计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.190600060

基于海林格距离和SMOTE的多类不平衡学习算法

董明刚[1,2]　姜振龙[1]　敬　超[1,2]
1 桂林理工大学信息科学与工程学院　广西 桂林 541004
2 广西嵌入式技术与智能系统重点实验室　广西 桂林 541004
(d2015mg@qq.com)

摘　要　数据不平衡现象在现实生活中普遍存在。在处理不平衡数据时,传统的机器学习算法难以达到令人满意的效果。少数类样本合成上采样技术(Synthetic Minority Oversampling Technique,SMOTE)是一种有效的方法,但在多类不平衡数据中,边界点分布错乱和类别分布不连续变得更加复杂,导致合成的样本点会侵入其他类别区域,造成数据过泛化。鉴于基于海林格距离的决策树已被证明对不平衡数据具有不敏感性,文中结合海林格距离和SMOTE,提出了一种基于海林格距离和SMOTE的上采样算法(Based on Hellinger Distance and SMOTE Oversampling Algorithm,HDSMOTE)。首先,建立基于海林格距离的采样方向选择策略,通过比较少数类样本点的局部近邻域内的海林格距离的大小,来引导合成样本点的方向。其次,设计了基于海林格距离的采样质量评估策略,以免合成的样本点侵入其他类别的区域,降低过泛化的风险。最后,采用7种代表性的上采样算法和HDSMOTE算法对15个多类不平衡数据集进行预处理,使用决策树的分类器进行分类,以Precision,Recall,F-measure,G-mean和MAUC作为评价标准对各算法的性能进行评价。实验结果表明,相比于对比算法,HDSMOTE算法在以上评价标准上均有所提升:在Precision上最高提升了17.07%,在Recall上最高提升了21.74%,在F-measure上最高提升了19.63%,在G-mean上最高提升了16.37%,在MAUC上最高提升了8.51%。HDSMOTE相对于7种代表性的上采样方法,在处理多类不平衡数据时有更好的分类效果。

关键词:SMOTE;上采样;海林格距离;多类不平衡学习;分类
中图法分类号　TP311

Multi-class Imbalanced Learning Algorithm Based on Hellinger Distance and SMOTE Algorithm

DONG Ming-gang[1,2],JIANG Zhen-long[1] and JING Chao[1,2]
1 College of Information Science and Engineering,Guilin University of Technology,Guilin,Guangxi 541004,China
2 Guangxi Key Laboratory of Embedded Technology and Intelligent System,Guilin,Guangxi 541004,China

Abstract　Imbalanced data is common in real life. Traditional machine learning algorithms are difficult to achieve satisfied results on imbalanced data. The synthetic minority oversampling technique (SMOTE) is an efficient method to handle this problem. However,in multi-class imbalanced data,disordered distribution of boundary sample and discontinuous class distribution become more complicated,and the synthetic samples may invade other classes area,leading to over-generalization. In order to solve this issue,considering the algorithm based on Hellinger distance decision tree has been proved to be insensitive to imbalanced data,combining with Hellinger distance and SMOTE, this paper proposed an oversampling method SMOTE with Hellinger distance (HDSMOTE). Firstly,a sampling direction selection strategy was presented based on Hellinger distances of local neighborhood area,which can guide the direction of the synthesized sample. Secondly,a sampling quality evaluation strategy based on Hellinger distance was designed to avoid the synthesized sample into other classes,which can reduce the risk of over-generalization. Finally, to demonstrate the performance of HDSMOTE,15 multi-class imbalanced data sets were preprocessed by 7 representative oversampling algorithms and HDSMOTE algorithm,and were classified with C4.5 decision tree. Precision,Recall,F-measure,G-mean and MAUC are employed as the evaluation standards. Compared with competitive oversampling methods,the experimental results show that the HDSMOTE algorithm has improved in the these evaluation standards. It is increased by 17.07% in Precision, 21.74% in Recall,19.63% in F-measure,16.37% in G-mean,and 8.51% in MAUC. HDSMOTE has better classification performance than the seven representative oversampling methods on multi-class imbalanced data.

到稿日期:2019-06-12　返修日期:2019-08-17　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。
基金项目:国家自然科学基金(61563012,61802085);广西自然科学基金(2014GXNSFAA118371,2015GXNSFBA139260);广西嵌入式技术与智能系统重点实验室基金(2018A-04)
This work supported by the National Natural Science Foundation of China (61563012,61802085),Natural Science Foundation of Guangxi, China (2014GXNSFAA118371,2015GXNSFBA139260) and Guangxi Key Laboratory of Embedded Technology and Intelligent System Foundation (2018A-04).
通信作者:敬超(jingchao@glut.edu.cn)

Keywords SMOTE，Oversampling，Hellinger distance，Multi-class imbalanced learning，Classification

1 引言

随着计算机技术、网络技术、通信技术和存储技术的发展，互联网等领域存在大量多类不平衡数据，如何从这些多类不平衡数据中获取有价值的信息，已经成为当下的研究热点[1-5]。传统的分类方法通常假设数据类别分布均衡，且错分代价是相同的，对于二类不平衡数据，这样分类会导致少数类的样本被划分到多数类中。在多类不平衡数据中，边界点分布错乱和类别分布不连续变得更加复杂，上述问题会愈加明显。不平衡学习是指从这些类别分布不均衡的数据中进行学习[1]，代表性方法有代价敏感、主动学习、集成学习和采样方法等。其中，采样方法中的上采样技术是对少数类样本进行采样，使得少数类的样本数量与多数类的样本数量达到平衡。上采样技术备受国内外学者的关注，取得了很多成果，且被广泛应用于生物医疗、化学、金融、信息安全、工业、计算机视觉等领域[6-9]。

最简单的上采样技术是随机上采样（Random Oversampling，ROS）。ROS 随机选取少数类样本中的样本点进行复制，以此来增加少数类样本点的数量。但是，这种操作会使得少数类样本重复，从而造成数据过拟合。随后，Chawla 等提出了 SMOTE 算法[10]，对于少数类的样本，通过随机选择同类近邻样本来生成无重复的少数类样本。SMOTE 算法在一定程度上缓解了数据的过拟合；但是，在多类不平衡数据中，由于边界点分布错乱和类别分布不连续变得更加复杂，SMOTE 会造成数据过泛化。He 等提出了 ADASYN 算法[11]，其根据样本的学习难易级别来产生不同数量的新样本，从而解决不平衡学习问题。Barua 等提出了 MWMOTE 算法[12]，其通过确定难以学习的少数类样本，并根据它们与最近的多数类样本的欧氏距离来分配权重，应用聚类方法，在此基础上为少数类样本合成样本，避免合成错误的样本点。为了避免合成样本过程中偏向于多数类，Nekooeimehr 等提出 A-SUWO 算法[13]，其通过考虑更接近边界线的每个子集中的少数类样本来识别难以学习的样本点，并在此基础上合成样本点。Puntumapon 等提出的 Cluster-SMOTE 算法[14]，首先确定过泛化和过拟合之间的少数类区域，以聚类的形式识别少数类区域，然后将这些聚类合并成更为广泛的聚类，以解决过泛化问题。Han 等提出的 Borderline-SMOTE 算法[15]，仅为那些“更接近”边界的少数类样本点合成新的样本，解决了 SMOTE 算法对所有少数类样本进行采样而导致过拟合的问题。采用这些算法解决多类问题时，通常将多类不平衡学习问题分解为若干个二类不平衡学习问题。常用的分解方式是 One-against-all（OAA）[16]，即将 t 个类别的多类不平衡问题分解成 t 个二类不平衡问题，将当前类别视为正类，其他类别视为负类。但在多类不平衡数据中，边界点分布错乱和类别分布不连续变得更加复杂，致使这些算法合成的样本点会侵入其他类别区域，造成过泛化。除此之外，这些算法还有一个共同的缺点，即在解决多类不平衡学习问题时没有考虑整体数据[17]。

最近，研究人员也提出了一些针对多类不平衡学习的上采样算法。Zhu 等提出了 SMOM 算法[17]，其通过聚类为每个小类样本计算合适的权重，然后通过权重来控制合成样本点的方向和范围，有效地解决了过泛化问题。但是，该方法需要用户设置很多的参数来平衡泛化的拟合，实用性受到限制。Abdi 等提出了 MDO 算法[18]，不同于传统的 SMOTE 算法，MDO 是一种基于马氏距离的上采样方法，通过保留少数类样本的协方差结构，并基于协方差结构来合成新的样本点，能够保证合成样本前后的分布基本不变。随后，针对 MDO 不能处理混合变量的问题，Yang 等提出了 AMDO 算法[19]，其通过 HVDM 技术处理混合变量，有效地解决了混合变量问题。但是，以上二者仅考虑了密集区域的样本点来合成新的样本，未考虑整体数据，有可能造成过拟合。

基于海林格距离（Hellinger Distance）的决策树已被证明对不平衡数据具有不敏感性，其思路是将海林格距离作为决策树分裂的标准[20-21]。鉴于此，我们将海林格距离应用到上采样过程中，用来指导采样方向和评估合成样本点的质量。

在多类不平衡学习中，针对传统的上采样方法存在的过泛化和未考虑整体数据合成样本点的问题，本文提出了一种基于海林格距离和 SMOTE 的上采样算法——HDSMOTE。首先，建立基于海林格距离的采样方向选择策略，通过比较少数类别样本点局部近邻域内的海林格距离的大小，来引导合成样本点的方向。其次，建立基于海林格距离的采样质量评估策略，对合成的样本点进行评估，避免合成的样本点侵入其他类别的区域。最后，采用 7 种代表性的上采样算法和 HDSMOTE 算法对 UCI[22] 和 KEEL[23] 上的 15 个多类不平衡数据集进行预处理，使用基于 C4.5 决策树的 RIPPER 分类器进行分类，实验结果证明 HDSMOTE 算法具有更好的分类效果。

2 相关技术

2.1 SMOTE

SMOTE 算法的主要思想是通过人工合成少数类样本来改变原始样本的分布[10]。随机选择同类近邻样本，在相距较近的少数类样本之间进行线性插值。合成样本点的公式为：

$$x_{\text{syn}} = x_i + alpha * (x_j - x_i) \tag{1}$$

其中，x_{syn} 是人工合成的新的样本点，x_i 是少数类中一个随机的样本点，x_j 是 x_i 的近邻中的一个随机样本点；* 表示逐个属性相乘；$alpha$ 是常量，取值范围是（0，1）。

2.2 海林格距离

海林格距离是一种概率分布相似程度的度量，能够反映数据分布的相似程度[20-21]。基于海林格距离的决策树对不平衡数据具有不敏感性，不会随着数据的不平衡率的改变而发生质的改变[21]。基于海林格距离的决策树将海林格距离作为决策树的分裂标准。鉴于此，我们将当前少数类视为一个类别，将其他所有类别作为一个超类，视为另一个类别。计算当前少数类与超类的海林格距离，其反映了多类不平衡数据的分布情况。

在可度量的空间上，海林格距离被定义为：

$$d_H(X_+, X_-) = \sqrt{\sum_{j=1}^{p}\left(\sqrt{\frac{|X_{+j}|}{X_+}} - \sqrt{\frac{|X_{-j}|}{X_-}}\right)^2} \tag{2}$$

其中，X_+（X_-）代表正类（负类）样本的总数量，X_{+j}（X_{-j}）代表特征中第 j 个（有 p 个属性）属性上的正类（负类）的数量，p 表示属性的数量。

海林格距离具有以下特性：1）$d_H(X_+, X_-)$ 的界限为 $[0, \sqrt{2}]$；2）$d_H(X_+, X_-)$ 是对称且非负的。

3 HDSOMTE

为方便算法的描述，先进行如下定义。

定义 1（近邻域） 将数据整体集合记为 D，对于某一类别样本 $S \in D$ 和任意 $x_i \in S(i=1,2,3,\cdots,n_s)$，在 D 中找到 x_i 的 k_1 个近邻。这 k_1 个样本组成的区域为 x_i 的近邻域。

定义 2（混淆点） 对于任意 $x_i \in S(i=1,2,3,\cdots,n_s)$，计算其近邻域，并将其记为 A，再计算 A 中属于 S 的样本点数量，若数量小于或等于 1，则 x_i 为混淆点。

定义 3（局部近邻域） 将某一类别样本点集合记为 S，对于任意 $x_i \in S(i=1,2,3,\cdots,n_s)$，在 S 中找到 x_i 的 k_2 个近邻，并将其记为 knn_i。计算 x_i 和 knn_i 中所有样本点的欧氏距离，将其中最大的距离记为 d。以 x_i 为圆心、以 d 为半径做圆，将圆内所有样本点组成的区域记为样本点 x_i 的局部近邻域。

3.1 基于海林格距离的采样方向选择策略

海林格距离反映数据分布的相似程度，通过计算样本点局部近邻域内的海林格距离，可以得到局部近邻域内的样本分布相似程度。对于机器学习算法而言，若不同类别的样本的相似程度高，则说明样本点不易被学习和分类；若样本相似程度低，则说明样本点易于被学习和分类。在合成样本点的过程中，引导样本点向其周围相似程度低的近邻点学习，合适地加强数据的泛化性。以两个样本点为例，如图 1 中第 a 步所示，样本点 1 和样本点 2 的局部近邻域分别为 B 和 C。若区域 B 的海林格距离比区域 C 的海林格距离大，则 C 向 B 学习，反之 B 向 C 学习。基于海林格距离的采样方向选择策略的具体步骤是：对于样本点 $x_i \in S_m'(i=1,2,3,\cdots,n_s', S_m'$ 为删除混淆点之后的当前少数类），比较 H_i 和 H_{knni} 的大小，将 $H_{knni} > H_i$ 对应的样本点记为 G；若 $H_{knni} \leqslant H_i$，则对 H_{knni} 进行降序排列，选择 k 个最大的值，并将对应的样本点记为 G。从 G 中随机选取样本点并将其记为 x_j，然后通过 SMOTE 技术合成新的样本点。其中，H_i 是 x_i 的海林格距离，H_{knni} 是 x_i 的 k_2 个近邻对应的海林格距离。基于海林格距离的采样方向选择策略（HDCS）如算法 1 所示。

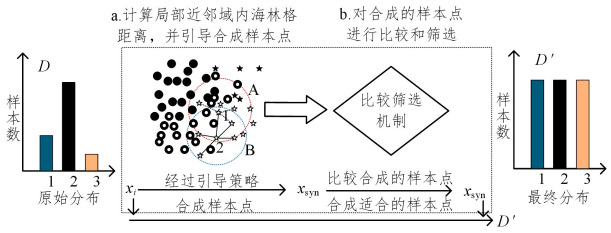


图 1 HDSMOTE 示意图

Fig. 1 HDSMOTE schematic

算法 1 HDCS

输入：样本点 x_i；样本点 x_i 对应的 K 近邻 knn_i；样本点 x_i 及其对应 K 近邻的局部海林格距离 H_i 和 H_{knni}

输出：合成的样本点 x_{syn}；x_j 的集合 G

Step 1 G=∅；

Step 2 if $H_{knni} > H_i$

Step 3 G←将 knn_i 中对应的点记为 G；

Step 4 else

Step 5 G←对 H_{knni} 进行降序排序，选择 k(k≥3)个最大的值，将对应的样本点记为 G；

Step 6 end

Step 7 x_j←从 G 中随机选取一个样本点；

Step 8 $x_{syn} = x_i + alpha * (x_j - x_i)$。

3.2 基于海林格距离的采样质量评估策略

多类不平衡数据中类别分布不连续和边界点分布复杂的特性，导致以 SMOTE 为基础合成的新样本可能会侵入其他类别的样本区域。海林格距离反映概率分布的相似程度，若合成的样本点侵入其他类别区域，则会导致当前少数类样本的特征属性与其他类别更加相似，使得海林格距离变小。为避免合成的样本点侵入其他类别的样本区域，为每个属性计算海林格距离，通过比较合成样本点之后的所有属性的海林格距离与未合成样本点时所有属性的海林格距离，来判断合成的样本点是否侵入其他类别的区域。在基于海林格距离的采样质量评估策略的过程中，不仅考虑了局部近邻域内的数据分布情况，还考虑了整体数据的分布情况，计算当前少数类与超类的海林格距离，并将其记为 H_t，其反映了整体数据的分布情况，将被作为全局最低衡量标准。基于海林格距离的采样质量评估策略的步骤为：计算 x_i 的局部近邻域内的所有属性的海林格距离，并将其记为 H_1；将合成的样本点加入到 x_i 的局部近邻域中，计算所有属性的海林格距离，并将其记为 H_2。比较 H_1，H_2 和 H_t 的大小，若 $H_2 \geqslant H_1$ 且 $H_2 \geqslant H_t$，则保留合成的样本点，否则重新合成样本点，由于个别样本点周围的分布非常复杂，重复循环可能依然找不到合适的样本点，为避免程序一直循环，定义最多循环 R 次，并保留最佳的合成样本点。Evaluation 算法的流程图如图 2 所示，伪代码如算法 2 所示。

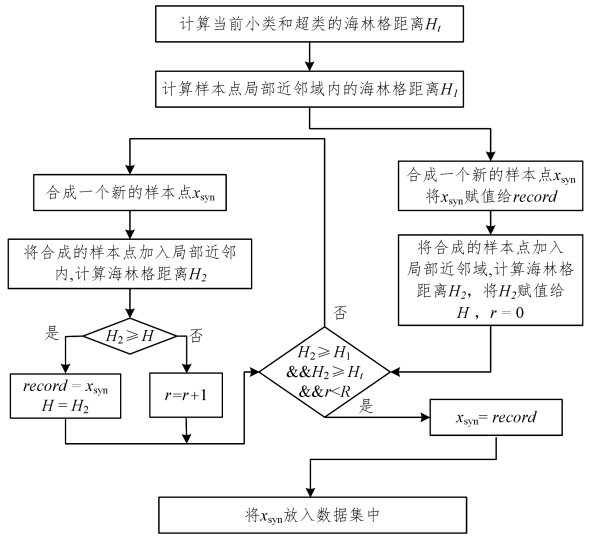


图 2 Evaluation 流程图

Fig. 2 Flow chart of Evaluatio

算法 2 Evaluation

输入：样本点 x_i；x_j 的集合 G；x_i 对应的局部近邻域 D_i'；全局海林格距离 H_t；x_i 对应的局部近邻域内的海林格距离 H_i；合成的样本点 x_{syn}

输出：合成样本点 x_{syn}

Step 1 H_1←H_i；
Step 2 H_2←将 x_{syn}置于 D_i' 中计算海林格距离；
Step 3 r=0；
Step 4 record =x_{syn}；
Step 5 while (H_2<H_1 && H_2<H_t) && r<R
Step 6 x_j←从 G 中随机选取一个样本点；
Step 7 x_{syn}=x_i+alpha*(x_j−x_i)；
Step 8 H_2←x_{syn}置于 D_i'中计算海林格距离；
Step 9 record←比较 x_{syn}和 record 的海林格距离，保留最优的结果；
Step 10 r = r+1；
Step 11 end
Step 12 x_{syn}=record。

3.3 HDSMOTE 算法

HDSMOTE 算法综合考虑了数据的整体性和局部性。通过计算整体数据的海林格距离，反映数据的全局分布情况；通过计算样本点局部近邻域内的海林格距离，反映样本点周围的分布情况。结合数据的全部分布情况和局部分布情况，在海林格距离的采样方向选择策略和采样质量评估策略的基础上提出了 HDSMOTE 算法。其具体步骤为：1）对当前少数类 S_m 样本点进行选择，删除混淆点，形成新的小类 S_m'；2）计算当前少数类 S_m 与超类的海林格距离 H_t；3）计算 S_m'中所有样本点的局部近邻域内的海林格距离；4）通过基于海林格距离的采样方向选择策略合成样本点 x_{syn}；5）用基于海林格距离的采样质量评估策略找到合适的 x_{syn}；6）当前少数类采样完成后，将采样后的样本放入原始数据 D 中；7）重复步骤 1）—步骤 6），直至遍历完所有少数类。HDSMOTE 的示意图如图 1 所示，伪代码如算法 3 所示。算法 3 中，n_s 表示当前少数类的样本数量，n_s'表示删除混淆点后的少数类的样本数量，n 代表要合成的当前小类的样本数量。

算法 3 HDSMOTE

输入：数据集 D；所有小类的集合小类 S_T，T=1，2，3，…，m；K 近邻 k_1 和 k_2
输出：平衡数据集 D′
Step 1 for j=1 to m
Step 2 S_m←从 S_T取出对应的小类 S_j；
Step 3 for i=1 to n_s
Step 4 knn←x_i在 D 中的 k_1 个近邻；
Step 5 number←knn 中小类的数量；
Step 6 if number>1
Step 7 S_m'←将 x_i置于 S_m' 中；
Step 8 end
Step 9 end
Step 10 H_t←在 D 中计算少数类与超类海林格距离；
Step 11 for i=1 to n_s'
Step 12 knn_i←在 S_m' 中找出 x_i的 k2 个近邻；
Step 13 H_i←在 x_i 的局部近邻域内计算海林格距离；
Step 14 D_i'←将 x_i 的局部近邻域内样本点保存；
Step 15 end
Step 16 D′=Ø；
Step 17 for i=1 to n
Step 18 x_i←从 S_m' 随机选取一个样本点；
Step 19 H_{knni}← 从 H_i 中找出样本点 x_i 对应的 K 近邻的海林格距离；
Step 20 (x_{syn}，G)←HDCS(x_i，knn_i，H_i，H_{knni})；
Step 21 x_{syn}←Evaluation(x_i，G，D_i'，H_t，H_i，x_{syn})；
Step 22 D′←将 x_{syn}置于 D′ 中；
Step 23 end
Step 24 D←将 D′ 置于 D 中；
Step 25 end
Step 26 D′←D。

本文用一个简单的实例来形象地说明 HDSMDTE 算法的有效性。人工合成一个简单的二维数据集，如图 3 所示，其中“+”“x”代表大类，实心五角星和圆点代表小类。用 7 种代表性算法与 HDSMOTE 算法进行采样，结果如图 4 所示。

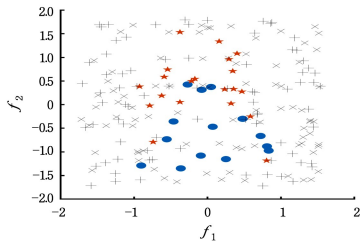
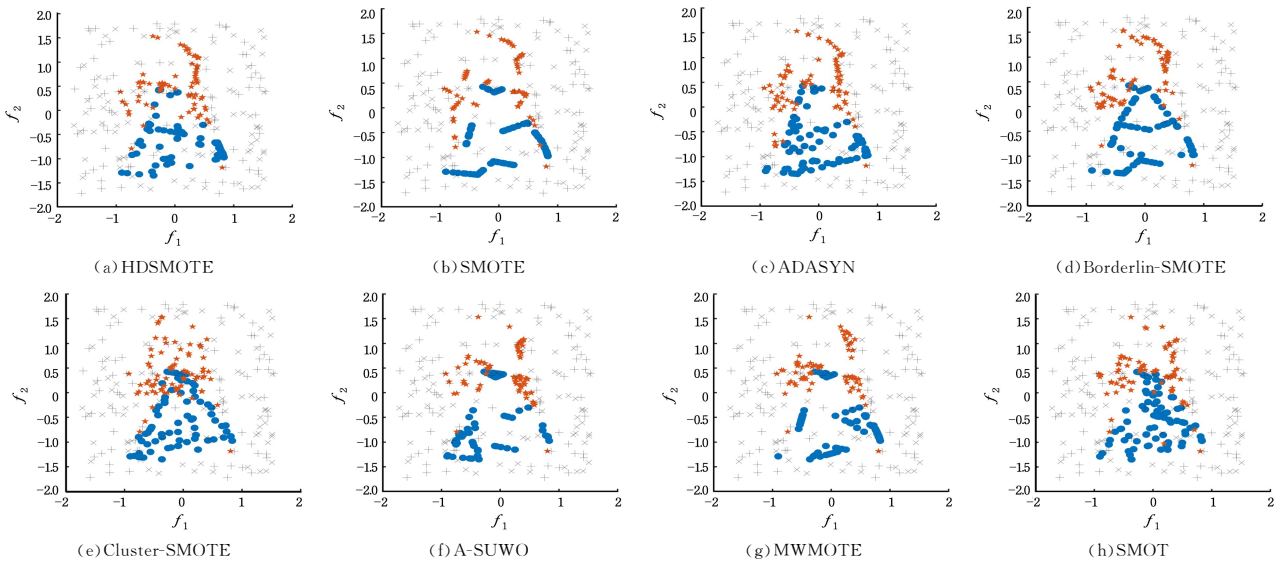


图 3 原始分布
Fig. 3 Original distribution



(a) HDSMOTE (b) SMOTE (c) ADASYN (d) Borderlin-SMOTE
(e) Cluster-SMOTE (f) A-SUWO (g) MWMOTE (h) SMOT

图 4 采样结果
Fig. 4 Results of oversampling

从图中可以看出，HDSMOTE算法能够很好地在泛化的基础上避免侵入其他类别区域，其他算法新合成的样本点可能会侵入到其他类别的区域。

3.4 时间复杂性分析

本文首先考虑HDSMOTE对一个少数类合成样本点时的时间复杂度，再计算HDSMOTE对多个少数类合成样本点的时间复杂度。在删除混淆点的过程中，为少数类中的每个样本计算 k_1 个近邻，时间复杂度为 $n_s * O(\log(n_s))$。在基于海林格距离的采样方向选择策略中，为少数类中的每个样本计算 k_2 个近邻，时间复杂度为 $n_s' * O(\log(n_s'))$。计算样本点与其他样本的距离的时间复杂度为 $n_s' * O(n_s')$，而计算海林格距离的时间复杂度为 $n_s * O(n_s * p)$。在基于海林格距离的采样质量评估策略中，涉及多次海林格距离的计算，时间复杂度为 $(R+1) * (n_k+1) * O((n_k+1) * p)$，其中 n_k 表示对应的局部近邻域中的全部样本数量，则一个小类的时间复杂度为：

$$O(n_s' * \log(n_s') + n_s * \log(n_s) + n_s'^2 + n_s^2 * p + (R+1)(n_k+1)^2 * p \quad (3)$$

由于 $n_s^2 \geqslant (n_k+1)^2 \geqslant n_k^2$，$n_s \geqslant n_s'$，$n_s^2 * p > (R+1)(n_k+1)^2 * p$，假设一个数据集的样本总数为 N，且 $\sum n_s < N$，则时间复杂度为：$2 * O(N * \log(N) + N^2 * p)$。由于 $N * \log(N) \leqslant N^2$，因此算法的时间复杂度为：$O(N^2 * p)$。

4 评价标准和实验结果

在不平衡学习问题中，以基于混淆矩阵的分类精度(Precision)、召回率(Recall)、F-measure、G-mean和MAUC作为评价标准[1-5]。MAUC[24]是在AUC(Area Under ROC Curve)的基础上扩展的多类不平衡学习的评价标准。

$$MAUC = \frac{2}{c(c-1)} \sum_{i<j} \frac{A_{i,j} + A_{j,i}}{2} \quad (4)$$

其中，$A_{i,j}$ 是类别 i 和类别 j 之间的AUC，c 代表数据集中所有类别的个数。因为 $A_{i,j}$ 和 $A_{j,i}$ 的值可能不同，所以需要分别计算，然后取平均值。

为验证所提方法在多类不平衡学习上的有效性，在15个多类不平衡数据集上，采用HDSMOTE，SMOTE，Borderline-SMOTE(B_S)，Cluster SMOTE(C_S)，ADASYN(ADA)，SMOM，MWMOTE(MW)和A-SUWO(ASU)进行采样处理，然后在基于C4.5决策树的分类器RIPPER[25]上对采样后的结果进行分类，对实验结果进行分析比较，以证明HDSMOTE算法的有效性。15个数据集的详细信息如表1所列。

表1 数据集

Table 1 Data sets

数据集名称	样本数	属性数	类别个数	IR
abalone8discrce	2148	10	8	11.18
abalone10discrce	2297	10	10	11.18
balance	625	4	3	5.88
cleveland	597	13	5	12.31
housing5	506	13	5	7.71
hayes-routh	160	4	3	2
newthyroid	215	5	3	5.3
plates-faults1	1941	27	7	12.24
plates-faults3	1941	27	5	15.69
sat4	4374	36	4	2.45
vehicle-mc	864	18	3	2.16
wine	178	13	3	1.5
yeast8	1484	8	8	18.52
yeast52	982	8	5	18.52
yeast71	1055	8	7	23.15

实验中，数据经过零均值规范化，采用五折交叉验证，所有采样方法独立运行10次后取平均值，不平衡率(IR)取1.45。其中，k_1 取值为10，k_2 取值为5，R 取值为10，k 取值为3。k_1 的大小会影响到混淆点的选择，实验中曾将 k_1 取值为10，20和30，结果发现 k_1 取10时效果最好。对于比较次数 R，比较次数越大，效果越好，但是在经过一定次数之后效果不会再显著提升，实验发现，R 取10时得到的实验效果最优。

Precision，Recall，F-measure和G-mean评价标准仅通过样本数量最小的少数类和样本数量最大的多数类计算，MAUC是基于数据集中所有类别的综合性指标，最终结果如表2—表6所列。在时间复杂度方面，记录实验过程中采样部分的运行时间，每种采样算法独立运行10次，取平均值，结果如表7所列。整体而言，HDSMOTE算法在RIPPER分类器上有更好的分类效果。

表2 8种上采样算法在15个数据集上的 *Precision*

Table 2 *Precision* of eight oversampling algorithms over fifteen data sets

数据集名称	HDSMOTE	SMOTE	ADA	B_S	C_S	ASU	MW	SMOM
abalone8discre	0.9341	0.8692	0.9131	0.9498	0.7962	0.9179	**0.9595**	0.8486
abalone10discre	0.9383	0.8659	0.9462	0.8815	0.7964	0.9036	**0.9595**	0.8432
balance	0.7913	0.7882	0.7691	**0.7927**	0.7563	0.7316	0.7597	0.7674
cleveland	**0.8900**	0.7788	0.8075	0.8844	0.7813	0.8800	0.7438	0.7544
housing5	0.9598	0.9590	0.9464	0.9619	0.9360	0.9579	**0.9715**	0.9544
hayes-roth	0.9277	0.8892	0.9215	0.8938	0.9031	**0.9610**	0.9123	0.9092
newthyroid	0.9780	0.9773	0.9820	**0.9827**	0.9787	0.9649	0.9713	0.9807
plates-faults1	**0.9774**	0.9593	0.9612	0.9713	0.9230	0.9766	0.9728	0.9626
plates-faults3	0.9808	0.9731	0.9721	0.9751	0.9512	**0.9829**	0.9784	0.9691
sat4	**0.8859**	0.8810	0.8784	0.8748	0.8658	0.8702	0.8757	0.8729
vehicle-mc	0.9436	0.9459	**0.9473**	0.9483	0.9382	0.9395	0.9471	0.9403
wine	0.9042	0.9083	**0.9250**	0.9250	0.9229	0.9208	0.9146	0.9083
yeast8	**0.9555**	0.8564	0.8827	0.9158	0.9257	0.9252	0.7875	0.8903
yeast52	**0.9646**	0.8765	0.8533	0.9376	0.9397	0.9289	0.8428	0.8879
yeast71	**0.9739**	0.8644	0.8832	0.9361	0.9287	0.9312	0.8032	0.9257

表 3 8 种上采样算法在 15 个数据集上的 *Recall*

Table 3 *Recall* of eight oversampling algorithms over fifteen data sets

数据集名称	HDSMOTE	SMOTE	ADA	B_S	C_S	ASU	MW	SMOM
abalone8discre	0.6980	0.6738	0.6155	**0.8090**	0.6173	0.6894	0.6340	0.6286
abalone10discre	**0.7903**	0.6881	0.7801	0.7202	0.6005	0.6625	0.7530	0.7174
balance	0.8440	0.8340	0.8240	0.8325	0.8491	**0.8935**	0.8823	0.8368
cleveland	0.8365	0.7421	0.7392	**0.8396**	0.7049	0.8287	0.6280	0.7211
housing5	0.9480	0.9296	0.9415	0.9480	0.9001	0.9254	**0.9485**	0.9411
hayes-roth	0.8720	0.8906	0.8861	0.9069	0.8164	**0.9878**	0.8900	0.8805
newthyroid	0.9639	0.9609	0.9591	0.9524	0.9704	0.9702	0.9677	**0.9705**
plates-faults1	0.9308	0.9143	0.9271	0.9230	0.8935	0.9081	**0.9472**	0.8763
plates-faults3	**0.9586**	0.9347	0.9379	0.9544	0.9027	0.9236	0.9455	0.9319
sat4	0.8651	0.8543	**0.8866**	0.8859	0.8455	0.8460	0.8764	0.8556
vehicle-mc	**0.9328**	0.9229	0.9146	0.9198	0.9167	0.9138	0.9205	0.9320
wine	0.9340	0.9488	0.9430	0.9551	0.9524	0.9477	**0.9554**	0.9448
yeast8	**0.8955**	0.7524	0.7831	0.8512	0.8347	0.7627	0.6872	0.7573
yeast52	**0.9519**	0.8352	0.8036	0.8704	0.8825	0.8806	0.7639	0.8159
yeast71	**0.9545**	0.8292	0.8692	0.9219	0.9108	0.8535	0.7371	0.8885

表 4 8 种上采样算法在 15 个数据集上的 *F-measure*

Table 4 *F-measure* of eight oversampling algorithms over fifteen data sets

数据集名称	HDSMOTE	SMOTE	ADA	B_S	C_S	ASU	MW	SMOM
abalone8discre	0.7924	0.7496	0.7200	**0.8684**	0.6757	0.7694	0.7465	0.7037
abalone10discre	**0.8534**	0.7584	0.8472	0.7877	0.6712	0.7571	0.8338	0.7662
balance	**0.8165**	0.8099	0.7950	0.8117	0.7992	0.8035	0.8156	0.8002
cleveland	**0.8618**	0.7587	0.7707	0.8607	0.7395	0.8532	0.6799	0.7353
housing5	0.9537	0.9437	0.9437	0.9546	0.9174	0.9411	**0.9597**	0.9476
hayes-roth	0.8964	0.8865	0.9007	0.8989	0.8521	**0.9738**	0.8992	0.8909
newthyroid	0.9709	0.9690	0.9704	0.9672	0.9744	0.9675	0.9694	**0.9755**
plates-faults1	0.9533	0.9356	0.9437	0.9461	0.9074	0.9406	**0.9597**	0.9167
plates-faults3	**0.9695**	0.9534	0.9546	0.9645	0.9261	0.9521	0.9616	0.9499
sat4	0.8753	0.8674	**0.8824**	0.8802	0.8555	0.8579	0.8760	0.8641
vehicle-mc	0.9489	0.9437	0.9346	0.9367	0.9446	0.9409	**0.9492**	0.9455
wine	0.9181	0.9277	0.9336	**0.9395**	0.9365	0.9336	0.9338	0.9258
yeast8	**0.9225**	0.7965	0.8261	0.8790	0.8731	0.8320	0.7303	0.8158
yeast52	**0.9579**	0.8544	0.8253	0.9021	0.9079	0.9034	0.8009	0.8478
yeast71	**0.9639**	0.8456	0.8757	0.9283	0.9190	0.8899	0.7671	0.9062

表 5 8 种上采样算法在 15 个数据集上的 *G-mean*

Table 5 *G-mean* of eight oversampling algorithms over fifteen data sets

数据集名称	HDSMOTE	SMOTE	ADA	B_S	C_S	ASU	MW	SMOM
abalone8discre	0.5128	0.4521	0.4142	**0.5507**	0.3930	0.4443	0.5219	0.4086
abalone10discre	**0.6562**	0.5308	0.5817	0.6263	0.4925	0.5614	0.5758	0.4997
balance	0.8218	0.8254	0.8202	0.8228	0.8233	0.8347	**0.8386**	0.8188
cleveland	**0.7921**	0.7270	0.7112	0.7806	0.6800	0.7558	0.6504	0.6984
housing5	**0.8393**	0.8148	0.8163	0.8091	0.7915	0.8026	0.8311	0.8170
hayes-roth	0.8125	0.8270	0.8193	0.8318	0.7928	**0.8877**	0.8235	0.8163
newthyroid	0.9558	0.9533	0.9609	0.9611	0.9616	0.9555	0.9548	**0.9641**
plates-faults1	**0.8030**	0.7888	0.7821	0.7798	0.7505	0.7659	0.7954	0.7591
plates-faults3	**0.8458**	0.8317	0.8319	0.8403	0.8073	0.8202	0.8329	0.8226
sat4	0.9062	0.9024	**0.9225**	0.9211	0.8935	0.8927	0.9137	0.9016
vehicle-mc	**0.9299**	0.9265	0.9247	0.9239	0.9159	0.9196	**0.9244**	0.9286
wine	0.9146	0.9156	0.9224	0.9226	0.9242	0.9211	0.9294	0.9165
yeast8	**0.6874**	0.5953	0.5863	0.6119	0.6461	0.5739	0.5487	0.5907
yeast52	**0.7443**	0.6726	0.6556	0.7150	0.6695	0.7208	0.6395	0.6692
yeast71	**0.8282**	0.7217	0.7478	0.7676	0.7413	0.7507	0.6796	0.7448

表 6 8 种上采样算法在 15 个数据集上的 *MAUC*

Table 6 *MAUC* of eight oversampling algorithms over fifteen data sets

数据集名称	HDSMOTE	SMOTE	ADA	B_S	C_S	ASU	MW	SMOM
abalone8discre	**0.8672**	0.8051	0.8041	0.8602	0.7913	0.8408	0.8081	0.8111
abalone10discre	0.6163	0.6263	0.6242	0.5877	0.6472	0.5967	**0.6481**	0.6280
balance	0.9101	**0.9167**	0.9121	0.9039	0.9001	0.9045	0.9090	0.9055
cleveland	0.8921	0.8450	**0.8540**	0.8954	0.8252	0.8825	0.8070	0.8376
housing5	**0.9496**	0.9431	0.9370	0.9364	0.9360	0.9406	0.9441	0.9417
hayes-roth	0.9207	0.9156	0.9186	0.9170	0.8997	**0.9436**	0.9166	0.9083
newthyroid	0.9792	0.9765	0.9788	0.9731	0.9786	0.9759	0.9787	**0.9816**
plates-faults1	**0.9605**	0.9551	0.9504	0.9549	0.9515	0.9578	0.9571	0.9524
plates-faults3	**0.9597**	0.9538	0.9536	0.9590	0.9518	0.9571	0.9564	0.9547
sat4	0.9643	0.9627	**0.9679**	0.9678	0.9597	0.9596	0.9679	0.9630
vehicle-mc	**0.9690**	0.9671	0.9645	0.9652	0.9607	0.9645	0.9644	0.9683
wine	0.9371	0.9497	0.9492	0.9507	**0.9553**	0.9488	0.9511	0.9453
yeast8	**0.9372**	0.9083	0.9024	0.9178	0.9122	0.9109	0.8921	0.9114
yeast52	**0.9426**	0.9040	0.8920	0.9196	0.8994	0.9261	0.8924	0.9031
yeast71	**0.9630**	0.9264	0.9220	0.9427	0.9308	0.9364	0.8961	0.9306

表 7 8 种上采样算法在 15 个数据集上的采样时间

Table 7 Sampling time of eight oversampling algorithms over fifteen data sets

(单位:s)

数据集名称	HDSMOTE	SMOTE	ADA	B_S	C_S	ASU	MW	SMOM
abalone8discre	34.4744	7.4698	0.3463	8.3383	5.5288	31.2275	4.6712	14.3642
abalone10discre	81.3533	9.6792	0.4122	12.1184	10.7792	49.6335	6.1597	19.1997
balance	1.0967	0.4933	0.0106	0.6731	0.5867	1.2200	0.0721	0.5691
cleveland	11.1954	1.2017	0.0194	1.4488	1.2974	2.5502	0.1801	0.8597
housing5	7.7944	1.8261	0.0402	2.0156	1.7505	5.0059	0.5019	2.0605
hayes-roth	0.2072	0.0548	0.0040	0.0679	0.0733	0.1842	0.0227	0.2046
newthyroid	1.0046	0.5384	0.0080	0.6740	0.5999	1.1830	0.0604	0.4144
plates-faults1	46.2955	9.3131	0.5622	21.8247	7.8195	65.0621	9.1125	25.7622
plates-faults3	35.5943	5.9156	0.1639	14.5429	5.7244	15.0510	1.1144	4.3402
sat4	30.7180	5.5796	1.5597	14.5863	5.3512	214.5633	27.2138	83.6970
vehicle-mc	4.3985	0.9902	0.0811	1.1962	1.0574	8.1575	1.7666	5.2547
wine	0.0005	0.0003	0.0004	0.0002	0.0002	0.0002	0.0003	0.0015
yeast8	35.3110	5.5324	0.1403	6.4866	5.5103	27.6887	2.4222	6.5544
yeast52	26.9518	3.2646	0.0288	3.8179	3.1814	5.9192	0.1805	0.7000
yeast71	45.6017	5.3735	0.1233	6.3620	5.5413	16.8991	2.3328	5.6910

以 Precision 为评价标准时,HDSMOTE 比 SMOTE 平均提升了 4.08%;在 cleveland 数据集上的表现最好,提高了 11.12%。HDSMOTE 分别比 ADA,B_S,C_S,ASU,MW 和 SMOM 平均增加了 2.77%,1.16%,4.41%,1.42%,4.04% 和 3.93%;最高分别提升了 11.13%,5.68%,14.19%,5.97%,17.07% 和 13.56%。以 Recall 为评价标准时,HDSMOTE 比 SMOTE 平均提升了 4.43%;在 yeast8 数据集上的表现最好,提高了 14.31%。HDSMOTE 分别比 ADA,B_S,C_S,ASU,MW 和 SMOM 平均增加了 3.77%,0.57%,5.19%,2.55%,5.59%和 4.52%;最高分别提升了 14.83%,8.15%,18.98%,13.28%,21.74% 和 13.82%。以 F-measure 为评价标准时,HDSMOTE 比 SMOTE 平均提升了 4.36%;在 yeast8 数据集上的表现最好,提高了 12.60%。HDSMOTE 分别比 ADA,B_S,C_S,ASUMW 和 SMOM 平均增加了 3.54%,0.86%,5.03%,2.26%,5.15%和 4.42%;最高分别提升了 13.26%,6.57%,8.22%,9.63%,19.63% 和 12.65%。以 G-mean 为评价标准时,HDSMOTE 比 SMOTE 平均提升了 3.77%;在 abalone10discre 数据集上的表现最好,提高了 12.54%。HDSMOTE 分别比 ADA,B_S,C_S,ASU,MW 和 SMOM 平均增加了 3.69%,1.24%,5.11%,2.95%,3.93%和 4.63%;最高分别提升了 10.11%,7.55%,16.37%,11.35%,14.86%和 15.65%。以 MAUC 为评价标准时,HDSMOTE 比 SMOTE 平均提升了 1.42%;在 abalone8discre 数据集上的表现最好,提高了 6.21%。HDSMOTE 分别比 ADA,B_S,C_S,ASU,MW 和 SMOM 平均增加了 1.59%,0.78%,1.79%,0.82%,1.86%和 1.51%;最高分别提升了 6.31%,2.86%,7.59%,2.66%,8.51%和 5.61%。

在采样运行时间方面,与 SMOTE,ADA,B_S,C_S,MW 和 SMOM 相比,HDMOTE 的运行时间更长,但其比 ASU 的运行时间短。整体而言,HDSMOTE 算法在以上 5 个评价标准上性能均有所提升。例如,在 yeast8,yeast52 和 yeast71 上,HDSMOTE 的运行时间比其他算法都长,但在以上 5 个评价标准上性能均有大幅度的提升。

总体而言,HDSMOTE 算法比 SMOTE,ADASYN,Borderline-SMOTE,Cluster-SMOTE,A-SUWO,MWMOTE 和 SMOM 算法在 RIPPER 分类器上有更好的分类效果。

结束语 针对多类不平衡数据,本文提出了一种基于海林格距离和 SMOTE 的采样算法。该算法首先建立基于海林格距离的采样方向选择策略,通过比较少数类别样本点局部近邻域内的海林格距离的大小,来引导合成样本点的方向。其次,建立基于海林格距离的采样质量评估策略,对合成的样本点进行评估,以避免合成的样本点侵入其他类别的区域。最后,在 15 个多类不平衡数据集上采用 7 种代表性的采样方法和 HDSMOTE 算法对数据进行预处理,用基于 C4.5 决策树的 RIPPER 分类器进行分类,实验结果表明 HDSMOTE 算法有更好的分类效果。但是,该算法的运行时间较长,不能很好地处理混合变量,且对局部近邻域的定义较简单。下一步将根据数据分布定义局部近邻域,寻找新的突破点,将海林格距离更好地应用到上采样中,以解决多类不平衡学习中的问题。

参 考 文 献

[1] HE H,GARCIA E A. Learning from Imbalanced Data [J]. IEEE Transactions on Knowledge & Data Engineering, 2009, 21(9):1263-1284.

[2] KRAWCZYK,BARTOSZ. Learning from imbalanced data:open challenges and future directions [J]. Progress in Artificial Intelligence, 2016, 5(4):221-232.

[3] LI Y X,CHAI Y,HU Y Q,et al. Review of imbalanced data classification methods[J]. Control and Decision, 2019, 34(4): 673-688.

[4] ZHAO N,ZHANG X F,ZHANG L J. Overview of Imbalanced Data Classification[J]. Computer Science, 2018, 45(S1):22-27, 57.

[5] LI Y,LIU Z D,ZHANG H J. Review on ensemble algorithms for imbalanced data classification[J]. Application Research of Computers, 2014, 31(5):1287-1291.

[6] GUO H X,LI Y J,JENNIFER S,et al. Learning from class-imbalanced data:Review of methods and applications [J]. Expert Systems with Applications, 2017, 73:220-239.

[7] MIAO Z M,ZHAO L W,TIAN S W,et al. Class Imbalance Learning for Identifying NLOS in UWB Positioning[J]. Journal

of Signal Processing,2016,32(1):8-13.

[8] XIA P P,ZHANG L. Application of Imbalanced Data Learning Algorithms to Similarity Learning[J]. Pattern Recognition and Artificial Intelligence | Patt Recog Artif Intell,2014,27(12):1138-1145.

[9] WEI W W,LI J J,GAO L B. Effective detection of sophisticated online banking fraud on extremely;imbalanced data[J]. World Wide Web-internet & Web Information Systems,2013,16(4):449-475.

[10] CHAWLA N V,BOWYER K W,HALL L O,et al. SMOTE:Synthetic Minority Over-sampling Technique[J]. Journal of Artificial Intelligence Research,2002,16(1):321-357.

[11] HE H,BAI Y,GARCIA E A,et al. ADASYN:Adaptive Synthetic Sampling Approach for Imbalanced Learning[C]// IEEE International Joint Conference on Neural Networks,2008(IJCNN 2008). IEEE,2008:1322-1328.

[12] BARUA S,ISLAM M M,YAO X,et al. MWMOTE—Majority Weighted Minority Oversampling Technique for Imbalanced Data Set Learning[J]. IEEE Transactions on Knowledge and Data Engineering,2014,26(2):405-425.

[13] NEKOOEIMEHR I,LAI-YUEN S K. Adaptive semi-unsupervised weighted oversampling (A-SUWO) for imbalanced datasets[J]. Expert Systems with Applications,2016,46:405-416.

[14] PUNTUMAPON K,RAKTHAMAMON T,WAIYAMAI K. Clusterbased minority over-sampling for imbalanced datasets [J]. IEICE TRANSACTIONS on Information and Systems,2016,99(12):3101-3109.

[15] HAN H,WANG W Y,MAO B H. Borderline-SMOTE:A New Over-Sampling Method in Imbalanced Data Sets Learning[M]// Advances in Intelligent Computing. Springer Berlin Heidelberg,2005:878-887.

[16] ANAND R,MEHROTRA K,MOHAN C K,et al. Efficient classification for multiclass problems using modular neural networks[J]. IEEE Transactions on Neural Networks,1995,6(1):117-124.

[17] ZHU T,LIN Y,LIU Y. Synthetic minority oversampling technique for multiclass imbalance problems [J]. Pattern Recognition,2017,72:327-340.

[18] ABDI L,HASHEMI S. To combat multi-class imbalanced problems by means of over-sampling techniques [J]. IEEE Transactions on Knowledge and Data Engineering,2016,28(1):238-251.

[19] YANG X,KUANG Q,ZHANG W,et al. AMDO:An Over-Sampling Technique for Multi-Class Imbalanced Problems[J]. IEEE Transactions on Knowledge & Data Engineering,2018,30(9):1672-1685.

[20] CIESLAK D A,CHAWLA N V. Learning Decision Trees for Unbalanced Data[C]// Joint European Conference on Machine Learning and Knowledge Discovery in Databases. Berlin:Springer,2008:241-256.

[21] CIESLAK D A,HOENS T R,CHAWLA N V,et al. Hellinger distance decision trees are robust and skew-insensitive[J]. Data Mining and Knowledge Discovery,2012,24(1):136-158.

[22] UCI. Machine Learning Repository[OL]. http://mlr. cs. umass. edu/ml/datasets. html.

[23] KEEL Dataset[OL]. https://sci2s. ugr. es/keel/category. php?cat=clas&order=name#sub2.

[24] HAND D J,TILL R J. A Simple Generalisation of the Area Under the ROC Curve for Multiple Class Classification Problems [J]. Machine Learning,2001,45(2):171-186.

[25] COHEN W W. Fast Effective Rule Induction[C]// Twelfth International Conference on International Conference on Machine Learning. Elsevier,1995:115-123.

DONG Ming-gang, born in 1977, Ph.D, professor, is senior member of China Computer Federation (CCF). His main research interests include intelligent computing, multi-objective optimization and machine learning.

JING Chao, born in 1983, Ph.D, associate professor. His main research interests include cloud computing and big data processing, workflow scheduling on cloud data center and deep reinforcement learning.