

基于稀疏正则化的加权叠加集成多标签分类

肖建芳¹ 刘緬芳²

¹(汕头职业技术学院 广东 汕头 515041)

²(湖南科技大学数学与计算机科学学院 湖南 湘潭 411100)

摘要 为了充分挖掘成对标签的相关性以及分类器权重与分类器选择之间的关系,提出一种基于稀疏正则化的加权叠加集成多标签分类方法。提出一个稀疏正则化的加权叠加集成模型,以便于多标签分类器的选择和集成成员的构建。利用分类器权值和标签相关性来提高分类性能。进一步提出基于加速近端梯度和块坐标下降技术的优化算法来有效地获得最优解。在多个数据集上的实验结果表明,该方法能够有效实现较高精度的多标签分类。

关键词 多标签分类 相关性 稀疏正则化 权值

中图分类号 TP181

文献标志码 A

DOI:10.3969/j.issn.1000-386x.2024.05.042

WEIGHTED SUPERPOSITION ENSEMBLE MULTIPLE LABEL CLASSIFICATION BASED ON SPARSE REGULARIZATION

Xiao Jianfang¹ Liu Mianfang²

¹(Shantou Polytechnic, Shantou 515041, Guangdong, China)

²(School of Mathematics and Computational Science, Hunan University of Science and Technology, Xiangtan 411100, Hunan, China)

Abstract In order to fully exploit the correlation of paired labels and the relationship between classifier weight and classifier selection, a weighted superposition ensemble multiple label classification method based on sparse regularization is proposed. A sparse regularized weighted superposition ensemble model was proposed to facilitate the selection of multiple label classifiers and the construction of ensemble members. The classifier weight and label correlation were used to improve the classification performance. An optimization algorithm based on accelerated proximal gradient and block coordinate descent technique was proposed to obtain the optimal solution effectively. Experimental results on several data sets show that the proposed method can effectively achieve high precision multiple label classification.

Keywords Multiple label classification Correlation Sparse regularization Weight

0 引言

多标签学习被广泛应用于各个领域,例如文本分类、语义标注和医学诊断等,其中每个示例可以同时与多个类别标签相关联^[1-2]。与单标签分类任务不同的是,多标签分类可能受到内在潜在标签相关性的影响,因此其分类难度更大,受到了广泛的关注^[3]。

许多算法被提出来解决多标签分类问题,这些算法可分为两类,即问题转化方法和算法自适应方法。

第一类方法将多标签分类问题转化为多个独立的二分类问题或一个多分类问题。这一类的典型算法包括二进制相关(BR)、分类器链(CC)和标签功率集(LP)等^[4-6]。第二类方法扩展了特定的学习算法来处理多标签数据,文献[7]提出了一种加权分类器集成方法,该方法利用 MLKNN 中的距离得到的置信系数来调整 MLKN 的权值。文献[8]提出了一种基于秩 SVM 多标签分类算法。然而由于这些模型涉及到特定算法的属性,因此很难将这些方法推广到任何其他基础分类器中,并且上述算法仍然面临着标签相关性、类不平衡性

和高维性的挑战,这也导致了在实际应用中分类性能的不尽如人意。

近年来,集成学习算法通过将异构或同构模型中的个体学习者结合起来,获得一个联合学习模型,通过克服模型过度拟合和初始化敏感性提高学习持久性,引起了广泛的关注。文献[9]提出一种预测聚类树随机森林方法(RF-PCT),通过平均每个标签的所有分类器的置信值来预测一个新的实例。文献[10]提出一种基于加权叠加的多目标回归方法,该方法不考虑分类器的选择。文献[11]使用加权和规则(WS)和类相关加权和规则(CWS)进行多类分类。虽然上述方法在元层次上考虑了标签之间的全局相关性,但它仍然忽略了局部成对标签相关性的影响。此外,现有的叠加集成方法没有考虑分类器权重与分类器选择的关系。

为了解决上述问题,提出一种基于稀疏正则化的加权叠加集成多标签分类方法,有效解决了分类器选择问题,同时利用分类器权重和标签相关性来提高分类性能。进一步提出一种基于加速近端梯度和块坐标下降技术的优化算法来有效地获得最优解。实验证明了本文方法的有效性。

1 相关理论

1.1 置信度矩阵的生成

假设 $\mathbf{X} = \mathbf{R}^d$ 表示 d 维输入空间, $y = \{y_1, y_2, \dots, y_l\}$ 表示具有 l 个可能的类标签的标签空间。 $D = \{(\mathbf{x}_i, \mathbf{y}_i) \mid 1 \leq i \leq n\}$ 表示一个具有 N 个实例的训练数据集。 $(\mathbf{x}_i, \mathbf{y}_i), \mathbf{x}_i \in \mathbf{X}$ 是一个 D 维特征向量 $\mathbf{x}_i = [x_{i1}, x_{i2}, \dots, x_{id}]$ 和 $\mathbf{y}_i = [y_{i1}, y_{i2}, \dots, y_{il}]$ 是 \mathbf{x}_i 的真实标签。每个元素 $y_{ij} = 1$ 如果标签 \mathbf{y}_i 与 \mathbf{x}_i 相关,则 $y_{ij} = 0$ 。多标签学习的任务是学习一个来自训练集 D 函数 $h: \mathbf{X} \rightarrow 2^y$ 。对于任何隐藏样本 $\mathbf{x} \in \mathbf{X}$,多标签分类器 $h(\mathbf{x}) \subseteq y$ 作为适合 \mathbf{x} 的标签集。本文将输入数据表示为矩阵 $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n] \in \mathbf{R}^{n \times d}$, 输出标签矩阵表示为 $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_n]^T \in \mathbf{R}^{n \times l}$ 。

在具有置信度分数输出的分类器组合问题中,组合过程接受来自基本分类器的不同标签的预测分数作为其输入。 s_j^k 是从分类器 k 得到的标签 j 的数据实例的预测得分。设 $\mathbf{s}^k = [s_1^k, s_2^k, \dots, s_l^k]$ 是从分类器 k 得到的所有标签的预测得分,那么到组合器的输入是 $\mathbf{s} = [\mathbf{s}^1, \mathbf{s}^2, \dots, \mathbf{s}^m]$, 其中 m 是分类器的数量。假设 \mathbf{s}_i 包含从基本分类器获得的训练数据点 i 的得分,那么最终的置信得分矩阵是 $\mathbf{S} = [s_{ij}^k]$, 表示为:

$$\mathbf{S} = \begin{bmatrix} \overbrace{s_{11}^1 \quad s_{12}^1 \quad \dots \quad s_{1l}^1}^{\mathbf{s}^1} \quad \dots \quad \overbrace{s_{11}^k \quad s_{12}^k \quad \dots \quad s_{1l}^k}^{\mathbf{s}^k} \quad \dots \\ s_{21}^1 \quad s_{22}^1 \quad \dots \quad s_{2l}^1 \quad \dots \quad s_{21}^k \quad s_{22}^k \quad \dots \quad s_{2l}^k \quad \dots \\ \vdots \quad \vdots \quad \dots \quad \vdots \quad \dots \quad \vdots \quad \vdots \quad \dots \quad \vdots \quad \dots \\ s_{n1}^1 \quad s_{n2}^1 \quad \dots \quad s_{nl}^1 \quad \dots \quad s_{n1}^k \quad s_{n2}^k \quad \dots \quad s_{nl}^k \quad \dots \end{bmatrix} \quad (1)$$

1.2 加权分类器选择与叠加集成算法

在叠加集成框架中,将组合器定义为一个函数 $g: \mathbf{R}^{m \times l} \rightarrow \mathbf{R}^l$, 因此,本文目标是通过数据 $\{(s_i, \mathbf{y}_i)\}_{i=1}^n$ 来学习 g 函数。对于式(1)中给出的损失函数,需要最小化的目标函数如下:

$$g(\mathbf{w}^1, \mathbf{w}^2, \dots, \mathbf{w}^m) = \sum_{i=1}^n \sum_{j=1}^l \left(\sum_{k=1}^m (s_{ij}^k w_j^k - y_{ij}) \right)^2 \quad (2)$$

式中: w_j^k 表示标签 j 的分类器 k 的权重; $\mathbf{w}^k = [w_1^k, w_2^k, \dots, w_l^k]$ 是分类器 k 的权重向量。

令 $\mathbf{W}_j = [w_j^1 \mid w_j^2 \mid \dots \mid w_j^m]^T$ 表示第 j 个标签的所有分类器的组合权重向量, $\mathbf{Y}_j = [y_{1j}, y_{2j}, \dots, y_{nj}]^T$ 表示 \mathbf{Y} 的第 j 列, $1 \leq j \leq l$ 。在生成置信得分矩阵的基础上,式(2)中给出的目标函数可进一步改写为:

$$\min_{\mathbf{w}_j} \frac{1}{2} \|\mathbf{S}\mathbf{W}_j - \mathbf{Y}_j\|_2^2 \quad (3)$$

1.2.1 选择分类算法中的稀疏正则化

如上所述,生成置信度得分矩阵可能包含不利于标签预测的无关信息。为了结合分类器的选择,本文增加了一个正则化项来保证权重的稀疏性,以防止叠加集合并所有的基础分类器。使用稀疏正则化选择分类器的一个主要优点是,分类器是自动选择的,并且不预先指定所选分类器的数目。在这方面, l_1 范数正则化可以作为诱导稀疏性的最成功的方法。将 l_1 范数正则化引入到每个权重向量 \mathbf{W}_j 的模型中。结合式(3)中给出的最小二乘损失,分类器选择的目标函数可以表示为:

$$\min_{\mathbf{w}_j} \frac{1}{2} \|\mathbf{S}\mathbf{W}_j - \mathbf{Y}_j\|_2^2 + \alpha \|\mathbf{W}_j\|_1 \quad (4)$$

式中: α 是所有标签共享的正则化参数,用于平衡损失和正则化项,并且可以调整该参数以确定所选分类器的数目。此外,同时考虑所有二值分类器,式(4)可以改写为:

$$\min_{\mathbf{W}} \frac{1}{2} \|\mathbf{S}\mathbf{W} - \mathbf{Y}\|_F^2 + \alpha \|\mathbf{W}\|_1 \quad (5)$$

如果 $\mathbf{w}_j^k = 0$, 则表示消除第 k 个分类器,并且对第 j 个标签的预测是无用的,从而加速测试实例。然而, MLWSE 的 l_1 范数正则化的问题是,并非所有 \mathbf{w}_j^k 都为零,这意味着来自所选分类器的所有信息不能被有效地使用。受群稀疏正则化的启发,本文提出用群稀疏

正则化来解决这个问题。正则化和群稀疏正则化之间的区别如图 1 所示。正则化使许多置信度得分无用,并且相应的分类器权重为零。群组正则化只选择两个组(G2 和 G4),不选择其他两个组(G1 和 G3)。然而,在许多情况下,并非选定组中的所有信息都有用。群稀疏正则化利用了正则化和群正则化的优点,它首先选择群,然后再从所选群中进行另一个选择,即它同时考虑分类器内和分类器间的稀疏性。结合正则化和群正则化的正则化项,得到了群稀疏正则化的 MLWSE:

$$\min_{\mathbf{W}} \frac{1}{2} \|\mathbf{S}\mathbf{W} - \mathbf{Y}\|_F^2 + \alpha \|\mathbf{W}\|_1 + (1 - \alpha) \lambda \sum_{k=1}^m c_k \|\mathbf{W}_{G_k}\|_2 \quad (6)$$

式中: $\alpha \in [0, 1]$ 是正则化惩罚和群正则化惩罚的凸组合,稀疏性由调谐参数 λ 的大小决定; c_k 是第 k 个群 \mathbf{W}_{G_k} 的权重,可以定义为在分类器选择过程中第 k 个群的贡献之前的形式,令 $c_k = \sqrt{l}$ 。

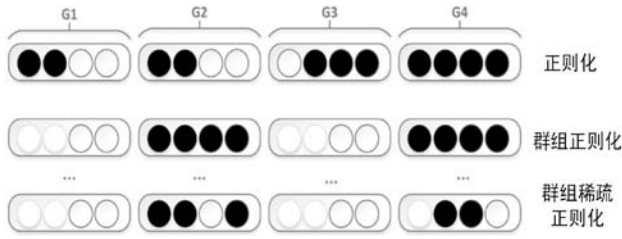


图 1 正则化和群稀疏正则化结构

1.2.2 类别相关性建模

标签相关性通常在多标签分类中起着关键作用,又因为多任务学习相关研究的兴起,该研究通过考虑任务或模式之间的相关性,在任务或模式之间共享相关信息。假设如果标签 y_j 和标签 y_k 是强相关的,那么对 y_j 有区别的分类器也可能对 y_k 有更高的区别概率。换言之,如果两个标签 y_j 和 y_k 是强相关的,则权重向量对 $(\mathbf{W}_j, \mathbf{W}_k)$ 应具有高相似度;否则,它们的相似度较低。本文在标签空间中构造一个图 $\langle V, E \rangle$, 其中: V 表示顶点/标签集, E 是包含每个标签对之间边的边集。给定 E 上的标签相关矩阵 \mathbf{R} , 最小化目标过程如下:

$$\frac{1}{2} \sum_{j=1}^l \sum_{k=1}^l \|\mathbf{W}_j - \mathbf{W}_k\|^2 R_{jk} = \text{tr}(\mathbf{W}(\mathbf{D} - \mathbf{R})\mathbf{W}^T) = \text{tr}(\mathbf{W}\mathbf{H}\mathbf{W}^T) \quad (7)$$

式中: $\mathbf{H} = \mathbf{D} - \mathbf{R}$ 是图拉普拉斯矩阵, \mathbf{D} 是满足 $D_{ij} = \sum_{k=1}^n \mathbf{R}_{ik}$ 的对角矩阵。 \mathbf{R}_{ik} 表示标签 y_j 和标签 y_k 之间的相似性。在本研究中,用余弦相似度来计算标签相关矩阵。通过将式(5)和式(7)结合,可以得到基于正则化的最终目标函数 MLWSE-L1:

$$\min_{\mathbf{W}} \frac{1}{2} \|\mathbf{S}\mathbf{W} - \mathbf{Y}\|_F^2 + \alpha \|\mathbf{W}\|_1 + \frac{\beta}{2} \text{tr}(\mathbf{W}\mathbf{H}\mathbf{W}^T) \quad (8)$$

通过将式(6)和式(7)结合,可以得到基于群稀疏正则化的最终目标函数 MLWSE-L21:

$$\min_{\mathbf{W}} \frac{1}{2} \|\mathbf{S}\mathbf{W} - \mathbf{Y}\|_F^2 + \alpha \lambda \|\mathbf{W}\|_1 + (1 - \alpha) \lambda \sum_{k=1}^m c_k \|\mathbf{W}_{G_k}\|_2 + \frac{\beta}{2} \text{tr}(\mathbf{W}\mathbf{H}\mathbf{W}^T) \quad (9)$$

在 MLWSE-L1 和 MLWSE-L21 中, α 和 β 是非负的平衡参数。在 MLWSE-L1 中,参数 α 控制模型的稀疏性,参数 β 平衡标签相关性和权重学习的影响程度。在 MLWSE-L21 中,参数 α 是与正则化和群正则化的平衡参数,参数 k 控制模型的稀疏性,而参数 β 平衡标签相关性和权重学习的贡献。

1.2.3 多标签预测

对 MLWSE-L1 和 MLWSE-L21 进行训练,得到分类器权重矩阵 \mathbf{W}^* 。给定由矩阵 \mathbf{X}^* 表示的测试数据,首先用不同的基础分类器生成置信度矩阵 \mathbf{S}^* ,然后用一个阈值函数符号 $\mathbf{X} \rightarrow \mathbf{R}$ 来确定预测标签。

$$\text{sign}(\mathbf{S} \times \mathbf{W}^*, \tau) = \begin{cases} 1 & \text{如果 } \mathbf{S} \times \mathbf{W}^* \geq \tau \\ 0 & \text{其他} \end{cases} \quad (10)$$

式中: τ 是阈值,令其为 0.5。

1.3 优化方法

1.3.1 MLWSE-L1 的优化

一般的快速近似梯度法可表述为以下凸优化问题:

$$\min_{\mathbf{W} \in H} \{F(\mathbf{W}) = f(\mathbf{W}) + g(\mathbf{W})\} \quad (11)$$

式中: H 是实 Hilbert 空间; $f(\mathbf{W})$ 是凸光滑的; $g(\mathbf{W})$ 是凸的,但可以是非光滑的。如果 $f(\mathbf{W})$ 有一个 Lipschitz 常数为 L 的 Lipschitz 连续梯度,即 $\|\nabla f(\mathbf{W}_1) - \nabla f(\mathbf{W}_2)\| \leq L\|\mathbf{W}_1 - \mathbf{W}_2\|$, 则近似梯度算法不必直接最小化 $F(\mathbf{W})$, 而是可以最小化其复合二次逼近。

$$Q_L(\mathbf{W}, \mathbf{W}^{(t)}) = f(\mathbf{W}^{(t)}) + \langle \nabla f(\mathbf{W}^{(t)}), \mathbf{W} - \mathbf{W}^{(t)} \rangle + \frac{L}{2} \|\mathbf{W} - \mathbf{W}^{(t)}\|_F^2 + g(\mathbf{W}) \quad (12)$$

根据式(8)和式(11), $f(\mathbf{W})$ 和 $g(\mathbf{W})$ 的定义如下:

$$f(\mathbf{W}) = \frac{1}{2} \|\mathbf{S}\mathbf{W} - \mathbf{Y}\|_F^2 + \frac{\beta}{2} \text{tr}(\mathbf{W}\mathbf{H}\mathbf{W}^T) \quad (13)$$

$$g(\mathbf{W}) = \alpha \lambda \|\mathbf{W}\|_1 \quad (14)$$

根据式(13), $\nabla f(\mathbf{W})$ 的计算公式为:

$$\nabla f(\mathbf{W}) = \mathbf{S}^T(\mathbf{S}\mathbf{W} - \mathbf{Y}) + \beta \mathbf{H}\mathbf{W} \quad (15)$$

对于 MLWSE-L1, 给定 \mathbf{W}_1 和 \mathbf{W}_2 , 可以得到 Lipschitz 常数:

$$L = \sqrt{\|\mathbf{S}^T \mathbf{S}_2^2 + 2\|\beta \mathbf{H}\|_2^2} \quad (16)$$

根据式(12)、式(14)和式(16), 权重矩阵 \mathbf{W} 可以通过式(17)进行优化。

$$\begin{aligned} \mathbf{W}^* &= \arg \min_{\mathbf{W}} Q_L(\mathbf{W}, \mathbf{W}^{(t)}) = \arg \min_{\mathbf{W}} \frac{1}{2} \|\mathbf{W} - \mathbf{Z}^{(t)}\|_F^2 + g(\mathbf{W}) = \\ & \arg \min_{\mathbf{W}} \frac{1}{2} \|\mathbf{W} - \mathbf{Z}^{(t)}\|_F^2 + \frac{\alpha}{2} \|\mathbf{W}\|_1 \end{aligned} \quad (17)$$

式中: $\mathbf{Z}^{(t)} = \mathbf{W}^{(t)} - \frac{1}{L} \nabla f(\mathbf{W}^{(t)})$ 。加速的近端梯度表明,对于满足 $b_t^2 - b_t \leq b_{t-1}^2$ 的序列 b_t , $\mathbf{W}^{(t)} = \mathbf{W}_t + \frac{b_{t-1} - 1}{b_t} (\mathbf{W}_t - \mathbf{W}_{t-1})$ 可以将收敛速度提高到 $O(1/t^2)$, 其中 \mathbf{W}_t 是第 t 次迭代时 \mathbf{W} 的结果。与式(17)的 $g(\mathbf{W})$ 相关联的近端算子是软阈值算子;然后,在每个迭代步骤中,可以通过以下优化问题获得 \mathbf{W} :

$$\mathbf{W}^{(t+1)} = \text{prox}_{\varepsilon}[\mathbf{Z}^{(t)}] = \arg \min_{\mathbf{W}} \frac{1}{2} \|\mathbf{W} - \mathbf{Z}^{(t)}\|_F^2 + \varepsilon \|\mathbf{W}\|_1 \quad (18)$$

式中: $\text{prox}_{\varepsilon}[\cdot]$ 是软阈值运算符。 $\text{prox}_{\varepsilon}[\cdot]$ 定义为:

$$\text{prox}_{\varepsilon}[w_{ij}] = \begin{cases} w_{ij} - \varepsilon & w_{ij} > \varepsilon \\ w_{ij} + \varepsilon & w_{ij} < -\varepsilon \\ 0 & \text{其他} \end{cases} \quad (19)$$

根据式(17)和式(19),可以通过以下软阈值函数得到 \mathbf{W} 。

$$\mathbf{W}^{(t+1)} = \text{prox}_{\frac{\alpha}{L}}[\mathbf{Z}^{(t)}] \quad (20)$$

算法 1 总结了 MLWSE-L1 的快速近似梯度的实现步骤。

算法 1 MLWSE-L1 的优化过程

输入: 训练数据矩阵 $\mathbf{X} \in \mathbf{R}^{n \times d}$, 标签矩阵 $\mathbf{Y} \in \mathbf{R}^{n \times l}$, 分类器 $\{C_i\}_{i=1}^m$, 参数 α, β, η 。

输出: 权重矩阵 $\mathbf{W}^* \in \mathbf{R}^{m \times l}$ 。

1. 由分类器 $\{C_i\}_{i=1}^m$ 生成置信度矩阵 $\mathbf{S} \in \mathbf{R}^{n \times ml}$ 。
2. 初始化: $b_0, b_1 \leftarrow 1; \mathbf{W}_0, \mathbf{W}_1 \leftarrow [\mathbf{S}^T \mathbf{S} + \eta \mathbf{I}]^{-1} \mathbf{S}^T \mathbf{Y}$
3. 计算 \mathbf{Y} 上的图拉普拉斯矩阵 \mathbf{H} ;
4. 根据式(16)计算 L
5. **While** 不收敛 **do**
6. $\mathbf{W}^{(t)} \leftarrow \mathbf{W}_t + \frac{b_{t-1} - 1}{b_t} (\mathbf{W}_t - \mathbf{W}_{t-1})$
7. 根据式(15)计算 $\nabla f(\mathbf{W}^{(t)})$
8. $\mathbf{Z}^{(t)} \leftarrow \mathbf{W}^{(t)} - \frac{1}{L} \nabla f(\mathbf{W}^{(t)})$
9. $\mathbf{W}^{(t+1)} \leftarrow \text{prox}_{\frac{\alpha}{L}}[\mathbf{Z}^{(t)}]$
10. $b_{t+1} \leftarrow \frac{1 + \sqrt{4b_t^2 + 1}}{2}$
11. $t \leftarrow t + 1$
12. **return** $\mathbf{W}^* \leftarrow \mathbf{W}^{(t+1)}$

1.3.2 MLWSE-L21 的优化

本文使用块坐标下降法来优化 MLWSE-L21, 主要包括两个组成部分: 不同特征组上的外循环和解决每个块子问题的内循环。式(9)中给出的置信得分矩阵

\mathbf{S} 可分解为 m 组, $\mathbf{S}^1, \mathbf{S}^2, \dots, \mathbf{S}^m$, 每个 $\mathbf{S}^k \in \mathbf{R}^{n \times l}$ 。设 \mathbf{S}^{-k} 表示其余的群, 当 \mathbf{S} 与群 k 相关联时, \mathbf{W}^k 是 \mathbf{W} 在其他群上的分量。

当选择 k 组时, 当前 \mathbf{W} 的其他组是固定的, 并且目标函数仅在 \mathbf{W}^k 上最小化, 然后最小化每个块。

$$\begin{aligned} & \frac{1}{2} \|\mathbf{r}_{-k} - \mathbf{S}^{(k)} \mathbf{W}^{(k)}\|_2^2 + (1 - \alpha) \lambda c_k \|\mathbf{W}^{(k)}\|_2 + \\ & \alpha \lambda \|\mathbf{W}^{(k)}\|_1 + \frac{\beta}{2} \text{tr}(\mathbf{W}^{(k)} \mathbf{H} \mathbf{W}^{(k)T}) \end{aligned} \quad (21)$$

式中: \mathbf{r}_{-k} 是 \mathbf{Y} 的部分残差。减去除 k 组以外的所有组拟合:

$$\mathbf{r}_{-k} = \|\mathbf{Y} - \sum_{j \neq k} \mathbf{S}^{(j)} \mathbf{W}^{(j)}\| \quad (22)$$

设 $\ell(\mathbf{r}_{-k}, \mathbf{W}^{(k)}) = \frac{1}{2} \|\mathbf{r}_{-k} - \mathbf{S}^{(k)} \mathbf{W}^{(k)}\|_2^2$ 表示最小二乘损失函数, $\nabla \ell(\mathbf{r}_{-k}, \mathbf{W}^{(k)})$ 表示其梯度, 则本文目标是求 $\mathbf{W}_*^{(k)}$ 从而最小化式(21), 那么相当于最小化式(23), 以 $\mathbf{W}_0^{(k)}$ 点为中心。

$$\begin{aligned} & \frac{1}{2t} \|\mathbf{W}^{(k)} - (\mathbf{W}_0^{(k)} - t \nabla \ell(\mathbf{r}_{-k}, \mathbf{W}_0^{(k)}))\|_2^2 + \\ & (1 - \alpha) \lambda c_k \|\mathbf{W}^{(k)}\|_2 + \alpha \lambda \|\mathbf{W}^{(k)}\|_1 + \frac{\beta}{2} \text{tr}(\mathbf{W}^{(k)} \mathbf{H} \mathbf{W}^{(k)T}) \end{aligned} \quad (23)$$

式中: t 表示梯度步长。由参考文献[12]可知, 如果满足式(24), 那么 $\mathbf{W}_*^{(k)} = 0$ 。

$$\|\zeta(\mathbf{W}_0^{(k)} - t \nabla \ell(\mathbf{r}_{-k}, \mathbf{W}_0^{(k)}), t \alpha \lambda)\|_2 \leq t(1 - \alpha) \lambda c_k \quad (24)$$

否则 $\mathbf{W}_*^{(k)}$ 满足:

$$\left(1 - \frac{t(1 - \alpha) \lambda c_k}{\|\zeta(\mathbf{W}_0^{(k)} - t \nabla \ell(\mathbf{r}_{-k}, \mathbf{W}_0^{(k)}), t \alpha \lambda)\|_2}\right)_+ \zeta(\mathbf{W}_0^{(k)} - t \nabla \ell(\mathbf{r}_{-k}, \mathbf{W}_0^{(k)}), t \alpha \lambda) \quad (25)$$

式中: $\zeta(\cdot)$ 表示软阈值函数。

$$(\zeta(z, t \alpha \lambda))_i = \text{sign}(z_i) (|z_i| - t \alpha \lambda)_+ \quad (26)$$

算法 2 中总结了 MLWSE-L21 的块坐标下降的细节。

算法 2 MLWSE-L21 的优化过程

输入: 训练数据矩阵 $\mathbf{X} \in \mathbf{R}^{n \times d}$, 标签矩阵 $\mathbf{Y} \in \mathbf{R}^{n \times l}$, 分类器 $\{C_i\}_{i=1}^m$, 参数 $\alpha, \beta, \lambda, \eta$ 。

输出: 权重矩阵 $\mathbf{W}^* \in \mathbf{R}^{m \times l}$ 。

1. 由分类器 $\{C_i\}_{i=1}^m$ 生成置信度矩阵 $\mathbf{S} \in \mathbf{R}^{n \times ml}$
2. 计算 \mathbf{Y} 上的图拉普拉斯矩阵 \mathbf{H}
3. 根据式(16)计算 L
4. 根据式(22)计算 \mathbf{r}_{-k}
5. 循环遍历各个组的数据; 在每个组 (k) 上执行步骤 6
6. 初始化: $t \leftarrow 1/L, \mathbf{W}^{(k)} \leftarrow [\mathbf{S}^T \mathbf{S} + \eta \mathbf{I}]^{-1} \mathbf{S}^T \mathbf{Y}$
7. 根据式(24)检查是否满足 $\mathbf{W}^{(k)} = 0$, 否则在各个组中执行步骤 8

8. **While** 不收敛 **do**
9. 更新梯度 $\nabla \ell(r_{-k}, W_0^{(k)})$
10. 根据式(25)更新 $W^{(k+1)}$
11. **return** $W^* \leftarrow W^{(t+1)}$

2 实验

2.1 实验数据集

2.1.1 二维合成数据

为了评估加权分类器的选择能力,在四个不同分布场景的二维合成数据集上进行实验。由于多标签分类问题可以转化为多个独立的单标签分类问题,只需要考虑单标签分类的情况。所有四个场景如图 3 所示,包括从 $[-4, +4]$ 中均匀分布得出的单变量 X ,结果如下:

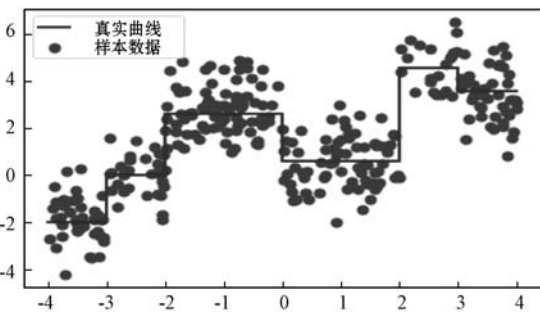
场景一: $Y = -2 \times I(X < -3) + 2.55 \times I(X > -2) - 2 \times I(X > 0) + 4 \times I(X > 2) - 1 \times I(X > 3) + N(0, 1)$

场景二: $Y = 5 + 0.4X - 0.36X^2 + 0.005X^3 + N(0, 1)$

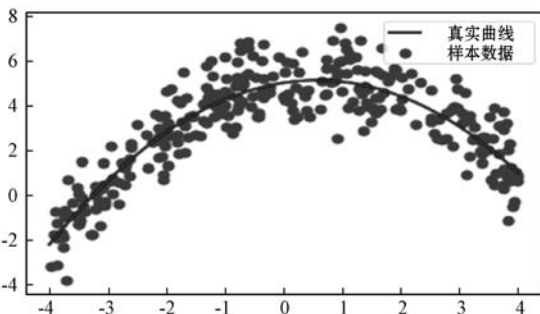
场景三: $Y = 2.85 \times \sin\left(\frac{\pi}{2} \times X\right) + N(0, 1)$

场景四: $Y = 3.85 \times \sin(3\pi \times X) \times I(X > 0) + N(0, 1)$

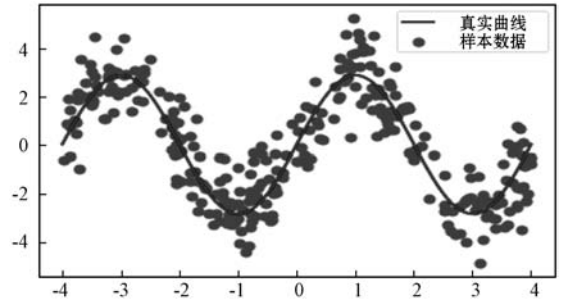
其中: $I(\cdot)$ 是常用的指标函数, $N(0, 1)$ 是每个场景的独立标准正态分布。如图 2 所示,它包含每个模拟 300 个样本的散点图,每个场景的真实曲线用红线表示。因为这些场景代表不同的真实模型,因此本文选用了这些场景做实验。利用这些二维合成数据集,可以评估本文方法的加权分类器选择能力。



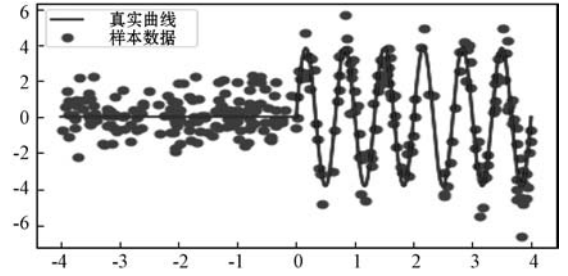
(a) 场景一



(b) 场景二



(c) 场景三



(d) 场景四

图 2 四个场景散点图

2.1.2 典型数据集

为了验证本文方法的性能,在 13 个热门的多标签基准数据集上进行评估。数据集相关介绍如表 1 所示。 $L_c = \frac{1}{N} \sum_{i=1}^N |Y_i|$ 表示标签基数,即与每个实例相关联的标签的平均数量。利用这些基准多标签数据集,比较了本文方法与七种最先进的集成多标签分类方法。

表 1 典型数据集的介绍

数据集	涉及领域	实例数	特征数	标签数	L_c
Emotions	音乐	593	72	6	1.868
Flags	图像	194	19	7	3.392
Scene	图像	2 407	294	6	1.074
Yeast	生物	2 417	103	14	4.237
Birds	视频	645	260	19	1.014
GpositiveGO	生物	519	912	4	1.008
CHD-49	药物	555	49	6	2.580
Enron	文本	1 702	1 001	53	3.378
Langlog	文本	1 460	1 004	75	1.180
Medical	文本	978	1 449	45	1.245
VirusGo	生物	207	749	6	1.217
Water-qy	化学	1 060	16	14	5.073
3s-bbc1000	文本	352	1 000	6	1.125

2.1.3 真实数据集

为了探索本文方法的潜在应用领域,将本文方法应用于一个真实的心脑血管疾病 (CCD) 数据集,以证明其在医学诊断中的实际应用潜力,并将 CCD 数据集

作为另一个基准数据集来运行实验。数据集采集自中国云南省一家医院的心脑血管病患者。它包含 3 823 个样本、59 个特征和 9 个标签,其中 9 个标签是缺血性脑卒中(CIS)、脑出血(CH)、蛛网膜下腔出血(SAH)、脑静脉血栓形成(CVT)、颅内动脉瘤(IA)、脑血管畸形(CVM)、心脏病(HD)、糖尿病(DM)和高血压(HT)。表 2 中列出了每个标签对应的样本数。

表 2 样本尺寸与 CCD 数据集中不同标签的对应关系

标签	样本数	标签频率
CIS	3 380	0.884
CH	140	0.036
SAH	134	0.035
CVT	8	0.002
IA	23	0.006
CVM	20	0.005
HD	1 133	0.296
DM	920	0.240
HT	2 513	0.657

2.2 实验参数设置

2.2.1 基本方法

比较了本文方法 MLWSE 与七个经典的集成多标签分类方法:

EBR^[6]:BR 模型的集成版本。每个 BR 的实例是随机生成的,不考虑标签之间的关系。

ECC^[7]:CC 的集成版本,其中每个 CC 的链顺序是随机生成的,考虑了全局标签相关性。

EPS^[8]:LP 改进的集成版本,它通过剔除不经常出现的标签集来关注标签之间最重要的关系。

RAkEL^[9]:基于 k 个标签的随机小子集的 LP 的改进的集成版本,其中将 k 设为 3,它考虑一小部分的标签关系。

CDE^[10]:Chi-Dep 算法的集合,它为每一组相关标签建立一个 LP 分类器,标签集的分区设置为 100,考虑了不同组之间的标签关系。

AdaBoost.MH^[13]:基于 BR 的加权集成版本,它不仅像 AdaBoost 那样在实例样本上仅生成一组权重,而且在标签上仅生成一组权重。

MLS^[14]:基于 BR 的叠加集成版本,它考虑了元级别中标签之间的关系。

2.2.2 超参数设置

所有的基本方法都是使用 Mulan 和 Meka 框架实现的,这两个框架提供了 Java 的 API。这些方法在集合中使用了 $n = 10$ 个分类器,阈值为 $\tau = 0.5$,并且将

C4.5 决策树作为单标签基分类器。

在本文实验中,置信度矩阵 S 是通过使用 BR、CC 和 LP 生成的,其中 SVM 用于单标签基分类器,其他参数参考 Scikit 多类学习中的默认参数^[15]。组数 m 设置为 3。对于 MLWSE-L1, α 和 β 的取值范围为 $\{10^{-5}, 10^{-4}, \dots, 10^3, 10^4\}$; η 的取值范围为 $\{0.1, 1\}$ 。对于 MLWSE-L21, α 的取值范围为 $\{0.01, 0.05, 0.1, 0.15, 0.2\}$; β 的取值范围为 $\{10^{-4}, 10^{-3}, \dots, 10^1, 10^2\}$; λ 的取值范围为 $\{10^{-5}, 10^{-4}, \dots, 10^1, 10^2\}$; η 的取值范围为 $\{0.1, 1\}$ 。

2.2.3 评估指标

本文使用六种常用的评估指标来评估多标签分类的性能。一般来说,它们可以分为两组,即基于示例的度量(汉明损失、准确度、Ranking 损失和 F1)和基于标签的度量(Macro B(h)和 Micro B(h))。对于每个评价指标,测试数据集定义为 $D_l = \{(\mathbf{x}_i, \mathbf{y}_i) \mid 1 \leq i \leq n\}$,其中 $\mathbf{y}_i \in \{0, 1\}^l$ 是第 i 个测试样本的实际标签,并且 $\hat{y}_i = h(\mathbf{x}_i)$ 是预测的标签。

Hamming 损失:计算错误分类的示例标签对的分数,损失越小,分类器的性能越准确, $H_{\text{hamming loss}} = \frac{1}{n} \sum_{i=1}^n \frac{1}{l} \sum_{j=1}^l I(y_{ij} \neq \hat{y}_{ij})$,其中 $I(\cdot)$ 是一个指示函数,如果 $I(\cdot)$ 保持不变返回 1,否则返回 0。

准确度:评估真实标签和预测标签之间的相似性,

$$A_{\text{accuracy}} = \frac{1}{n} \sum_{i=1}^n \frac{|y_i \cap \hat{y}_i|}{|y_i \cup \hat{y}_i|}$$

Ranking 损失:评估反向排列的标签对的分数,即当一个不相关的标签排名高于一个相关的标签。

$R_{\text{ranking loss}} = \frac{1}{n} \sum_{i=1}^n \frac{1}{|y_i| |\bar{y}_i|} \left| \{(y_1, y_2) \mid f(x_i, y_1) \leq f(x_i, y_2), (y_1, y_2) \in y_i \times \bar{y}_i\} \right|$,其中: $f(x, y)$ 为 $y \in Y$ 的置信度得分, $y \in Y$ 是 x 的正确标签。

F1:查全率和查准率的调和平均值,其中 p_i 和 q_i 是第 i 个示例的查全率和查准率, $F_1 = \frac{1}{n} \sum_{i=1}^n \frac{2p_i q_i}{p_i + q_i}$,基于标签的分类度量可以通过以下任一情况得到:

$$\text{Macro-B}(h) = \frac{1}{l} \sum_{j=1}^l B(T_{p_j}, F_{p_j}, T_{N_j}, F_{N_j})$$

$$\text{Micro-B}(h) = B\left(\sum_{j=1}^l T_{p_j}, \sum_{j=1}^l F_{p_j}, \sum_{j=1}^l T_{N_j}, \sum_{j=1}^l F_{N_j}\right) \quad (27)$$

式中: $T_{p_j}, F_{p_j}, T_{N_j}, F_{N_j}$ 表示与标签 y_j 有关的真阳性、假阳性、真阴性和假阴性测试样本的数量。 $B(T_{p_j}, F_{p_j}, T_{N_j}, F_{N_j})$ 表示某些特定的二进制分类度量。 $\text{Macro-B}(h)$ 和 $\text{Micro-B}(h)$ 分别假定标签和样本的权重相等。

2.3 实验结果

2.3.1 二维合成数据集结果

使用二维合成数据集,通过逐渐添加不同的元素来评估本文方法的加权分类器选择能力,包括式(3)中

给出的加权设置作为基线方法,式(5)中给出的正则化选择,式(6)中给出的群稀疏选择。将每个数据集随机分成一个训练集、一个验证集(35%)和一个测试集(30%)。四种情况下的实验结果如表3所示。

表3 基于合成数据集的实验结果

算法		场景一		场景二		场景三		场景四	
		精度	权重向量	精度	权重向量	精度	权重向量	精度	权重向量
基础分类算法	SVM (linear)	0.522 2	—	0.444 4	—	0.533 3	—	0.766 7	—
	SVM (poly)	0.533 3	—	0.466 7	—	0.533 3	—	0.766 7	—
	Random Forest	0.522 2	—	0.466 7	—	0.833 3	—	0.7111	—
基本算法	SVM (linear)	0.488 9	0.678 5	0.500 0	-0.927 4	0.800 0	-59.118 3	0.766 7	-30.116 6
	SVM (poly)		1.620 0		2.185 8		56.915 3		30.060 6
正则化选择	Random Forest	0.522 2	0.036 4	0.500 0	-0.197 4	0.833 3	0.937 2	0.766 7	0.1643
	SVM (linear)		0.339 1		-0.852 1		-0.018 4		0.0715
	SVM (poly)		0.589 4		2.104 6		0.1053		0.660 3
群稀疏正则化选择	Random Forest	0.533 3	0.052 1	0.511 1	-0.197 0	0.833 3	0.919 7	0.766 7	0.277 1
	SVM (linear)		0		0		0		0
	SVM (poly)		0.932 6		1.171 6		0.079 7		0.731 7
	Random Forest		0.044 5		-0.181 1		0.922 6		0.273 7

在场景一中,三种基本分类器的性能都很好,其中基于 SVM 的多核分类器的性能最好。正则化选择法和群稀疏正则化选择法的精度分别为 0.522 2 和 0.533 3,对应的权重向量分别为 $\langle 0.339 1, 0.589 4, 0.052 1 \rangle$ 和 $\langle 0, 0.932 6, 0.044 5 \rangle$,均高于基线方法。结果表明,与基线方法相比,正则化选择和群稀疏正则化选择学习的权值最优,从而为更有效的基础分类器赋予更高的权值。场景二中,基于 SVM 的多核分类器的性能最好。类似地,正则化选择和群稀疏正则化选择能够适应底层结构以获得更高的精度。在场景三和场景四中也表现出同样的趋势,其中加权分类器选择方法的性能几乎与单个最佳算法相当,甚至优于单个最佳算法。单个最佳算法是未知的,特别是在多标签设置中,其中基础分类器的性能可能在某些标签上有所提高,而在其他标签上的性能有所下降。然而,本

文算法可以自适应地学习最优权值来选择基础分类器,并适应真实的变化场景。结果表明,本文提出的加权分类器选择方法对提高多标签分类性能是有效的。

2.3.2 基准数据集结果

表4和表5比较了本文方法在 MLWSE-L1 和 MLWSE-L21 上与 13 个数据集上的最新算法的结果分析。对于每个数据集,MLWSE-L1 的参数 α 、 β 和 η 分别设置为 10^{-4} 、 10^{-3} 和 0.1, MLWSE-L21 的参数 α 、 λ 、 β 和 η 分别设置为 0.05、 10^{-3} 、 10^{-2} 和 0.1。每个算法执行五次验证,并计算每个指标的平均值和标准差。在最大的数据集中,一些算法不能用到所有的数据集,将这些情况记为 DNF。所有参与比较的算法中最好的结果用粗体突出显示,根据这些结果,可以得到以下结论。

表4 各算法(均值/标准差)在精确度、汉明损失和排名损失方面的实验比较结果

指标	数据集	EBR	ECC	EPS	RAKEL	CDE	AdaBoost. MH	MLS	MLWSE-L1	MLWSE-L21
精度	Emotions	0.517 ± 0.034	0.532 ± 0.039	0.533 ± 0.021	0.422 ± 0.028	0.524 ± 0.035	0.028 ± 0.016	0.422 ± 0.028	0.806 ± 0.007	0.807 ± 0.007
	Flags	0.598 ± 0.067	0.630 ± 0.067	0.590 ± 0.063	0.607 ± 0.051	0.609 ± 0.077	0.514 ± 0.064	0.607 ± 0.051	0.727 ± 0.014	0.743 ± 0.014
	Scene	0.605 ± 0.008	0.659 ± 0.013	0.642 ± 0.007	0.534 ± 0.017	0.538 ± 0.004	0.000 ± 0.000	0.534 ± 0.017	0.917 ± 0.001	0.915 ± 0.003
	Yeast	0.489 ± 0.014	0.505 ± 0.008	0.491 ± 0.015	0.434 ± 0.012	0.478 ± 0.008	0.335 ± 0.015	0.434 ± 0.012	0.804 ± 0.002	0.801 ± 0.002

续表 4

指标	数据集	EBR	ECC	EPS	RAkEL	CDE	AdaBoost. MH	MLS	MLWSE-L1	MLWSE-L21
精度	Birds	0.593 ± 0.021	0.602 ± 0.018	0.589 ± 0.015	0.568 ± 0.036	0.588 ± 0.039	0.456 ± 0.015	0.568 ± 0.036	0.949 ± 0.003	0.955 ± 0.002
	GpositiveGO	0.933 ± 0.011	0.929 ± 0.016	0.937 ± 0.008	0.930 ± 0.017	0.928 ± 0.018	0.000 ± 0.000	0.930 ± 0.017	0.971 ± 0.003	0.971 ± 0.005
	CHD-49	0.515 ± 0.020	0.533 ± 0.025	0.531 ± 0.022	0.470 ± 0.018	0.490 ± 0.031	0.464 ± 0.008	0.470 ± 0.018	0.706 ± 0.011	0.703 ± 0.013
	Enron	0.425 ± 0.015	0.467 ± 0.019	0.376 ± 0.020	0.414 ± 0.012	0.411 ± 0.013	0.151 ± 0.009	0.414 ± 0.012	0.953 ± 0.001	0.954 ± 0.000
	Langlog	0.232 ± 0.027	0.237 ± 0.023	0.231 ± 0.024	0.250 ± 0.026	DNF	0.142 ± 0.022	0.821 ± 0.019	0.820 ± 0.003	0.830 ± 0.001
	Medical	0.755 ± 0.024	0.767 ± 0.025	0.754 ± 0.024	0.752 ± 0.033	0.718 ± 0.040	0.000 ± 0.000	0.752 ± 0.033	0.986 ± 0.001	0.987 ± 0.000
	VirusGo	0.861 ± 0.058	0.859 ± 0.056	0.872 ± 0.043	0.861 ± 0.058	0.872 ± 0.058	0.000 ± 0.000	0.861 ± 0.058	0.956 ± 0.003	0.956 ± 0.005
	Water-qy	0.393 ± 0.007	0.414 ± 0.010	0.204 ± 0.019	0.318 ± 0.010	0.402 ± 0.006	0.157 ± 0.030	0.374 ± 0.007	0.715 ± 0.004	0.707 ± 0.007
3s-bbc1000	0.044 ± 0.010	0.123 ± 0.027	0.195 ± 0.027	0.144 ± 0.027	0.144 ± 0.019	0.000 ± 0.000	0.144 ± 0.027	0.805 ± 0.006	0.810 ± 0.005	
汉明损失	Emotions	0.197 ± 0.015	0.205 ± 0.016	0.211 ± 0.015	0.264 ± 0.018	0.212 ± 0.019	0.306 ± 0.010	0.264 ± 0.018	0.194 ± 0.007	0.193 ± 0.007
	Flags	0.249 ± 0.044	0.243 ± 0.045	0.258 ± 0.041	0.253 ± 0.036	0.258 ± 0.052	0.278 ± 0.026	0.253 ± 0.036	0.273 ± 0.014	0.241 ± 0.014
	Scene	0.093 ± 0.003	0.094 ± 0.004	0.099 ± 0.003	0.135 ± 0.007	0.136 ± 0.003	0.179 ± 0.002	0.135 ± 0.007	0.083 ± 0.001	0.085 ± 0.003
	Yeast	0.205 ± 0.006	0.210 ± 0.004	0.212 ± 0.007	0.248 ± 0.008	0.228 ± 0.006	0.232 ± 0.007	0.249 ± 0.008	0.197 ± 0.002	0.199 ± 0.002
	Birds	0.042 ± 0.003	0.043 ± 0.004	0.046 ± 0.002	0.051 ± 0.006	0.047 ± 0.006	0.053 ± 0.002	0.051 ± 0.006	0.051 ± 0.003	0.045 ± 0.001
	GpositiveGO	0.027 ± 0.004	0.030 ± 0.009	0.031 ± 0.005	0.027 ± 0.006	0.031 ± 0.009	0.255 ± 0.007	0.027 ± 0.006	0.029 ± 0.003	0.020 ± 0.005
	CHD-49	0.299 ± 0.013	0.304 ± 0.020	0.307 ± 0.016	0.325 ± 0.013	0.323 ± 0.022	0.307 ± 0.004	0.325 ± 0.013	0.294 ± 0.011	0.290 ± 0.013
	Enron	0.048 ± 0.001	0.048 ± 0.002	0.052 ± 0.002	0.051 ± 0.001	0.051 ± 0.001	0.062 ± 0.001	0.051 ± 0.001	0.047 ± 0.001	0.043 ± 0.001
Langlog	0.016 ± 0.001	0.016 ± 0.001	0.016 ± 0.001	0.020 ± 0.002	DNF	0.016 ± 0.001	0.037 ± 0.002	0.018 ± 0.003	0.011 ± 0.001	
Medical	0.010 ± 0.001	0.010 ± 0.001	0.012 ± 0.001	0.010 ± 0.001	0.012 ± 0.001	0.028 ± 0.001	0.010 ± 0.001	0.014 ± 0.001	0.010 ± 0.000	
VirusGo	0.045 ± 0.012	0.045 ± 0.014	0.047 ± 0.019	0.042 ± 0.017	0.042 ± 0.019	0.203 ± 0.013	0.042 ± 0.017	0.044 ± 0.003	0.040 ± 0.005	
Water-qy	0.293 ± 0.009	0.295 ± 0.009	0.323 ± 0.002	0.329 ± 0.004	0.303 ± 0.010	0.338 ± 0.008	0.311 ± 0.005	0.286 ± 0.004	0.293 ± 0.007	
3s-bbc1000	0.209 ± 0.011	0.223 ± 0.012	0.206 ± 0.010	0.251 ± 0.029	0.250 ± 0.013	0.188 ± 0.008	0.251 ± 0.029	0.195 ± 0.006	0.190 ± 0.005	
Ranking 损失	Emotions	0.171 ± 0.019	0.171 ± 0.013	0.196 ± 0.015	0.316 ± 0.031	0.176 ± 0.019	0.427 ± 0.029	0.326 ± 0.036	0.159 ± 0.013	0.149 ± 0.011
	Flags	0.201 ± 0.032	0.217 ± 0.041	0.220 ± 0.051	0.318 ± 0.042	0.256 ± 0.060	0.238 ± 0.034	0.272 ± 0.035	0.233 ± 0.021	0.200 ± 0.011
	Scene	0.079 ± 0.009	0.092 ± 0.009	0.101 ± 0.008	0.195 ± 0.015	0.138 ± 0.010	0.472 ± 0.013	0.227 ± 0.021	0.068 ± 0.003	0.069 ± 0.003
	Yeast	0.185 ± 0.010	0.191 ± 0.010	0.202 ± 0.008	0.336 ± 0.015	0.219 ± 0.009	0.363 ± 0.029	0.316 ± 0.012	0.171 ± 0.001	0.168 ± 0.001
	Birds	0.098 ± 0.012	0.111 ± 0.013	0.140 ± 0.014	0.199 ± 0.026	0.134 ± 0.015	0.229 ± 0.037	0.168 ± 0.012	0.120 ± 0.008	0.110 ± 0.003
	GpositiveGO	0.025 ± 0.005	0.027 ± 0.008	0.031 ± 0.011	0.034 ± 0.012	0.029 ± 0.012	0.301 ± 0.019	0.025 ± 0.006	0.026 ± 0.005	0.024 ± 0.004
	CHD-49	0.222 ± 0.015	0.230 ± 0.020	0.226 ± 0.021	0.313 ± 0.014	0.255 ± 0.027	0.222 ± 0.011	0.313 ± 0.020	0.215 ± 0.006	0.210 ± 0.007
	Enron	0.085 ± 0.008	0.150 ± 0.014	0.161 ± 0.011	0.302 ± 0.011	0.198 ± 0.001	0.240 ± 0.011	0.175 ± 0.005	0.105 ± 0.003	0.092 ± 0.007
Langlog	0.121 ± 0.005	0.273 ± 0.017	0.291 ± 0.013	0.413 ± 0.011	DNF	0.470 ± 0.015	0.166 ± 0.039	0.248 ± 0.005	0.230 ± 0.004	
Medical	0.031 ± 0.003	0.042 ± 0.011	0.057 ± 0.011	0.097 ± 0.016	0.074 ± 0.005	0.285 ± 0.010	0.070 ± 0.016	0.033 ± 0.009	0.025 ± 0.004	
VirusGo	0.030 ± 0.015	0.033 ± 0.015	0.030 ± 0.017	0.067 ± 0.055	0.043 ± 0.018	0.264 ± 0.045	0.042 ± 0.025	0.031 ± 0.004	0.032 ± 0.005	
Water-qy	0.253 ± 0.006	0.256 ± 0.006	0.347 ± 0.007	0.368 ± 0.007	0.275 ± 0.005	0.374 ± 0.011	0.325 ± 0.006	0.247 ± 0.008	0.262 ± 0.006	
3s-bbc1000	0.404 ± 0.034	0.417 ± 0.031	0.383 ± 0.037	0.497 ± 0.035	0.434 ± 0.003	0.422 ± 0.027	0.497 ± 0.058	0.381 ± 0.020	0.389 ± 0.025	

表 5 各算法(均值/标准差)在 F1、Macro-F1 和 Micro-F1 方面获得的实验比较结果

指标	数据集	EBR	ECC	EPS	RAKEL	CDE	AdaBoost. MH	MLS	MLWSE-L1	MLWSE-L21
F1	Emotions	0.597 ± 0.037	0.612 ± 0.037	0.615 ± 0.018	0.509 ± 0.036	0.608 ± 0.031	0.037 ± 0.020	0.509 ± 0.036	0.639 ± 0.024	0.614 ± 0.014
	Flags	0.711 ± 0.057	0.735 ± 0.050	0.699 ± 0.049	0.721 ± 0.043	0.721 ± 0.065	0.631 ± 0.063	0.721 ± 0.043	0.700 ± 0.020	0.721 ± 0.025
	Scene	0.620 ± 0.007	0.675 ± 0.014	0.655 ± 0.006	0.573 ± 0.016	0.573 ± 0.009	0.000 ± 0.000	0.573 ± 0.016	0.708 ± 0.005	0.672 ± 0.010
	Yeast	0.599 ± 0.014	0.611 ± 0.007	0.599 ± 0.013	0.556 ± 0.012	0.595 ± 0.007	0.456 ± 0.019	0.556 ± 0.012	0.647 ± 0.006	0.625 ± 0.004
	Birds	0.618 ± 0.022	0.631 ± 0.016	0.616 ± 0.019	0.603 ± 0.037	0.621 ± 0.040	0.456 ± 0.015	0.603 ± 0.037	0.152 ± 0.024	0.153 ± 0.009
	GpositiveGO	0.938 ± 0.012	0.931 ± 0.017	0.940 ± 0.008	0.934 ± 0.018	0.933 ± 0.018	0.000 ± 0.000	0.934 ± 0.018	0.945 ± 0.009	0.949 ± 0.008
	CHD-49	0.628 ± 0.022	0.643 ± 0.024	0.643 ± 0.016	0.587 ± 0.016	0.610 ± 0.032	0.580 ± 0.007	0.587 ± 0.016	0.659 ± 0.008	0.663 ± 0.016
	Enron	0.537 ± 0.015	0.579 ± 0.017	0.472 ± 0.020	0.525 ± 0.012	0.523 ± 0.012	0.231 ± 0.013	0.525 ± 0.012	0.578 ± 0.011	0.576 ± 0.006
	Langlog	0.239 ± 0.026	0.246 ± 0.020	0.236 ± 0.024	0.267 ± 0.025	DNF	0.142 ± 0.022	0.115 ± 0.026	0.487 ± 0.004	0.496 ± 0.002
	Medical	0.785 ± 0.025	0.795 ± 0.026	0.779 ± 0.024	0.783 ± 0.031	0.751 ± 0.043	0.000 ± 0.000	0.783 ± 0.031	0.773 ± 0.015	0.779 ± 0.011
VirusGo	0.883 ± 0.057	0.879 ± 0.055	0.893 ± 0.037	0.880 ± 0.056	0.893 ± 0.047	0.000 ± 0.000	0.880 ± 0.056	0.913 ± 0.008	0.925 ± 0.013	
Water-qy	0.532 ± 0.007	0.556 ± 0.011	0.299 ± 0.022	0.452 ± 0.011	0.543 ± 0.006	0.244 ± 0.043	0.513 ± 0.006	0.550 ± 0.009	0.561 ± 0.011	
3s-bbc1000	0.047 ± 0.012	0.128 ± 0.027	0.207 ± 0.028	0.162 ± 0.029	0.159 ± 0.019	0.000 ± 0.000	0.162 ± 0.029	0.051 ± 0.022	0.059 ± 0.021	
Macro-F1	Emotions	0.639 ± 0.029	0.641 ± 0.027	0.631 ± 0.022	0.551 ± 0.039	0.635 ± 0.037	0.038 ± 0.018	0.551 ± 0.039	0.608 ± 0.023	0.584 ± 0.013
	Flags	0.657 ± 0.063	0.671 ± 0.086	0.587 ± 0.065	0.658 ± 0.077	0.668 ± 0.077	0.560 ± 0.129	0.658 ± 0.077	0.687 ± 0.024	0.711 ± 0.025
	Scene	0.709 ± 0.009	0.728 ± 0.013	0.707 ± 0.003	0.634 ± 0.015	0.629 ± 0.002	0.000 ± 0.000	0.634 ± 0.015	0.700 ± 0.005	0.665 ± 0.010
	Yeast	0.385 ± 0.009	0.398 ± 0.006	0.374 ± 0.005	0.383 ± 0.010	0.405 ± 0.011	0.122 ± 0.003	0.384 ± 0.009	0.619 ± 0.006	0.593 ± 0.004
	Birds	0.321 ± 0.055	0.291 ± 0.012	0.265 ± 0.052	0.349 ± 0.048	0.336 ± 0.057	0.053 ± 0.033	0.349 ± 0.048	0.141 ± 0.022	0.145 ± 0.010
	GpositiveGO	0.871 ± 0.045	0.854 ± 0.062	0.901 ± 0.047	0.859 ± 0.054	0.845 ± 0.056	0.000 ± 0.000	0.859 ± 0.054	0.943 ± 0.008	0.949 ± 0.007
	CHD-49	0.498 ± 0.015	0.512 ± 0.026	0.510 ± 0.017	0.470 ± 0.022	0.490 ± 0.030	0.270 ± 0.002	0.470 ± 0.022	0.629 ± 0.007	0.635 ± 0.017
	Enron	0.219 ± 0.015	0.225 ± 0.016	0.182 ± 0.010	0.214 ± 0.021	0.157 ± 0.000	0.085 ± 0.014	0.214 ± 0.021	0.548 ± 0.009	0.553 ± 0.005
	Langlog	0.270 ± 0.047	0.273 ± 0.048	0.264 ± 0.043	0.284 ± 0.048	DNF	0.237 ± 0.047	0.051 ± 0.001	0.474 ± 0.006	0.478 ± 0.003
	Medical	0.653 ± 0.029	0.630 ± 0.031	0.616 ± 0.058	0.669 ± 0.037	0.468 ± 0.002	0.324 ± 0.036	0.669 ± 0.037	0.758 ± 0.015	0.755 ± 0.011
VirusGo	0.796 ± 0.078	0.833 ± 0.072	0.844 ± 0.090	0.803 ± 0.069	0.858 ± 0.089	0.067 ± 0.082	0.803 ± 0.069	0.902 ± 0.009	0.912 ± 0.011	
Water-qy	0.502 ± 0.005	0.523 ± 0.011	0.177 ± 0.019	0.413 ± 0.012	0.503 ± 0.004	0.091 ± 0.020	0.466 ± 0.011	0.518 ± 0.011	0.526 ± 0.010	
3s-bbc1000	0.062 ± 0.032	0.115 ± 0.027	0.246 ± 0.028	0.189 ± 0.051	0.180 ± 0.002	0.000 ± 0.000	0.189 ± 0.051	0.149 ± 0.021	0.152 ± 0.023	
Micro-F1	Emotions	0.662 ± 0.028	0.663 ± 0.025	0.654 ± 0.023	0.564 ± 0.038	0.654 ± 0.034	0.063 ± 0.032	0.564 ± 0.038	0.664 ± 0.013	0.658 ± 0.013
	Flags	0.746 ± 0.051	0.760 ± 0.051	0.725 ± 0.050	0.745 ± 0.046	0.741 ± 0.063	0.693 ± 0.064	0.745 ± 0.046	0.719 ± 0.017	0.737 ± 0.017
	Scene	0.705 ± 0.007	0.718 ± 0.012	0.700 ± 0.006	0.624 ± 0.015	0.617 ± 0.003	0.000 ± 0.000	0.624 ± 0.015	0.750 ± 0.004	0.733 ± 0.009
	Yeast	0.628 ± 0.011	0.636 ± 0.006	0.625 ± 0.012	0.581 ± 0.012	0.617 ± 0.006	0.480 ± 0.016	0.581 ± 0.011	0.644 ± 0.006	0.621 ± 0.004
	Birds	0.431 ± 0.054	0.450 ± 0.031	0.402 ± 0.034	0.444 ± 0.048	0.456 ± 0.055	0.000 ± 0.000	0.444 ± 0.048	0.365 ± 0.031	0.369 ± 0.027
	GpositiveGO	0.947 ± 0.008	0.939 ± 0.018	0.939 ± 0.009	0.946 ± 0.013	0.938 ± 0.018	0.000 ± 0.000	0.946 ± 0.013	0.942 ± 0.005	0.943 ± 0.009
	CHD-49	0.655 ± 0.017	0.667 ± 0.025	0.663 ± 0.018	0.619 ± 0.019	0.638 ± 0.028	0.598 ± 0.004	0.619 ± 0.019	0.658 ± 0.006	0.659 ± 0.017
	Enron	0.562 ± 0.004	0.583 ± 0.013	0.481 ± 0.016	0.550 ± 0.009	0.544 ± 0.002	0.245 ± 0.014	0.550 ± 0.009	0.565 ± 0.007	0.566 ± 0.004
	Langlog	0.159 ± 0.022	0.174 ± 0.012	0.156 ± 0.027	0.191 ± 0.014	DNF	0.000 ± 0.000	0.192 ± 0.029	0.532 ± 0.006	0.544 ± 0.003
	Medical	0.810 ± 0.016	0.815 ± 0.024	0.780 ± 0.028	0.813 ± 0.026	0.781 ± 0.027	0.000 ± 0.000	0.813 ± 0.026	0.754 ± 0.013	0.759 ± 0.007
VirusGo	0.890 ± 0.033	0.890 ± 0.036	0.881 ± 0.047	0.897 ± 0.042	0.898 ± 0.046	0.000 ± 0.000	0.897 ± 0.042	0.894 ± 0.008	0.894 ± 0.011	
Water-qy	0.563 ± 0.006	0.585 ± 0.011	0.304 ± 0.024	0.480 ± 0.010	0.570 ± 0.008	0.259 ± 0.045	0.544 ± 0.008	0.559 ± 0.007	0.557 ± 0.009	
3s-bbc1000	0.079 ± 0.023	0.173 ± 0.034	0.277 ± 0.033	0.215 ± 0.036	0.208 ± 0.030	0.000 ± 0.000	0.215 ± 0.036	0.086 ± 0.033	0.088 ± 0.042	

与组合方法 (EBR、ECC、EPS、RAkEL 和 CDE) 相比,MLWSE 在更多的情况下优于组合方法。这是因为 MLWSE 能够综合地获取不同标签的最优权值,并考虑到标签的局部成对相关性。

与叠加组合方案 (MLS) 相比,MLWSE 在大多数情况下显著提高了性能 (精度和 F1)。这是因为本文方法可以利用不同基础分类器对不同标签的置信度加权,并考虑局部成对标签相关性。

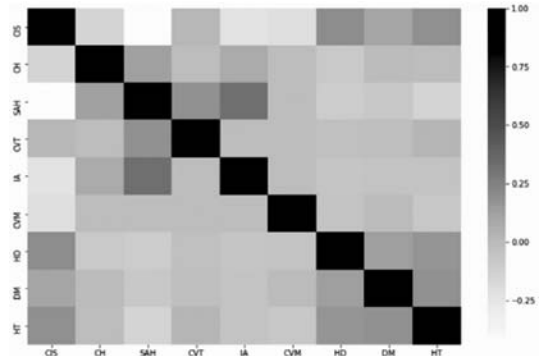
与加权叠加集成方法,即 AdaBoost. MH 相比,在大多数情况下,MLWSE 的表现优于 AdaBoost. MH,这是因为同时利用分类器权重和成对标签来解决多标签分类中的一些常见问题。

2.3.3 实际应用结果

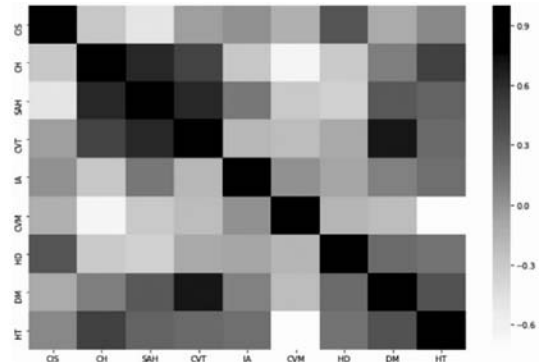
表 6 总结了不同多标签集成算法的实验结果。结果表明,与其他方法相比,本文方法在统计的性能较好。为了验证任意两个标签的相关程度,分类器权重向量具有很高的相似性。用于 CCD 的标签矩阵和学习的权重矩阵的亲矩阵见图 3,其中较强的灰色表示较强的标签相关性。图 3 表明,标签矩阵的亲矩阵与学习的权重矩阵的亲矩阵惊人地一致,这意味着,如果两个标签 y_j 和 y_k 强相关,则权重向量对 (W_j, W_k) 应该高度相似。实验结果验证了本文提出的假设的正确性。

表 6 每种算法基于 CCD 数据集的实验结果

算法	Accuracy	汉明损失	Ranking 损失	F1	Macro -F1	Micro -F1
EBR	0.692 3 ± 0.011 8	0.091 0 ± 0.005 0	0.039 5 ± 0.004 0	0.769 4 ± 0.010 2	0.403 8 ± 0.046 4	0.807 9 ± 0.010 0
ECC	0.704 1 ± 0.008 2	0.089 6 ± 0.004 1	0.047 2 ± 0.004 5	0.780 0 ± 0.006 4	0.419 6 ± 0.049 5	0.815 6 ± 0.007 4
EPS	0.690 4 ± 0.006 9	0.093 5 ± 0.003 4	0.049 2 ± 0.005 7	0.767 3 ± 0.006 0	0.404 5 ± 0.050 8	0.806 3 ± 0.006 9
RAkEL	0.679 7 ± 0.004 7	0.095 7 ± 0.002 8	0.085 3 ± 0.004 6	0.759 7 ± 0.004 0	0.398 5 ± 0.047 7	0.798 2 ± 0.005 8
CDE	0.695 3 ± 0.006 0	0.091 3 ± 0.003 4	0.057 9 ± 0.005 3	0.771 8 ± 0.004 9	0.409 6 ± 0.045 8	0.809 4 ± 0.006 6
AdaBoost. MH	0.617 8 ± 0.012 6	0.120 1 ± 0.004 5	0.053 6 ± 0.004 4	0.721 3 ± 0.011 0	0.214 6 ± 0.044 2	0.740 5 ± 0.009 4
MLS	0.679 7 ± 0.004 7	0.095 7 ± 0.002 8	0.081 4 ± 0.003 0	0.759 7 ± 0.004 0	0.398 5 ± 0.047 7	0.798 2 ± 0.005 8
MLWSE -L1	0.909 0 ± 0.001 5	0.091 0 ± 0.001 5	0.038 8 ± 0.001 6	0.797 9 ± 0.003 5	0.768 6 ± 0.002 7	0.810 2 ± 0.003 8
MLWSE -L21	0.910 1 ± 0.000 9	0.089 9 ± 0.000 9	0.038 4 ± 0.007 8	0.796 8 ± 0.003 5	0.768 1 ± 0.003 3	0.811 6 ± 0.002 3



(a) 标签矩阵的亲矩阵



(b) 权重矩阵的亲矩阵

图 3 亲矩阵

2.4 算法分析

2.4.1 Friedman 统计分析

本文采用 Friedman 检验对不同算法的性能进行了系统的统计分析。表 7 总结了弗里德曼统计 F_F 和每个评价指标的相应临界值,在显著性水平 $\alpha = 0.05$ 时,对于每个评估指标,相同绩效的零假设被拒绝。使用 Nemenyi 测试来分析本文方法 MLWSE-L1 和 MLWSE-L21 相对于其他算法的性能。如果对应的平均秩差至少达到临界差 $C_D = q_\alpha \sqrt{\frac{k(k+1)}{6N}}$,则两个分类器的性能将显著不同。在显著性水平 $\alpha = 0.05$, $q_\alpha = 3.102$,对于 $k = 9; N = 14$ (包括 13 个基准数据集和一个真实数据集)的 Nemenyi 检验, $C_D = 3.211$ 。弗里德曼统计结果表明:

EBR 在 Hamming 损失上优于其他方法。这是因为 EBR 是一种优化 Hamming 损失的一阶方法,不考虑标签相关性。但在 Hamming 损失方面,MLWSE-L21 的性能优于 MLWSE-L1 和其他方法。

ECC 在 Micro-F1 上的性能优于其他方法,因为 ECC 是一种高阶方法,考虑全局标签相关性,它试图对全局标签相关性进行建模。但是,本文提出的 MLWSE-L1 在 Micro-F1 方面优于 MLWSE-L21 和其他方法。

MLWSE 在其他四个评价指标上都优于其他方法,MLWSE 相对于这些方法的优越性能表明了学习标签

相关性和分类器权重的有效性。

表7 弗里德曼统计以及每个评估指标的临界值

指标	F_F	临界值($\alpha=0.05$)
精度	35.075	3.211
汉明损失	6.348	—
Ranking 损失	37.824	—
F1	9.261	—
Macro-F1	10.243	—
Micro-F1	8.312	—

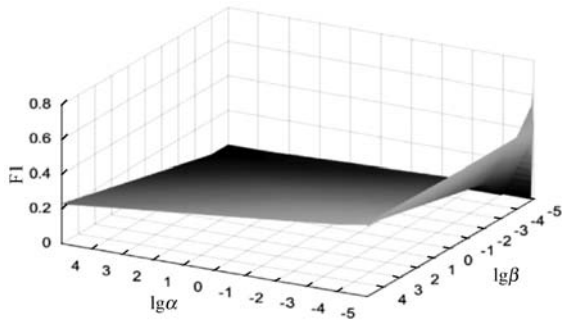
2.4.2 参数敏感性分析

通过对 Emotions 和 GpositiveGO 数据集的实验,分析了 MLWSE-L1 和 MLWSE-L21 的参数敏感性。本研究对每个数据集进行了五次交叉验证,并分析了相关参数。

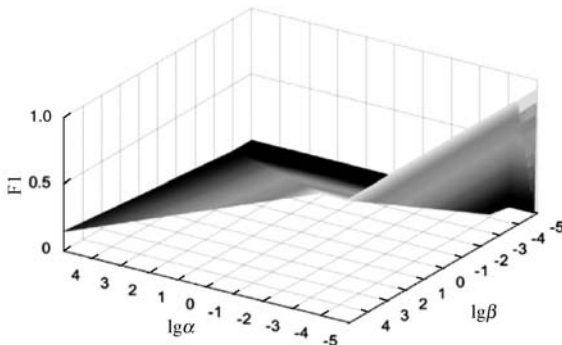
在 MLWSE-L1 中, α 和 β 的取值范围为 $\{10^{-5}, 10^{-4}, \dots, 10^3, 10^4\}$; η 设为 0.1。对于每个 $\{\alpha, \beta\}$ 对,计算 F1 的平均值。图 4(a)和图 4(b)分别描绘了参数 α 和 β 对 Emotions 和 GpositiveGO 数据集的影响。可以看出,在大多数情况下:

当 α 值较大时,MLWSE-L1 的性能较差,特别是当 $\alpha > 10$ 时效果最差。

MLWSE-L1 的性能随着 β 的增加先提高后降低。因此,将多标签基准数据集的正则化参数 α 和 β 分别设为 10^{-4} 和 10^{-3} 。



(a)

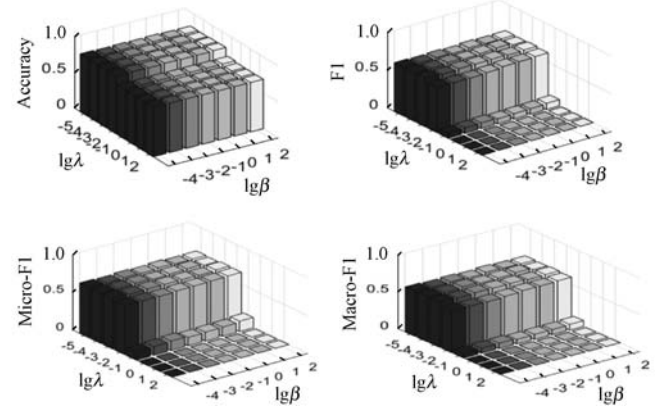


(b)

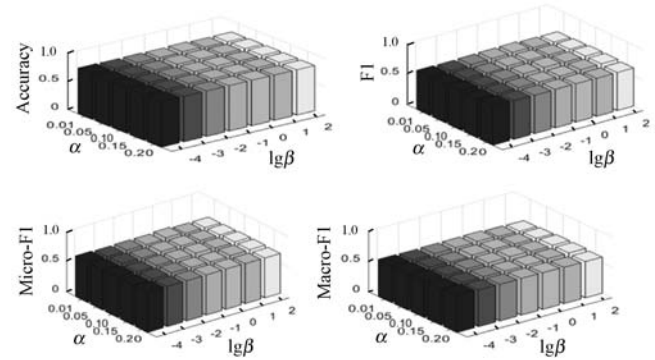
图4 参数的影响

在 MLWSE-L21 中,正则化参数 α 的取值范围为

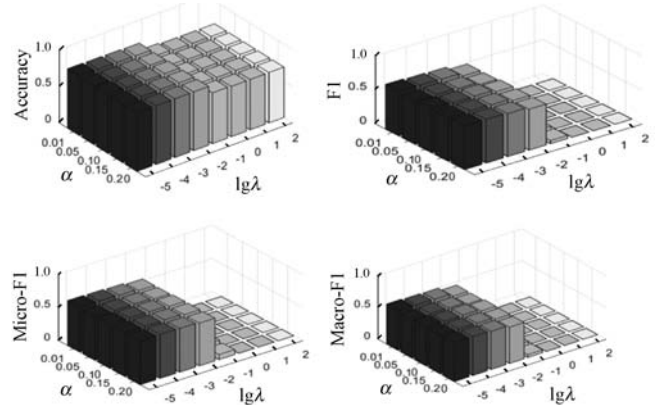
$\{0.01, 0.05, 0.1, 0.15, 0.2\}$; 参数 λ 的取值范围为 $\{10^{-5}, 10^{-4}, \dots, 10^1, 10^2\}$; β 的取值范围为 $\{10^{-4}, 10^{-3}, \dots, 10^1, 10^2\}$; η 设为 0.1。首先对 Emotions 数据集的训练数据进行五次交叉验证,找出参数的最佳配置组,然后保持一个参数的值不变,改变另外两个参数的值。图 5(a) - 图 5(c)描述了具有不同值 α, β 和 λ 的 MLWSE-L21 的平均结果。



(a)



(b)



(c)

图5 不同参数值的结果

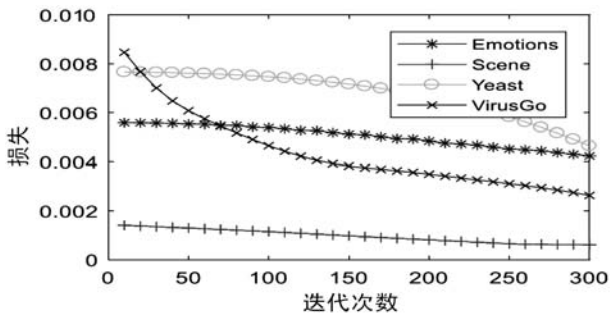
给定 α 情况下, λ 和 β 的候选集为 $\{10^{-4}, 10^{-3}, 10^{-2}\}$,可以得到较满意的结果。

给定 λ 情况下,当每对值 (α, β) 不同时,MLWSE-L21 的性能也是稳定的。

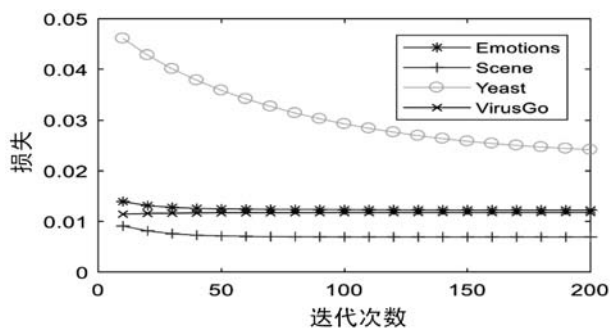
给定 β 情况下, α 和 λ 的候选集为 $\{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}\}$,可以得到较满意的结果。

2.4.3 收敛性分析

通过基于 Emotions、Scene、Yeast 和 VirusGO 数据集的实验,分析了 MLWSE-L1 和 MLWSE-L21 的收敛性。在这项工作中,本文方法是通过使用加速的近端梯度和块坐标下降来解决的,这可以看作是迭代收缩阈值算法。加速近端梯度收敛于函数值为 $O(1/t^2)$, 块坐标下降在函数值上收敛为 $O((\log t/t)^2)$ 。MLWSE-L1 和 MLWSE-L21 的损失函数的值分别与迭代周期的数目见图 6。对于 MLWSE-L1, 经过 300 个迭代周期后,损失值趋于稳定。图 6(a) 表明,当迭代循环次数超过 200 次时,损耗值对性能影响不大,因此,将实验中的循环次数设为 200 次。对于 MLWSE-L21, 损失值在 200 个循环后趋于稳定。这里,迭代循环的次数指的是外循环,而实验中内环的循环次数设置为 100 次。实验结果表明,该方法比大多数多标签集成方法的性能更好。



(a) MLWSE-L1



(b) MLWSE-L21

图6 对比结果

3 结语

为了充分挖掘成对标签的相关性以及分类器权重与分类器选择之间的关系,提出一种基于稀疏正则化的加权叠加集成多标签分类方法。通过多个数据集上的实验结果可以得出如下结论:

(1) 本文方法由于其快速收敛的学习过程,能够快速有效地完成多标签分类任务,并且具有较高的分类精度。

(2) 本文方法不仅提供了一种标签元特定的特征

选择方法,而且还兼容任何现有的多标签分类算法作为其基础分类器,具有良好的分类鲁棒性。

(3) 本文方法同时采用基于稀疏性的正则化方法进行分类器选择和集成构造,并采用余弦相似性方法计算标签相关矩阵,有效提高了计算效率。

参考文献

- [1] 贺成诚,汪海涛,姜瑛,等. 基于学习排序的多分类标签排序方法研究[J]. 计算机应用与软件,2019,36(2):7-11,119.
- [2] 霍东雪,刘辉,尚振宏,等. 一种异构集成学习的儿科疾病诊断方法研究[J]. 计算机应用与软件,2018,35(6):54-57,157.
- [3] 杨菊英,刘焱,罗佳. 基于划分子集主题模型的多标签极限分类[J]. 计算机工程与设计,2020,41(12):3432-3437.
- [4] 牟甲鹏,蔡剑,余孟池,等. 基于标签相关性的类属属性多标签分类算法[J]. 计算机应用研究,2020,37(9):2656-2658,2673.
- [5] 宋攀,景丽萍. 基于神经网络探究标签依赖关系的多标签分类[J]. 计算机研究与发展,2018,55(8):1751-1759.
- [6] Yang Y, Jiang J M. Adaptive Bi-weighting toward automatic initialization and model selection for HMM-based hybrid meta-clustering ensembles[J]. IEEE Transactions on Cybernetics,2019,49(5):1657-1668.
- [7] Moyano J, Gibaja E, Cios K, et al. Review of ensembles of multi-label classifiers: Models, experimental study and prospects[J]. Information Fusion,2018,44(6):33-45.
- [8] Kumar V, Pujari A, Padmanabhan V, et al. Group preserving label embedding for multi-label classification[J]. Pattern Recognition,2019,90(8):23-34.
- [9] 耿晗. 基于随机森林的苹果内部品质多标签分类方法研究[D]. 咸阳:西北农林科技大学,2018.
- [10] 温柳英,袁伟. 多标签符号型属性值划分的聚类方法[J]. 山东大学学报(理学版),2020,55(3):58-69.
- [11] 柴瑞敏,闫婷. 基于模糊 C 均值改进的粒化特征加权多标签分类算法[J]. 计算机应用与软件,2018,35(6):286-291.
- [12] Deng M Q, Wang C, Tang M, et al. Extracting cardiac dynamics within ECG signal for human identification and cardiovascular diseases classification[J]. Neural Networks,2018,100(6):70-83.
- [13] Szymáński P, Kajdanowicz T. Scikit-multilearn: A scikit-based Python environment for performing multi-label classification[J]. Journal of Machine Learning Research,2019,20(8):209-230.
- [14] 高翔,李春庚,安居白. 基于注意力和多标签分类的图像实时语义分割[J]. 计算机辅助设计与图形学学报,2021,33(1):59-67.
- [15] 杨岚雁,靳敏,张迎春,等. 一种基于关联规则的 MLKNN 多标签分类算法[J]. 计算机工程与科学,2020,42(7):1309-1317.