

标记分布集成学习

沈小霞, 许哲源, 於东军, 贾修一

(南京理工大学 计算机科学与工程学院, 江苏 南京 210094)

摘要: 标记分布学习是一种新型的学习范式, 该文提出了一种适用于标记分布问题形式的 Adaboost 集成算法, 能够有效地提升各种单独算法的预测精度。该文设计了一种新的用于反映排序损失的评价指标 SortLoss。该文将 Adaboost 应用在标记分布学习问题上。实验结果表明, 该文设计的 Adaboost-LDL 集成框架在 13 个真实数据集上能够显著提升标记分布学习算法的预测精度, 该文的方法将排序损失指标 SortLoss 平均降低至原先的 41.2%, KL 散度指标平均降低至原先的 68.5%。

关键词: 标记分布学习; 自适应提升; 排序损失; 集成学习

中图分类号: TP391 文章编号: 1005-9830(2020)06-0660-09

DOI: 10.14177/j.cnki.32-1397n.2020.44.06.004

Label distribution ensemble learning

Shen Xiaoxia, Xu Zheyuan, Yu Dongjun, Jia Xiuyi

(School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China)

Abstract: Label distribution learning is a new learning paradigm. This paper proposes an Adaboost ensemble algorithm suitable for the form of label distribution learning problem, which can effectively improve the prediction accuracy of various individual algorithms. This paper designs a new evaluation index SortLoss to reflect the sorting loss. This paper applies Adaboost to label distribution learning problems. The experimental results show that the Adaboost-LDL integration algorithm proposed here can significantly improve various existing LDL algorithms on 13 real data sets. Compared with the pre-integration algorithm, the sorting loss can be reduced to an average of 41.2% and the Kullback-Leibler Divergence indicator is reduced to an average of 68.5%.

Key words: label distribution learning; adaboost; sort loss; ensemble learning

收稿日期: 2020-09-14 修回日期: 2020-10-12

基金项目: 国家自然科学基金(61773208); 江苏省自然科学基金(BK20191287); 中央高校基本科研业务费专项资金(30920021131)

作者简介: 沈小霞(1996-), 女, 硕士生, 主要研究方向: 机器学习和数据挖掘, E-mail: xiaoxiashen@njust.edu.cn; 通讯作者: 贾修一(1983-), 男, 博士, 副教授, 主要研究方向: 粒计算与不确定性分析、机器学习和数据挖掘, E-mail: jiaxy@njust.edu.cn。

引文格式: 沈小霞, 许哲源, 於东军, 等. 标记分布集成学习[J]. 南京理工大学学报, 2020, 44(6): 660-668.

投稿网址: <http://zrxuebao.njust.edu.cn>

标记多义性问题是机器学习领域的热门研究方向。在现有的机器学习范式中,解决标记多义性问题的框架主要有两种,即单标签学习(Single-label learning, SLL)和多标签学习(Multi-label learning, MLL)^[1]。SLL 即为标准监督学习的特例,其假设每个样例只属于单个类,对应于现实世界中每个实例只有一个标签的情况。MLL 是 SLL 框架的扩展。在 MLL 中,一个实例可以同时与多个类别标记相关联。实际上这两种传统标签学习都旨在回答一个基本问题,“哪个/些标签可以描述实例?”。相对于 SLL 而言,MLL 允许答案由多个标签构成,从而可以在一定程度上处理标签的歧义问题。然而, SLL 和 MLL 都不能应用到更详尽的问题,即它们没有办法进一步回答“每个标记描述实例的程度是多少?”。也就是说,传统的标签学习范式不能给出标签所占实例描述的相对比重。

在现实世界中,标签间相对比重所隐含的信息可能比许多人想象的更常见。一个典型例子是面部表情的情绪分析。面部表情经常传达多种基本情绪的复杂混合,每种基本情感在表达中起着不同的作用,所有基本情绪的相对强度自然形成面部表情的情绪分布。通过将具有最高强度的情绪或具有高于阈值的强度的情绪视为阳性标记,该问题可以选择 SLL 或 MLL 作为解决方案。然而这样做的代价是,将会失去相关情绪的不同强度的重要信息,这也就意味着,单靠 SLL 或 MLL 框架无法准确预测人物在现实图片所表现的复杂情感。因此,针对该问题,文献[3]提出了标记分布学习范式(Label distribution learning, LDL)。标记分布可以给出标签强度的分布,所以能够很好地回答“每个标签描述实例的程度是多少?”,进而可以得到更加准确的预测结果。也就是说,相对于传统标签学习而言,LDL 范式可以提供更丰富的实例描述信息,分配给实例的标签的不同相对重要性,能更好地表示标记多义性问题。

LDL 已广泛应用于许多现实世界的应用中。Geng 等为了解决通过图片预测人物年龄问题首次显式提出标记分布学习的概念^[4-5],随后将 LDL 范式应用于预测图片中头部姿势^[6],并将 LDL 的应用拓展到面部图片表情识别上,通过面部图片来预测人物表情的情绪分布^[7-8],紧接着又发表了一项结合图片相关文本进行表情预测的研究^[9]。另外, Geng 等还将 LDL 应用在电影评级

预测^[10]、美感感知^[11]等工作中。Xue 等则应用 LDL 范式尝试对人格倾向进行预测^[12]。除此之外还有 Zhang 等提出以 LDL 范式对视频监控中的人群计数^[13], Sun 等应用 LDL 范式对视频中出现的异常行为进行识别^[14]。

尽管越来越多的 LDL 算法被提出并取得了良好的效果,但是关于 LDL 的集成学习研究则很少,仅有 Geng 等提出的 LogitBoost 算法^[15]、Shen 等提出的 LDL forests 算法^[16]和 Zhai 等提出的集成神经网络(Ensemble neural networks, ENN)算法^[17]。LogitBoost 算法使用 LDL 模型族来学习样本以避免特定模型的潜在影响。LDL forests 算法则是利用可微决策树进行算法集成。ENN 算法则是使用具有不同标签集的训练集来训练具有偏好的神经网络。然而,上述的算法都是作为独立算法而设计的,目前没有用于提升 LDL 算法的通用集成框架。本文希望通过集成学习的方式,设计出一种适用于 LDL 算法的通用集成框架,能够有效地对目前各种独立算法进行提升。

1 理论基础

1.1 问题描述

本文中使用的符号如下所示。实例变量用 x 表示,标签变量用 y 表示, x 的属性集用 $X = \{x_1, x_2, \dots, x_n\}$ 表示,在特定的第 j 个标签上的描述度用 d_j 表示,而 x 的描述度分布则用 $D = \{d_1, d_2, \dots, d_n\}$ 来表示,其中 n 为标签的总数。LDL 是一种通过为每个标签 y 指定描述度来标记实例 x 的方法,即输入属性集 X ,输出为描述度分布 D 。同时需满足归一化条件,即对任意给定实例而言,描述度之和为 1。

1.2 Adaboost 方法

Adaboost 是一种实用的 Boosting 方法,它依赖于大量的分类器,使用一种基于权重的多数投票机制。分类器的权重是基于它的性能而分配的,可靠程度越高,权重就越大。它的核心思想是根据样本的概率分布来训练分类器,使得被前面分类器错误分类的样本在之后被选中的概率更大,从而使分类器组可以逐渐地聚焦于样本集中难以处理的方面。

Adaboost 逐个创建分类器,每个分类器都由不同的训练子集产生,而训练子集的构成依赖于前一个分类器。Adaboost 依据当前概率分布来选

取训练子集的成员。设训练集的样本总数为 m , 则开始时每个样本被选取的概率是相同的, 以概率矩阵 P_1 生成了训练子集 T_1 , 并训练出第一个分类器 C_1 。然后, 样本被选中的概率会根据 C_1 的结果进行修改。具体而言, 被分类器 C_1 正确分类的样本再被 T_2 选中的概率降低, 而被分类器 C_1 错误分类的样本再被 T_2 选中的概率会增加, 分类器 C_2 则依据修改后的概率矩阵 P_2 生成训练子集 T_2 , 并按照这样的方式继续迭代下去直到指定迭代次数。

与传统 Boosting 算法另一个不同之处是最终结果的投票机制。Adaboost 使用一种基于权重的多数投票机制, 分类器的权重是基于它的性能而分配的, 可靠程度越高, 权重就越大。在实际应用中准确地赋予各分类器权重是非常重要的, 可以使用感知机的学习方法来确定权重。刚开始时给每个分类器相等的权重, 然后不断用不同样本去训练这个系统。每次系统错分一个样本, 就可以根据系统预测值和真实值之间的关系来增加或减少某些分类器的权重。因此, Adaboost 算法是一种实用的 Boosting 方法, 它既考虑到分类器间的互补性, 也相对于传统 boosting 方法减少对于样本的选择代价, 是结合 LDL 研究的优选方案。

2 Adaboost-LDL 框架设计

Adaboost 是一种实用的 Boosting 方法, 然而 Adaboost 仅适用于传统分类问题, 不能够直接应用到 LDL 问题上来。本文设计了一种适用于 LDL 问题的 Adaboost 集成框架, 并将其命名为 Adaboost-LDL, 能够将 Adaboost 框架扩展到标记分布学习上来。

2.1 SortLoss 指标的提出

为了能够在集成学习中有效地区分样本的预测质量好坏, 本文从 Learning To Rank 问题 and 多标记问题上的 PRO 损失^[18] 获得启发, 设计了一种 LDL 问题专用的量化排序损失评价指标, 并将其命名为 SortLoss。

给定描述度分布 $D = \{d_1, d_2, d_3, d_4\}$, 在 LDL 问题中描述度的和为 1, 即

$$\sum_{j=1}^n d_j = 1 \quad (1)$$

假设比例关系为

$$d_1 : d_2 : d_3 : d_4 = 2 : 3 : 4 : 1 \quad (2)$$

那么便可以通过式 (1) 和式 (2) , 解得 $D = \{0.2, 0.3, 0.4, 0.1\}$ 。接下来对描述度序列进行由高到低的排序, 便得到排序后的序列 $H = \{h_1, h_2, h_3, h_4\}$, 在上例中 h_1, h_2, h_3, h_4 分别对应着 d_3, d_2, d_1, d_4 , 本文将该索引序列记为 $I = \{3, 2, 1, 4\}$ 。

本文用一种分类器对分布 D 进行预测, 得到的预测值记为 $\hat{D} = \{\hat{d}_1, \hat{d}_2, \hat{d}_3, \hat{d}_4\}$, 假设 $\hat{D} = \{0.1, 0.4, 0.3, 0.2\}$, 用之前记录的索引序列 $I = \{3, 2, 1, 4\}$ 得到 $\hat{H} = \{\hat{h}_1, \hat{h}_2, \hat{h}_3, \hat{h}_4\} = \{\hat{d}_3, \hat{d}_2, \hat{d}_1, \hat{d}_4\}$, 即 $\hat{H} = \{0.3, 0.4, 0.1, 0.2\}$ 。由此可以发现描述度在预测值上排序关系已经发生改变。为了能够量化这种排序变动, 本文还设计了 SortLoss 指标, 计算公式如下

$$\text{SortLoss} = \frac{1}{\sum_{j=1}^{n-1} \frac{1}{\log_2(j+1)}} * \sum_{j=1}^{n-1} \sum_{k=j+1}^n \frac{\delta(\hat{h}_k - \hat{h}_j)}{\log_2(j+1)} \quad (3)$$

式中:

$$\delta(x) = \begin{cases} x & x > 0 \\ 0 & x \leq 0 \end{cases} \quad (4)$$

$\hat{H} = \{0.3, 0.4, 0.1, 0.2\}$, 有两对标记需要计算排序损失, 即 $\hat{h}_1 < \hat{h}_2$ 和 $\hat{h}_3 < \hat{h}_4$, 可以根据公式计算出 $\text{SortLoss} \approx 0.0734$ 。式 (3) 中的 $1/\log_2(j+1)$ 是作为衰减因子设置的。对于一般 LDL 问题而言, 描述度排序越靠后的标记对样本描述的贡献就越小, 即 $\hat{h}_3 < \hat{h}_4$ 对排序造成的损失应比 $\hat{h}_1 < \hat{h}_2$ 占的

比重要少。而 $\frac{1}{\sum_{j=1}^{n-1} \frac{1}{\log_2(j+1)}}$ 则是作为归一化项

设计的, 能够保证无论有多少个标签, SortLoss 指标都能够在 0 至 1 的范围内变化。

2.2 集成框架实现

本文所实现的 Adaboost 框架针对 LDL 问题做了一些调整, 区别于传统分类问题中以分类是否正确来修改样本选择概率, 它是根据预测结果的 SortLoss 来修改样本选择概率, 并将每轮迭代中分类器在训练全集上的 SortLoss 作为最终确定分类器权重的指标, 具体算法如表 1 所示。其中迭代次数 boostTimes 和迭代进化因子 α 被设置为自由参数, L 为基础学习器。boostTimes 即在集成中所使用的分类器数量, 也是进行迭代的次数, 每轮迭代将会训练出一个由基础学习算法产生的分类器, 每个分类器对应于一个从特征域 X 映射到标记域 D 的权重矩阵 W 。 α 为每轮迭代结果对样

本选择概率矩阵 P 与上一轮相比影响的比重 α 越大,当前迭代对选择概率矩阵 P 的影响也就越大,同时前一轮迭代对 P 所残留的影响也就越小。概率矩阵 P 中的元素对应着训练集中每个样本在重采样中被选入子训练集概率的大小。概率矩阵 P 的更新公式为表 1 的第 8 行公式,即在每一轮子训练集的选取中,使得那些有较高排序损失的样本对应着较高的被重采样的概率。权重矩阵 W 则依赖于基准算法所用模型进行更新,以 BFGS 算法为例,在每一轮迭代 i 中采用最大熵模型学习权重矩阵 W_i ,并将其作为下一轮迭代的初始化矩阵。此外,本文对集成框架在 LDL 问题上做了一些优化,对于需要输入初始特征权重矩阵的一类算法,则将上轮迭代所产生的特征权重作为当前迭代中的初始特征权重矩阵输入,这样优化措施可以减少之后迭代中分类器的训练时间。

表 1 Adaboost-LDL 算法

Algorithm: Adaboost-LDL	
Input: training number training set T , test features F , boostTimes α	
Output: predict distribution \hat{D}	
1	Initialize probability matrix P_1
2	for $i=1$ to boost Times do
3	Get training subset T_i from T based on P_i
4	Learn T_i with a base learner L to obtain features weight matrix W_i
5	Use training set T as a test set to predict results with W_i
6	Use test features F to predict \hat{D}_i with W_i
7	Calculate $SortLoss(x)_i$ of each sample x in T to update P_{i+1} :
8	$P_{i+1} = P_i + \alpha * SortLoss(x)_i / \sum_{x \in T} SortLoss(x)_i$
9	end
10	Determine the base learner weight m_i based on $SortLoss$ on T :
11	$m_i = \frac{\sum_{i=1}^{boostTimes} \sum_{x \in T} SortLoss(x)_i}{\sum_{x \in T} SortLoss(x)_i}$
12	Get \hat{D} with each base learner's \hat{D}_i and m_i :
13	$\hat{D} = \frac{m_i}{\sum_{i=1}^{boostTimes} m_i} * \hat{D}_i$

3 实验

3.1 数据集

本实验采用了 13 个数据集,包括 10 个酵母菌基因数据集、2 个面部表情数据集和 1 个自然

景色数据集,如表 2 所示。

表 2 实验所用 13 个真实数据集的基本信息

序号	数据集	样本	特征	标签
1	Yeast-alpha	2465	24	18
2	Yeast-cdc	2465	24	15
3	Yeast-elu	2465	24	14
4	Yeast-diau	2465	24	7
5	Yeast-heat	2465	24	6
6	Yeast-spo	2465	24	6
7	Yeast-cold	2465	24	4
8	Yeast-dtt	2465	24	4
9	Yeast-spo5	2465	24	3
10	Yeast-spoem	2465	24	2
11	SJAFFE	213	243	6
12	SBU_3DFE	2500	243	6
13	Natural Scene	2000	294	9

Yeast 系列共 10 个数据集是从出芽酵母的生物实验中收集的真实数据集。每个数据集记录一次实验的结果,包括总共 2 465 个酵母基因,每个基因由长度为 24 的系统发育谱向量表示。对于每个数据集,每个标记对应于一个生物学实验期间的离散时间点。每个时间点的归一化后基因表达水平则提供了相应标记描述程度的自然测量。10 个酵母基因数据集中的标记数量总结在表 2 中。所有标记(每个标记对应一个实验离散时间点)的描述度(对应归一化后的基因表达水平)构成特定酵母基因的标记分布。SJAFFE 和 SBU_3DFE 是两个广泛使用的面部表情图像数据库的扩展,即 JAFFE 和 BU_3DFE。JAFFE 数据库包含由 10 个日本女模特构成的 213 个灰度表达图像。Geng 等人通过局部二值模式(LBP)的方法从每个图像中提取 243 维特征向量。每个图像由 6 个人在 6 个基本情绪标记(即:快乐、悲伤、惊讶、恐惧、愤怒和厌恶)上进行评分,具有 5 级量表。每种情绪的平均得分用于表示情绪强度。本文中使用的数据集 SJAFFE(Scored JAFFE)保留了所有得分,并将它们标准化为所有 6 个情感标记上的标记分布。类似地,对于包含 2 500 个面部表情图像的较大数据库 BU_3DFE,每个图像由 23 个人以与 JAFFE 评分相同的方式评分,从而得到数据集 SBU_3DFE(Scored BU_3DFE)的标记分布版本。Natural Scene 收集自 2 000 张拥有不一致排序的自然景色图片。每张图片分别由 10 个人对

9 个可能的标签进行排序,即植物、沙漠、天空、山、云、水、雪、建筑物和太阳。对于每张图片,每个人类排序者挑选相关的标签,并对这些挑选出来的标签按相关度进行降序排序。因此,每个人的多标记排序的结果将会是高度不一致的。然后,利用一个非线性规划过程将多标记排序结果转换为标记分布。最后,每张图片被提取为一个 294 维的向量。

3.2 评价指标

标记分布的语义是对样本描述度分布,因此排序关系是分布非常的一个重要的特征,所以本实验主要采用量化排序损失的 SortLoss 指标作为评价指标,计算方法由式(3)给出。目前在 LDL 问题上的现有指标主要有切比雪夫距离(Chebyshev distance)、克拉克距离(Clark distance)、堪培拉度量(Canberra metric)、KL 散度(Kullback-Leibler divergence)、余弦系数(Cosine coefficient)和交叉相似度(Intersection similarity)。前四个是距离度量,越小越好,后两个是相似性度量,越大越好。这六种评价指标非常相似,本文分别从距离评价指标和相似性评价指标各挑一个作为本文实验使用的评价指标,归纳为表 3,其中 D 表示真实分布, \hat{D} 表示预测分布, H 表示从高到低排序的真实分布, \hat{H} 表示与 H 相对应的预测分布。

表 3 实验所用评价指标

评价指标	公式
SortLoss ↓	$\text{SortLoss}(H, \hat{H}) = \frac{1}{\sum_{j=1}^{n-1} \frac{1}{\log_2(j+1)}} * \sum_{j=1}^{n-1} \sum_{k=j+1}^n \frac{\delta(\hat{h}_k - \hat{h}_j)}{\log_2(j+1)}$
KL ↓	$KL(D, \hat{D}) = \sum_{j=1}^n d_j \ln \frac{d_j}{\hat{d}_j}$
Intersection ↑	$\text{Intersection}(D, \hat{D}) = \sum_{j=1}^n \min(D_j, \hat{D}_j)$

3.3 实验设置

本文设计的 Adaboost 框架希望通过集成基础分类器做出加权预测,所以实现了 LDL 问题上 7 种不同算法的集成,即 PT-BAYES^[19]、CNPP^[4]、SA-IIS^[18]、BFGS-LDL^[19]、LC-LDL^[20]、LDL_LCLR^[21]和 LDL-LS^[22]。PT-Bayes 是问题转化算法的代表,将贝叶斯规则计算的后验概率作为预测的描

述度。CNPP 则是利用类似于 Modha 神经网络的网络结构,但是以受监督的方式接受培训。在训练神经网络时已知真实的标签分布,然后可以基于网络结构预测新的分布。SA-IIS 和 BFGS-LDL 属于 LDL 专门算法,其思想在于最小化以 KL 散度为核心的目标函数,不同之处在于两种算法采取的优化方法不同。BFGS-LDL 在 SA-IIS 的基础上采用了拟牛顿法 BFGS 来加速迭代收敛。LC-LDL 利用标记间的相关性设计了目标函数,同样采用拟牛顿法进行优化。LDL_LCLR 利用标签之间的全局和局部相关性为训练模型提供更多的信息。LDL-LS 同时学习所有标签的共同特征和每个标签的特定特征,以增强 LDL 模型。本文在实验中对单独算法和集成算法采用论文默认参数。在 13 个真实世界数据集中,对单独算法和集成算法进行 10 次留出法验证,并记录每个评估量度的平均值和标准偏差。

3.4 Adaboost 与基准算法对比实验

本文将 7 个 LDL 算法分别进行 Adaboost 集成,在 13 个数据集上对集成前后的算法进行实验对比。文中把基准算法名称的一部分作为后续来区分不同基准算法的 Adaboost 框架。在实验中统一设定自由参数 boostTimes = 5, $\alpha = 1.0$ 。需要说明的是,由于下载的源码 PT-BAYES 和 SA-IIS 在两个表情数据集上无法正常工作,所以未给出其在 SJAFFE 和 SBU_3DFE 上的结果。表 4 给出了 SortLoss 指标上对比实验的完整实验数据,表 5 给出了 KL 散度指标上的完整实验数据。

本文给出以 SortLoss 指标评价的完整实验数据表格,SortLoss 指标越小表明预测结果越好。由于数据集过多,本文将其分为 3 张表格,每张表格包含 4 个数据集的结果,其中单独算法与相应集成算法放置于上下两行以供对比。表 4 实验结果表明,Adaboost-LDL 集成后的算法在 SortLoss 指标上平均下降至原先指标的 41.2%,提升效果是非常显著的。

为了能够在距离指标上也能客观的反映 Adaboost-LDL 框架的性能,本文也同时给出了 KL 散度和 Intersection 指标上实验结果,如表 5 和表 6 所示,7 个算法在 KL 散度的提升效果则有较大差异,平均下降到原先指标的 68.5%。对于 BFGS-LDL,CPNN 和 LC-LDL 一类以距离为优化目标函数的算法而言,其距离指标未发生显著变化。对于 PT-BAYES 和 SA-IIS 算法这一类自身性能欠

佳的算法 通过框架可以大幅提升。这也就意味 损失 而且会对一些算法上的距离评价指标带来
着本文的集成框架不仅能够降低各类算法的排序 提升 这与本文所期望的结果相一致。

表 4 SortLoss ↓ (mean±std) 指标在 13 个数据集上的实验结果

算法	Yeast-alpha	Yeast-cdc	Yeast-elu	Yeast-diau	Yeast-heat	Yeast-spo	Yeast-cold
PT-BAYES	.1884±.0122	.1805±.0085	.1874±.0137	.1694±.0098	.1717±.0157	.1735±.0097	.1602±.0178
Adaboost-BAYES	.0723±.0064	.0725±.0040	.0713±.0046	.0629±.0042	.0659±.0045	.0648±.0053	.0573±.0064
SA-HIS	.0960±.0038	.0952±.0017	.1005±.0028	.0959±.0037	.0985±.0057	.1204±.0060	.0929±.0058
Adaboost-HIS	.0250±.0011	.0253±.0006	.0257±.0009	.0226±.0011	.0219±.0015	.0244±.0012	.0207±.0017
CPNN	.0092±.0036	.0170±.0189	.0090±.0057	.0141±.0026	.0096±.0026	.0169±.0043	.0160±.0068
Adaboost-CPNN	.0036±.0013	.0036±.0007	.0035±.0014	.0037±.0005	.0032±.0011	.0039±.0010	.0042±.0012
BFGS-LDL	.0064±.0003	.0081±.0003	.0080±.0003	.0146±.0008	.0112±.0006	.0195±.0013	.0134±.0011
Adaboost-BFGS	.0049±.0002	.0057±.0002	.0054±.0002	.0101±.0004	.0054±.0004	.0075±.0006	.0063±.0007
LDL_LCLR	.0063±.0003	.0080±.0004	.0079±.0003	.0150±.0010	.0108±.0006	.0192±.0014	.0133±.0009
Adaboost-LCLR	.0048±.0003	.0057±.0003	.0055±.0002	.0103±.0007	.0054±.0003	.0076±.0007	.0062±.0006
LDL-LS	.0061±.0003	.0081±.0004	.0076±.0003	.0145±.0009	.0108±.0006	.0196±.0014	.0132±.0009
Adaboost-LS	.0048±.0002	.0057±.0003	.0053±.0002	.0103±.0006	.0053±.0004	.0076±.0005	.0061±.0006
LC-LDL	.0053±.0006	.0074±.0004	.0073±.0002	.0139±.0007	.0102±.0008	.0187±.0011	.0129±.0009
Adaboost-LC	.0029±.0001	.0040±.0002	.0039±.0001	.0094±.0004	.0037±.0003	.0061±.0005	.0053±.0006

算法	Yeast-dtt	Yeast-spo5	Yeast-spoem	SJAFEE	SBU_3DFE	Natural Scene
PT-BAYES	.1569±.0201	.1563±.0146	.1516±.0255	N/A	N/A	.0269±.0057
Adaboost-BAYES	.0524±.0063	.0554±.0041	.0449±.0070	N/A	N/A	.0948±.0075
SA-HIS	.0865±.0056	.0804±.0058	.0710±.0058	N/A	N/A	.1180±.0059
Adaboost-HIS	.0218±.0016	.0188±.0009	.0163±.0019	N/A	N/A	.0964±.0033
CPNN	.0071±.0062	.0117±.0030	.0236±.0028	.0166±.0111	.0450±.0138	.0932±.0093
Adaboost-CPNN	.0019±.0003	.0045±.0007	.0072±.0007	.0088±.0026	.0118±.0032	.0686±.0071
BFGS-LDL	.0081±.0007	.0142±.0017	.0253±.0021	.0611±.0126	.1143±.0082	.1479±.0079
Adaboost-BFGS	.0034±.0003	.0057±.0011	.0076±.0009	.0566±.0127	.1007±.0061	.1387±.0068
LDL_LCLR	.0081±.0006	.0147±.0016	.0243±.0030	.0149±.0023	.0310±.0018	.1444±.0089
Adaboost-LCLR	.0034±.0003	.0065±.0007	.0075±.0012	.0112±.0009	.0167±.0009	.1382±.0068
LDL-LS	.0081±.0006	.0149±.0016	.0242±.0030	.0246±.0045	.0370±.0022	.1710±.0104
Adaboost-LS	.0033±.0003	.0062±.0009	.0077±.0010	.0215±.0041	.0234±.0012	.1660±.0093
LC-LDL	.0072±.0007	.0129±.0015	.0243±.0020	.0208±.0036	.0724±.0056	.1211±.0063
Adaboost-LC	.0027±.0003	.0054±.0010	.0072±.0010	.0114±.0009	.0471±.0035	.1087±.0048

表 5 KL ↓ (mean±std) 指标在 13 个数据集上的实验结果

算法	Yeast-alpha	Yeast-cdc	Yeast-elu	Yeast-diau	Yeast-heat	Yeast-spo	Yeast-cold
PT-BAYES	.2751±.0371	.2852±.0364	.2776±.0420	.2607±.0290	.2645±.0456	.2591±.0229	.2196±.0245
Adaboost-BAYES	.0395±.0059	.0414±.0038	.0383±.0049	.0419±.0041	.0404±.0041	.0504±.0036	.0354±.0043
SA-HIS	.0549±.0026	.0546±.0019	.0589±.0027	.0877±.0047	.0622±.0032	.0998±.0065	.0580±.0048
Adaboost-HIS	.0092±.0003	.0104±.0003	.0099±.0004	.0170±.0007	.0154±.0006	.0285±.0014	.0147±.0009
CPNN	.0060±.0004	.0099±.0053	.0070±.0011	.0152±.0009	.0127±.0004	.0259±.0018	.0131±.0015
Adaboost-CPNN	.0057±.0003	.0071±.0003	.0064±.0003	.0157±.0007	.0127±.0005	.0267±.0016	.0130±.0007
BFGS-LDL	.0055±.0003	.0068±.0002	.0062±.0003	.0136±.0006	.0125±.0005	.0250±.0015	.0122±.0008
Adaboost-BFGS	.0055±.0003	.0068±.0003	.0062±.0003	.0140±.0006	.0126±.0005	.0260±.0016	.0125±.0008
LDL-LCLR	.0055±.0002	.0070±.0003	.0062±.0003	.0132±.0010	.0126±.0005	.0246±.0018	.0122±.0007
Adaboost-LCLR	.0055±.0002	.0070±.0003	.0062±.0003	.0135±.0009	.0128±.0005	.0256±.0018	.0125±.0007
LDL-LS	.0055±.0002	.0070±.0003	.0062±.0003	.0131±.0010	.0126±.0005	.0246±.0018	.0122±.0007
Adaboost-LS	.0055±.0002	.0070±.0003	.0062±.0003	.0135±.0009	.0129±.0005	.0256±.0018	.0125±.0007
LC-LDL	.0055±.0003	.0068±.0002	.0063±.0003	.0136±.0006	.0125±.0004	.0249±.0014	.0122±.0008
Adaboost-LC	.0055±.0003	.0069±.0003	.0063±.0003	.0141±.0006	.0127±.0005	.0261±.0015	.0125±.0007

算法	Yeast-dtt	Yeast-spo5	Yeast-spoem	SJAFFE	SBU_3DFE	Natural Scene
PT-BAYES	.1974±.0332	.2125±.0219	.1796±.0357	N/A	N/A	2.1929±.2648
Adaboost-BAYES	.0238±.0025	.0444±.0035	.0333±.0035	N/A	N/A	2.1110±.4100
SA-HIS	.0455±.0035	.0603±.0019	.0381±.0046	N/A	N/A	.9441±.0331
Adaboost-HIS	.0088±.0005	.0302±.0019	.0272±.0023	N/A	N/A	.9079±.0282
CPNN	.0069±.0009	.0286±.0015	.0246±.0018	.0767±.0074	.0817±.0040	.9534±.0530
Adaboost-CPNN	.0066±.0004	.0285±.0017	.0257±.0019	.0756±.0080	.0820±.0043	.9635±.0325
BFGS-LDL	.0064±.0004	.0281±.0015	.0248±.0017	.0748±.0126	.0542±.0022	.8230±.0523
Adaboost-BFGS	.0065±.0004	.0284±.0016	.0258±.0018	.0677±.0109	.0548±.0020	.8120±.0512
LDL-LCLR	.0063±.0007	.0293±.0028	.0244±.0021	.0681±.0110	.0753±.0023	.8072±.0560
Adaboost-LCLR	.0064±.0007	.0294±.0031	.0257±.0028	.0684±.0107	.0780±.0024	.8119±.0474
LDL-LS	.0063±.0007	.0293±.0027	.0244±.0021	.0421±.0087	.0624±.0029	.7202±.0423
Adaboost-LS	.0064±.0007	.0294±.0030	.0257±.0028	.0418±.0082	.0644±.0023	.7531±.0357
LC-LDL	.0064±.0004	.0280±.0015	.0248±.0017	.0784±.0090	.0805±.0039	.7574±.0424
Adaboost-LC	.0065±.0004	.0284±.0016	.0258±.0019	.0774±.0085	.0807±.0040	.7687±.0379

表 6 Intersection \uparrow (mean \pm std) 指标在 13 个数据集上的实验结果

算法	Yeast-alpha	Yeast-cdc	Yeast-elu	Yeast-diau	Yeast-heat	Yeast-spo	Yeast-cold
PT-BAYES	.7695±.0146	.7732±.0101	.7704±.0165	.7743±.0139	.7772±.0157	.7716±.0097	.7945±.0145
Adaboost-BAYES	.9033±.0066	.9014±.0035	.9039±.0052	.8975±.0047	.8973±.0047	.8819±.0050	.9043±.0056
SA-IIS	.8735±.0029	.8742±.0019	.8700±.0030	.8457±.0023	.8644±.0032	.8306±.0053	.8708±.0048
Adaboost-IIS	.9495±.0009	.9461±.0010	.9463±.0012	.9300±.0015	.9325±.0013	.9084±.0022	.9343±.0020
CPNN	.9600±.0014	.9495±.0144	.9562±.0026	.9351±.0018	.9399±.0010	.9132±.0034	.9382±.0042
Adaboost-CPNN	.9616±.0009	.9573±.0011	.9577±.0009	.9339±.0011	.9401±.0011	.9131±.0026	.9387±.0016
BFGS-LDL	.9624±.0011	.9577±.0009	.9586±.0009	.9392±.0011	.9407±.0012	.9145±.0030	.9407±.0019
Adaboost-BFGS	.9624±.0010	.9579±.0010	.9585±.0009	.9384±.0011	.9405±.0012	.9139±.0027	.9400±.0018
LDL-LCLR	.9624±.0008	.9574±.0008	.9589±.0009	.9402±.0017	.9402±.0013	.9156±.0026	.9409±.0017
Adaboost-LCLR	.9623±.0008	.9576±.0009	.9587±.0009	.9394±.0017	.9398±.0014	.9147±.0026	.9402±.0017
LDL-LS	.9623±.0008	.9574±.0008	.9589±.0009	.9403±.0017	.9402±.0013	.9156±.0026	.9409±.0017
Adaboost-LS	.9623±.0008	.9576±.0009	.9586±.0009	.9394±.0017	.9398±.0014	.9147±.0027	.9402±.0017
LC-LDL	.9622±.0010	.9576±.0010	.9584±.0009	.9392±.0011	.9406±.0010	.9147±.0028	.9408±.0019
Adaboost-LC	.9623±.0010	.9579±.0011	.9582±.0009	.9383±.0011	.9403±.0012	.9138±.0025	.9398±.0018
算法	Yeast-dtt	Yeast-spo5	Yeast-spoem	SJAFIE	SBU_3DFE	Natural Scene	
PT-BAYES	.8039±.0214	.7948±.0100	.8176±.0142	N/A	N/A	.3511±.0100	
Adaboost-BAYES	.9231±.0038	.8878±.0045	.8997±.0063	N/A	N/A	.3392±.0098	
SA-IIS	.8856±.0037	.8702±.0031	.8887±.0061	N/A	N/A	.4617±.0124	
Adaboost-IIS	.9495±.0011	.9068±.0032	.9096±.0044	N/A	N/A	.4677±.0113	
CPNN	.9561±.0030	.9095±.0027	.9127±.0038	.8411±.0089	.8406±.0045	.4538±.0212	
Adaboost-CPNN	.9570±.0010	.9100±.0031	.9126±.0039	.8429±.0099	.8419±.0047	.4319±.0126	
BFGS-LDL	.9579±.0010	.9104±.0029	.9120±.0035	.8523±.0121	.8717±.0027	.5479±.0130	
Adaboost-BFGS	.9574±.0010	.9099±.0031	.9123±.0037	.8594±.0111	.8692±.0023	.5425±.0152	
LDL-LCLR	.9583±.0011	.9087±.0043	.9133±.0032	.8531±.0138	.8453±.0027	.5532±.0146	
Adaboost-LCLR	.9579±.0012	.9087±.0049	.9126±.0041	.8524±.0138	.8437±.0029	.5421±.0134	
LDL-LS	.9583±.0011	.9087±.0043	.9133±.0032	.8875±.0113	.8588±.0029	.5082±.0133	
Adaboost-LS	.9578±.0012	.9086±.0048	.9126±.0040	.8863±.0120	.8557±.0025	.4999±.0115	
LC-LDL	.9579±.0010	.9105±.0030	.9122±.0035	.8428±.0106	.8409±.0043	.5480±.0147	
Adaboost-LC	.9573±.0009	.9100±.0029	.9123±.0037	.8426±.0106	.8424±.0044	5346±.0137	

4 结论

Adaboost 是一种有效减少样本偏差的集成学习方法,但过去的研究仅对传统分类问题有效,在标记分布问题上没有可以直接使用的方案。本文设计了一种在 LDL 问题上进行 Adaboost 的可行方案,并提出了一种用于衡量排序损失的评价指标 SortLoss。与以往工作所设计的集成算法不同,Adaboost-LDL 是一种强鲁棒性的集成方案,可以对目前存在的各种 LDL 单独算法进行提升。在算法的对比实验中,单独算法与集成后的算法进行了对比。实验结果表明,本文设计的 Adaboost 框架对于 LDL 中的基准算法是非常有效的,能够对所有实验中使用的单独算法在排序指标上进行大幅度提升。

同时,PT-Bayes 和 SA-IIS 算法在 Adaboost-LDL 集成后在 KL 散度指标上也有大幅提升。当结合距离指标和排序损失指标对 LDL 算法预测质量进行评价时,通过 Adaboost-LDL 集成后的算法在预测质量上有大幅提升。

参考文献:

- [1] Tsoumakas G, Katakis I. Multi-label classification: an overview [J]. International Journal of Data Warehousing and Mining, 2007, 3(3): 1-13.
- [2] 张佳欢, 李磊军, 李美争, 等. 基于变精度邻域粗糙集的多标记子空间研究[J]. 南京理工大学学报, 2019, 43(4): 414-422.

Zhang Jiahuan, Li Leijun, Li Meizheng, et al. Research

- on multi-label subspace based on variable precision neighborhood rough sets [J]. Journal of Nanjing University of Science and Technology, 2019, 43(4): 414–422.
- [3] Geng Xin. Label distribution learning [J]. IEEE Transactions on Knowledge and Data Engineering. 2016, 28(7): 1734–1748.
- [4] Geng Xin, Yin Chao, Zhou Zhihua. Facial age estimation by learning from label distributions [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2013, 35(10): 2401–2412.
- [5] Geng Xin, Wang Qin, Xia Yu. Facial age estimation by adaptive label distribution learning [C]//Proceedings of the 22nd International Conference on Pattern Recognition. Stockholm, Sweden: IEEE Press, 2014: 4465–4470.
- [6] Geng Xin, Xia Yu. Head pose estimation based on multivariate label distribution [C]//Proceedings of the 27th Conference on Computer Vision and Pattern Recognition. Columbus, OH, USA: IEEE Press, 2014: 1837–1842.
- [7] Zhou Ying, Xue Hui, Geng Xin. Emotion distribution recognition from facial expressions [C]//Proceedings of the 23rd ACM International Conference on Multimedia. New York, USA: ACM Press, 2015: 1247–1250.
- [8] Jia Xiuyi, Zheng Xiang, Li Weiwei, et al. Facial emotion distribution learning by exploiting low-rank label correlations locally [C]//Proceedings of the 32nd Conference on Computer Vision and Pattern Recognition. Long Beach, CA, USA: IEEE Press, 2019: 9841–9850.
- [9] Zhou Deyu, Zhang Xuan, Zhou Yin, et al. Emotion distribution learning from texts [C]//Proceedings of Conference on Empirical Methods in Natural Language Processing. Austin, Texas, USA: ACL Press, 2016: 638–647.
- [10] Geng Xin, Hou Peng. Pre-release prediction of crowd opinion on movies by label distribution learning [C]//Proceedings of the 24th International Conference on Artificial Intelligence. Buenos Aires, Argentina: IEEE Press, 2015: 3511–3517.
- [11] Ren Yi, Geng Xin. Sense beauty by label distribution learning [C]//Proceedings of the 26th International Joint Conference on Artificial Intelligence. Melbourne, Australia: IJCAI. Org, 2017: 2648–2654.
- [12] Xue Di, Hong Zheng, Guo Shize, et al. Personality recognition on social media with label distribution learning [J]. IEEE Access, 2017, 5: 13478–13488.
- [13] Zhang Zhaoxiang, Wang Mo, Geng Xin. Crowd counting in public video surveillance by label distribution learning [J]. Neurocomputing, 2015, 166: 151–163.
- [14] Sun Min, Zhang Dongping, Qian Leyi, et al. Crowd abnormal behavior detection based on label distribution learning [C]//Proceedings of the 4th International Conference on Intelligent Computation Technology and Automation. IEEE Press, 2015: 345–348.
- [15] Xing Chao, Geng Xin, Xue Hui. Logistic boosting regression for label distribution learning [C]//Proceedings of the 29th conference on computer vision and pattern recognition. Las Vegas, NV, USA: IEEE Press, 2016: 4489–4497.
- [16] Shen Wei, Zhao Kai, Guo Yilu, et al. Label distribution learning forests [J]//Proceedings of Advances in Neural Information Processing Systems. Long Beach, CA, USA: IEEE Press, 2017: 834–843.
- [17] Zhai Yansheng, Dai Jianhua, Shi Hong. Label distribution learning based on ensemble neural networks [C]//Proceedings of the 25th International Conference on Neural Information Processing. Siem Reap, Cambodia: Springer Press, 2018: 593–602.
- [18] Xu Miao, Li Yufeng, Zhou Zhihua. Multi-label learning with PRO loss [C]//Proceedings of the 27th AAAI Conference on Artificial Intelligence. Bellevue, Washington, USA: AAAI Press, 2013: 998–1004.
- [19] Zhang Minling, Zhou Zhihua. A review on multi-label learning algorithms [J]. IEEE Transactions on Knowledge and Data Engineering, 2014, 26(8): 1819–1837.
- [20] Jia Xiuyi, Li Weiwei, Liu Junyi, et al. Label distribution learning by exploiting label correlations [C]//Proceedings of the 32nd AAAI Conference on Artificial Intelligence. New Orleans, Louisiana, USA: AAAI Press, 2018: 3310–3317.
- [21] Ren Tingting, Jia Xiuyi, Li Weiwei, et al. Label distribution learning with label correlations via low-rank approximation [C]//Proceedings of the 28th International Joint Conference on Artificial Intelligence. Macao, China: IJCAI Org, 2019: 3325–3331.
- [22] Ren Tingting, Jia Xiuyi, Li Weiwei, et al. Label distribution learning with label-specific features [C]//Proceedings of the 28th International Joint Conference on Artificial Intelligence. Macao, China: IJCAI Org, 2019: 3318–3324.