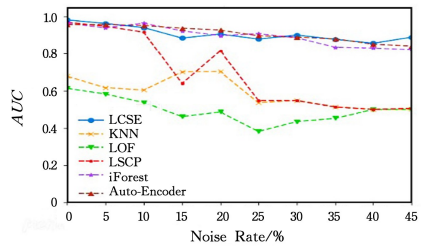

图 2　噪声下异常检测的 *AUC*

Fig. 2　Abnormal detection *AUC* under noise

结束语　本文分析了集成方法在异常检测中能有效减小偏差和方差，性能更稳定，并进一步提出了基于集成一致性和邻域相关性的集成异常检测方法 LCSE。该方法首先基于多样性构造集成单元，然后依据全局集成一致性筛选出异常得分一致意见下的异常候选点，最后依据候选点的邻域相关性选择并结合基础模型，得到最终异常检测结果。由于初始伪真值标签具有不确定性，因此采用迭代计算的方法进一步训练模型，得到稳定的结果。

本文在多个数据集上对各种对比算法进行实验，可以看出 LCSE 在多数数据集上表现都比较好，其 *AUC* 相比传统方法 LOF 和 KNN 平均提升了 20.7%；与集成算法 LSCP_AOM 和 iForest 相比，*AUC* 平均提升了 3.6%；并且在多数数据集上其性能优于神经网络方法 Auto-Encoder。实验结果验证了 LCSE 在中低维数据集上异常检测性能较好。同时 LCSE 有较好的稳定性，在噪声情况下能保持稳定的性能。

虽然 LCSE 在多数中低维数据集中表现不错，但是在高维数据集上性能明显降低，因为高维空间含有许多不相关和冗余的特征，LCSE 并没有对特征进行筛选，所以在高维空间表现不佳。接下来将进行高维数据特征选择的研究。

参 考 文 献

[1] AGGARWALC C. Outlier analysis [C]// Data mining. Cham: Springer, 2015: 237-263.

[2] SCHUBERT E, WOJDANOWSKI R, ZIMEK A, et al. On evaluation of outlier rankings and outlier scores[C]// Proceedings of the 2012 SIAM International Conference on Data Mining. Philadelphia: SIAM, 2012: 1047-1058.

[3] CAMPOS G O, ZIMEK A, MEIRA W. An unsupervised boosting strategy for outlier detection ensembles[C]// Pacific-Asia Conference on Knowledge Discovery and Data Mining. Cham: Springer, 2018: 564-576.

[4] ZHAO Y, NASRULLAH Z, HRYNIEWICKI M K, et al. LSCP: Locally selective combination in parallel outlier ensembles[C]// Proceedings of the 2019 SIAM International Conference on Data Mining. Philadelphia: SIAM, 2019: 585-593.

[5] CHEN Y P, YU L, CHEN H. Traffic Anomaly Detection Based on Wavelet Neural Network and ARMA Model in Big Data Environment[J]. Journal of Chongqing Institute of Technology (Natural Science), 2019, 33(10): 149-154.

[6] CHEN J, SATHE S, AGGARWAL C, et al. Outlier detection with autoencoder ensembles[C]// Proceedings of the 2017 SIAM International Conference on Data Mining. Philadelphia: SIAM, 2017: 90-98.

[7] XING H J, HAO Z. Novelty Detection Method Based on Global and Local Discriminative Adversarial Autoencoder[J]. Computer Science, 2021, 48(6): 202-209.

[8] CHALAPATHY R, CHAWLA S. Deep learning for anomaly detection: A survey[J]. arXiv: 1901. 03407, 2019.

[9] LAZAREVIC A, KUMAR V. Feature bagging for outlier detection[C]// Proceedings of the eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining. New York: ACM, 2005: 157-166.

[10] RAYANA S, AKOGLU L. Less is more: building selective anomaly ensembles[J]. ACM TKDD, 2016, 10(4): 1-33.

[11] LIU F T, TING K M, ZHOU Z H. Isolation forest[C]// 2008 Eighth IEEE International Conference on Data Mining. Piscataway: IEEE, 2008: 413-422.

[12] RAYANA S, ZHONG W, AKOGLU L. Sequential ensemble learning for outlier detection: A bias-variance perspective[C]// 2016 IEEE 16th International Conference on Data Mining (ICDM). Piscataway: IEEE, 2016: 1167-1172.

[13] GAO J, TAN P N. Converting output scores from outlier detection algorithms into probability estimates[C]// Sixth International Conference on Data Mining (ICDM'06). Piscataway: IEEE, 2006: 212-221.

[14] ZIMEK A, GAUDET M, CAMPELLO R J G B, et al. Subsampling for efficient and effective unsupervised outlier detection ensembles[C]// Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM, 2013: 428-436.

[15] ZIMEK A, CAMPELLO R J G B, SANDER J. Data perturbation for outlier detection ensembles[C]// Proceedings of the 26th International Conference on Scientific and Statistical Database Management. New York: ACM, 2014: 1-12.

[16] PASILLAS-DÍAZ J R, RATTÉ S. Bagged subspaces for unsupervised outlier detection[J]. Computational Intelligence, 2017, 33(3): 507-523.

[17] NGUYEN H V, ANG H H, GOPALKRISHNAN V. Mining outliers with ensemble of heterogeneous detectors on random subspaces[C]// International Conference on Database Systems for Advanced Applications. Berlin, Heidelberg: Springer, 2010: 368-383.

[18] CAMPOS G O, ZIMEK A, MEIRA W. An unsupervised boosting strategy for outlier detection ensembles[C]// Pacific-Asia Conference on Knowledge Discovery and Data Mining. Cham: Springer, 2018: 564-576.

[19] VAN STEIN B, VAN LEEUWEN M, BÄCK T. Local subspace-based outlier detection using global neighborhoods[C]// 2016 IEEE International Conference on Big Data (Big Data). Piscataway: IEEE, 2016: 1136-1142.

[20] BREUNIG M M, KRIEGEL H P, NG R T, et al. LOF: identifying density-based local outliers[C]// Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data. New York: ACM, 2000: 93-104.

[21] KRIEGEL H P, KRÖGER P, SCHUBERT E, et al. LoOP: local outlier probabilities[C]// Proceedings of the 18th ACM Conference on Information and Knowledge Management. New York: ACM, 2009: 1649-1652.

[22] RAYANA S. ODDS Library[DB/OL]. http://odds. cs. stonybrook. edu, 2016/2020-03-15.

[23] CAMPOS G O, ZIMEK A, SANDER J, et al. On the evaluation of unsupervised outlier detection: measures, datasets, and an empirical study[J]. Data Mining and Knowledge Discovery, 2016, 30(4): 891-927.

LIU Yi, born in 1996, postgraduate. Her main research interests include distributed computing, IoT and edge intelligence computing.

MAO Ying-chi, born in 1976, Ph.D, professor, is a senior member of China Computer Federation. Her main research interests include distributed computing and parallel processing, IoT, and edge intelligence computing.

(责任编辑:柯颖)