Re-weighting Large Margin Label Distribution Learning for Classification

Jing Wang, Xin Geng*, *Senior Member, IEEE,* and Hui Xue

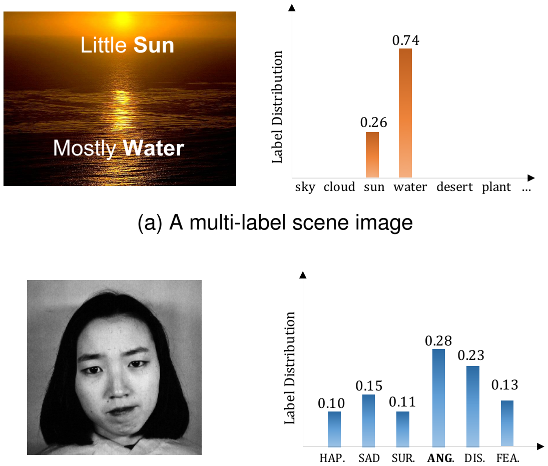
Abstract—Label ambiguity has attracted quite some attention among the machine learning community. The latterly proposed Label Distribution Learning (LDL) can handle label ambiguity and has found wide applications in real classification problems. In the training phase, an LDL model is learned first. In the test phase, the top label(s) in the label distribution predicted by the learned LDL model is (are) then regarded as the predicted label(s). That is, LDL considers the whole label distribution in the training phase, but only the top label(s) in the test phase, which likely leads to objective inconsistency. To avoid such inconsistency, we propose a new LDL method Re-Weighting Large Margin Label Distribution Learning (RWLM-LDL). First, we prove that the expected L_1-norm loss of LDL bounds the classification error probability, and thus apply L_1-norm loss as the learning metric. Second, re-weighting schemes are put forward to alleviate the inconsistency. Third, large margin is introduced to further solve the inconsistency. The theoretical results are presented to showcase the generalization and discrimination of RWLM-LDL. Finally, experimental results show the statistically superior performance of RWLM-LDL against other comparing methods.

Index Terms—Label Distribution Learning (LDL), Classification, Re-weighting, Large Margin, Generalization

1 Introduction

Label ambiguity [1] is the phenomenon that one instance is related to multiple labels by different degrees, which is a hot topic in the field of machine learning. Take multi-label image classification and emotion recognition as examples. Fig. 1a shows a multi-label scene image from [2]. Note that "Water" has higher importance than "Sun", although both are positive labels. Fig. 1b shows an image from the JAFFE database [3] with a ground-truth single-label "ANG.". However, the image is a mixture of many kinds of emotions by different relevance. Traditional supervised learning paradigms, such as Single-Label Learning (SLL) and Multi-Label Learning (MLL) [4], model the correspondence between instances and labels by 0 or 1, which fails to consider label ambiguity.

Recently, a novel learning paradigm called Label Distribution Learning (LDL) [5] is proposed as a possible solution to label ambiguity. Unlike SLL and MLL, LDL models the correspondence between instances and labels by real values. Specifically, LDL assigns each instance with a label distribution, and the elements of a label distribution are called the label description degrees that explicitly indicate the relative importance of labels. In LDL, for Fig. 1a, "Water" and "Sun" are respectively given the label description degrees of 0.74 and 0.26 (the label distribution is got from [6]), which tells the difference of label-importance. Similarly, in LDL, Fig. 1b is annotated with a label distribution $[0.10, 0.15, 0.11, 0.28, 0.23, 0.13]^{\top}$ (the label distribution is from the mean ratings by 60 annotators [3]), which models the relevance of emotions. Label distribution directly tells how much does each label describe the instance [5], which is more general than 0/1 label. The goal of LDL is to learn a mapping from instance to label distribution directly.



(a) A multi-label scene image

(b) A facial emotion image

Fig. 1. Illustration of label ambiguity and label distribution. Fig. 1a shows a multi-label scene image with little "Sun" and mostly "Water". Fig. 1b shows an image from the JAFFE database [3] with a ground-truth single-label "ANG.". The image is a mixture of emotions by different relevance. The label distribution of Fig. 1a is got from [6], and the label distribution of Fig. 1b is from the mean ratings from 60 annotators [5].

LDL has already been applied to varieties of real classification applications, such as age estimation [7], [8], head-pose estimation [9], expression recognition [10], beauty perception [11], acne image grading and counting [12], multi-label classification [13], *etc*. There are two phases involved in the applications. First, in the training phase, an LDL model is learned by minimizing the distance between the model's output and the specific label distribution (*e.g.*, age distribution [7]). Second, in the test phase, the top label(s) in

- *J. Wang, X. Geng and H. Xue are with the School of Computer Science and Engineering, and Key Lab of Computer Network and Information Integration, Southeast University, Nanjing 211189, China.*
E-mail: {wangjing91, xgeng, hxue}@seu.edu.cn

* *Corresponding author*
Manuscript received April 19, 2005; revised August 26, 2015.

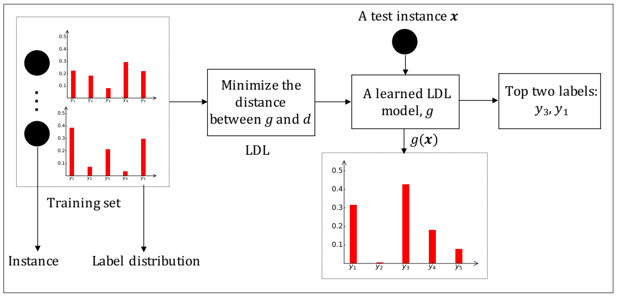

A test instance $\boldsymbol{x}$
Minimize the distance between g and d
LDL
A learned LDL model, g
Top two labels: y_3, y_1
$g(\boldsymbol{x})$
Training set
Instance
Label distribution

Fig. 2. Illustration of multi-label classification in the framework of LDL, where d is the given label distribution function, and g is an LDL model.

the label distribution predicted by the learned LDL model is (are) then regarded as the predicted label(s). Take multi-label classification as an instance, which is depicted in Fig. 2. First, an LDL model is learned from the training set described by label-importance-aware label distribution [13]. Next, an unknown instance is directly fed into the learned LDL model. The top two labels having the largest predicted label description degrees (*i.e.*, y_1 and y_3) are then treated as the predicted labels.

Although LDL has found wide applications, it faces the challenge of objective inconsistency between the training phase and the test phase. The goal of the training phase is to learn the whole label distribution, while the goal of the test phase is to learn the top label(s) – LDL may neglect the top label(s) for the sake of learning the whole label distribution. The objective inconsistency may result in a sub-optimal decision function [14]. To see that, for the example of Fig. 1b, suppose that the learned label distribution is $[0.10, 0.15, 0.11, 0.25, 0.26, 0.13]^\top$, which is a good approximation to the ground-truth label distribution with an L_1-norm loss of 0.05. However, for SLL, the predicted label in the predicted label distribution is "DIS.", which is different from the ground-truth label "ANG.".

This paper tackles the objective inconsistency. We design a new LDL method called **R**e-weighting **L**arge **M**argin **L**abel **D**istribution **L**earning (**RWLM-LDL**). *RWLM-LDL* incorporates three components, including L_1-norm loss, re-weighting schemes, and large margin. First, we prove that the expected L_1-norm loss of LDL bounds the classification error probability. According to that finding, we apply L_1-norm loss as the learning metric to minimize the classification error probability. Second, we alleviate the inconsistency by re-weighting instances *w.r.t.* the information entropy of label distributions and re-weighting labels *w.r.t.* label description degrees. Third, to further solve the inconsistency, we introduce large margin to LDL. To thoroughly validate *RWLM-LD*L, we conduct theoretical analysis and empirical evaluations. Theoretical results reveal the generalization and discrimination of *RWLM-LDL*. Experimental results show the statistically superior performance of *RWLM-LDL* against the baseline methods.

Preliminary results have already been presented in [15], which only considered SLL. The main contribution of this paper is that we establish a unified view of SLL and MLL in the framework of LDL. Specifically, a more general theorem (Theorem 2) is presented to uncover the relation between LDL and classification, and more explanations are added concerning the re-weighting schemes and large margin. Additionally, more experimental results are reported.

The rest of the paper is organized as follows. First, section 2 briefly reviews some related works. Then, section 3 presents the details of *RWLM-LDL*. Next, section 4 conducts the theoretical analysis. Besides, section 5 reports the experimental results. Finally, section 6 concludes.

2 Related Work

This paper is related to label distribution learning, large margin, and re-weighting, which are discussed as follows.

Geng *et al.* [7] first introduced label distribution to alleviate the insufficiency of training examples in age estimation and proposed *IIS-LLD* and *CPNN* to learn from such label distribution. Latter, Geng [5] formalized LDL as a new learning paradigm and put forward several baseline methods, including *PT-SVM*, *PT-Bayes*, *AA-kNN*, *AA-BP*, and *SA-BFGS*. Since then, many LDL algorithms have been designed. Shen *et al.* [16] used the differentiable decision trees to learn label distribution and proposed *LDLFs*. *LDLFs* can learn any form of label distribution and can be combined with representation learning [16]. Chen *et al.* [17] employed the structured random forest to exploit the structural information among different classes and proposed *StructRF*. Yang *et al.* [18] applied the regularized sample self-representation technique to LDL and proposed *RSSR-LDL21*. Jia *et al.* [19] exploited local label correlation and put forward *LDL-SCL*. Gao *et al.* [1] designed the first deep LDL model *DLDL*. Nevertheless, the objective inconsistency of LDL has not been considered in the design of the above proposals.

LDL has found wide applications in many classification tasks. In MLL, Zhang *et al.* [13] applied label distribution to model the relative importance of labels. Rather than learning 0/1 label, they directly learned the label-importance-aware label distribution [13]. For a test instance, the top labels in the predicted label distribution are regarded as the positive labels. In facial beauty perception, Liang *et al.* [11] adopted label distribution to describe frontal faces, which can keep all the rating information from raters. In acne image analysis, Wu *et al.* [12] used two label distributions to model the uncertainty of the number of lesions and the acne severity for a face image. Moreover, in head-pose estimation, Geng *et al.* [9] employed multivariate label distribution to alleviate the problem of inaccurate pose labels and directly learned a mapping from instances to the multivariate label distributions. Similarly, the pose label with the highest predicted label description degree is regarded as the predicted pose. However, none of the above works realize the objective inconsistency of LDL.

There are a few works on the objective inconsistency of LDL. Gao *et al.* [14] first recognized the objective inconsistency in the application of age estimation. They designed a lightweight network to jointly learn the label distribution and the ground-truth age label. However, the method is only suitable for real-valued label space, and theoretical guarantees are not provided. Besides, we analyzed the learnability of classification in the framework of LDL in a recent work [20]. However, the theory only applies to SLL, and the objective inconsistency is not considered. In contrast, *RWLM-LDL* is a general LDL approach that addresses the objective inconsistency and has theory guarantees.

The large margin (or maximum margin) was first introduced by Vapnik and Chervonenkis [21] and directly led to the Support Vector Machine (SVM) [22]. SVM attempts to maximize the margin of training instances. Large margin has been widely used in many SLL and MLL methods, such as multi-class SVM [23], Optimal Margin Distribution Machine (OMD) [24], Binary Relevance SVM (BR-SVM) [4], Rank-SVM [25], LIMO [26], *etc*. Moreover, margin theory is an important statistical tool that has been adopted to analyze the generalization of many algorithms [27]. Re-weighting assigns different weights to different samples, which has been well studied in the literature, such as importance sampling to match up different distributions [28], dataset sampling [29] to deal with imbalanced datasets, boosting methods [30], [31] to weight samples based on the training loss, *etc*. We first introduce large margin and re-weighting to LDL and solve the objective inconsistency.

3 The RWLM-LDL Approach

We start with the preliminaries and then establish the relation between LDL and classification, which is the theoretical foundation of *RWLM-LDL*. Next, we elaborate on the algorithm formulation.

3.1 Preliminaries

Denote by $\mathcal{X} \subseteq \mathbb{R}^q$ the input space and $\mathcal{Y} = \{y_1, \cdots, y_m\}$ the label space. In LDL, each instance $\boldsymbol{x} \in \mathcal{X}$ is associated with a label distribution $\boldsymbol{d_x} = [d_{\boldsymbol{x}}^{y_1}, \cdots, d_{\boldsymbol{x}}^{y_m}]^\top$, where $d_{\boldsymbol{x}}^{y_j}$ is called the label description degree and indicates the relative importance of y_j to $\boldsymbol{x}$. Furthermore, label description degree satisfies $\sum_j d_{\boldsymbol{x}}^{y_j} = 1$ and $d_{\boldsymbol{x}}^{y_j} \geq 0$. The summary of the mainly used notations is listed in Table 1. Given a training set $S = \{(\boldsymbol{x}_1, \boldsymbol{d}_{\boldsymbol{x}_1}), \ldots, (\boldsymbol{x}_n, \boldsymbol{d}_{\boldsymbol{x}_n})\}$ and a loss function $\ell : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}^+$, LDL can be cast as the following [5]

$$\min_{\boldsymbol{W}} \sum_{i=1}^{n} \ell(g(\boldsymbol{x}_i; \boldsymbol{W}), \boldsymbol{d}_{\boldsymbol{x}_i}),$$

where g is a parametric model, and $\boldsymbol{W}$ is the parameter.

Let $\mathcal{D}_{\mathcal{X}}$ be the underlying distribution over $\mathcal{X}$. Let $y \in \mathcal{Y}$ denote the (random) SLL label variable, and $\boldsymbol{y} \in \{0,1\}^m$ denote the (random) MLL label variable. For an SLL classifier $f : \mathcal{X} \to \mathcal{Y}$, the error probability [32] is defined by

$$L(f) = \mathbb{E}_{\boldsymbol{x},y}\left[\mathbb{I}(f(\boldsymbol{x}) \neq y)\right],$$

where $\mathbb{I}(\cdot)$ is the indicator function. For an MLL classifier $f : \mathcal{X} \to \{0,1\}^m$, the error probability is defined by

$$L(f) = \mathbb{E}_{\boldsymbol{x},\boldsymbol{y}}\left[\frac{1}{m}\sum_{j=1}^{m} \mathbb{I}(f_j(\boldsymbol{x}) \neq \boldsymbol{y}_j)\right],$$

where $\boldsymbol{y}_j$ is the jth element of $\boldsymbol{y}$, and $f_j(\boldsymbol{x})$ is the prediction for the jth label. Our objective is to minimize $L(f)$.

Let $\eta : \mathcal{X} \mapsto \mathbb{R}^m$ be the conditional probability distribution function, and $\eta_j(\cdot)$ be the jth output of $\eta(\cdot)$. For SLL, $\eta_j(\boldsymbol{x}) = \mathbb{P}(y = y_j \mid \boldsymbol{x})$. For MLL, we have

$$\eta_j(\boldsymbol{x}) = \mathbb{P}(\boldsymbol{y}_j = 1 \mid \boldsymbol{x}) = \sum_{\boldsymbol{y}:\boldsymbol{y}_j=1} \mathbb{P}(\boldsymbol{y} \mid \boldsymbol{x}).$$

For simplicity, let $\eta_{\boldsymbol{x}}^{y_j} = \eta_j(\boldsymbol{x})$ and $\boldsymbol{\eta_x} = [\eta_{\boldsymbol{x}}^{y_1}, \ldots, \eta_{\boldsymbol{x}}^{y_m}]^\top$. Let $d : \mathcal{X} \to \mathbb{R}^m$ be the ground-truth label distribution function.

TABLE 1
Summary of the mainly used notations.

Symbol	Definition
$\mathcal{X}$	Feature Space
$\mathcal{Y}$	Label Space
$\boldsymbol{x}_i$	The ith training instance
$\boldsymbol{d}_{\boldsymbol{x}_i}$	The ith training label distribution
$d_{\boldsymbol{x}_i}^{y_j}$	The label description degree of y_j to $\boldsymbol{x}_i$
$y, \boldsymbol{y}$	SLL and MLL (random) label variables
q	The number of feature dimensions
n	The number of training instances
m	The number of classes
$\mathcal{D}_{\mathcal{X}}$	The underlying distribution over $\mathcal{X}$
η	Conditional probability distribution function
$\mathbb{I}(\cdot)$	Indicator function
g^*	The optimal classifier
$L(\cdot)$	Error probability function

3.2 Relation between LDL and Classification

Assume that d ranks the labels the same as η does[1]. Then, for SLL, the optimal (Bayes) classifier [32] can be defined by

$$g^*(\boldsymbol{x}) = \arg\max_{y \in \mathcal{Y}} d_{\boldsymbol{x}}^{y},$$

which outputs the label having the largest label description degree. For MLL, the optimal classifier [33] is defined by

$$g^*(\boldsymbol{x}) = \{g_1^*(\boldsymbol{x}), g_2^*(\boldsymbol{x}), \ldots, g_m^*(\boldsymbol{x})\},$$

and $g_j^*(\boldsymbol{x})$ equals 1 if $j \in \operatorname{rank}_{k(\boldsymbol{x})}(\boldsymbol{d_x})$ and 0 otherwise, where $\operatorname{rank}_{k(\boldsymbol{x})}(\cdot)$ returns the indices of the $k(\boldsymbol{x})$ largest values ranked in descending order, and $k(\boldsymbol{x}) = |\{j : \eta_{\boldsymbol{x}}^{y_j} \geq 0.5\}|$ [33]. That is, the top $k(\boldsymbol{x})$ labels having the largest label description degrees are returned.

Let $g : \mathcal{X} \to \mathbb{R}^m$ denote a learned LDL function. For simplicity, let $g_{\boldsymbol{x}}^{y_j} = g_j(\boldsymbol{x})$ and $\boldsymbol{g_x} = [g_{\boldsymbol{x}}^{y_1}, g_{\boldsymbol{x}}^{y_2}, \ldots, g_{\boldsymbol{x}}^{y_m}]^\top$. Similarly, an SLL classifier can be induced by

$$\hat{g}(\boldsymbol{x}) = \arg\max_{y \in \mathcal{Y}} g_{\boldsymbol{x}}^{y}, \tag{1}$$

and an MLL classifier can be induced by

$$\hat{g}(\boldsymbol{x}) = \{\hat{g}_1(\boldsymbol{x}), \hat{g}_2(\boldsymbol{x}), \ldots, \hat{g}_m(\boldsymbol{x})\}, \tag{2}$$

where $\hat{g}_j(\boldsymbol{x})$ equals 1 if $j \in \operatorname{rank}_{k(\boldsymbol{x})}(\boldsymbol{g_x})$ and 0 otherwise.

Theorem 1. *Let g be a learned LDL function. Then, the error probability of an SLL classifier as Eq. (1) satisfies*

$$L(\hat{g}) - L(g^*) \leq \mathbb{E}_{\boldsymbol{x} \sim \mathcal{D}_{\mathcal{X}}}\left[\sum_{j=1}^{m} |g_{\boldsymbol{x}}^{y_j} - \eta_{\boldsymbol{x}}^{y_j}|\right],$$

and the error probability of an MLL classifier as Eq. (2) satisfies

$$L(\hat{g}) - L(g^*) \leq \frac{2}{m}\mathbb{E}_{\boldsymbol{x} \sim \mathcal{D}_{\mathcal{X}}}\left[\sum_{j=1}^{m} |g_{\boldsymbol{x}}^{y_j} - \eta_{\boldsymbol{x}}^{y_j}|\right].$$

[1] In statistical learning theory, η decides which label is the Bayes prediction [32]. In LDL, d determines the relative importance of labels. This assumption connects two of them.

Theorem 1 extends the plug-in decision theorem [32]. The right-hand sides of the bounds are (for MLL, $^2/_m\times$) the expected L_1-norm distance between g and η. The theorem says that the error probability of the induced (SLL and MLL) classifiers would approach that of the optimal classifiers as long as g is close to η in the L_1-norm sense. The proof is in the Supplementary Material. Notice that the scale of g may mismatch with that of η. To see that, g generally satisfies $\sum_j g_{\boldsymbol{x}}^{y_j} = 1$ while η may not satisfy the constraint for MLL. To avoid that, define $\hat{\eta}$ by $\hat{\eta}_{\boldsymbol{x}}^{y_j} = \eta_{\boldsymbol{x}}^{y_j}/N_{\boldsymbol{x}}$, where $N_{\boldsymbol{x}} = \sum_j \eta_{\boldsymbol{x}}^{y_j}$. Next, the following theorem discloses the relation between LDL and classification.

Theorem 2 (Relation between LDL and Classification)**.** *Let g be a learned LDL function, and $\hat{g}$ be the induced (SLL/MLL) classifier. Then, the error probability of $\hat{g}$ satisfies*

$$\begin{aligned} L(\hat{g}) - L(g^*) \leq\, & 2\mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}_{\mathcal{X}}}\left[\sum_{j=1}^{m} |g_{\boldsymbol{x}}^{y_j} - d_{\boldsymbol{x}}^{y_j}|\right] \\ & + 2\mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}_{\mathcal{X}}}\left[\sum_{j=1}^{m} |d_{\boldsymbol{x}}^{y_j} - \hat{\eta}_{\boldsymbol{x}}^{y_j}|\right]. \end{aligned} \tag{3}$$

Theorem 2 holds for both SLL and MLL cases. The right-hand side of Eq. (3) is the sum of two terms, where the first one is the expected L_1-norm loss of g, and the second one is the expected L_1-norm distance between d and $\hat{\eta}$. The second term is a constant independent of g, which can be ignored in LDL. As a result, the theorem states that the expected L_1-norm loss of LDL bounds the classification error probability. Theorem 2 is the theoretical foundation of *RWLM-LDL*. The proof is deferred to the Supplementary Material.

3.3 Algorithm Formulation

3.3.1 LDL with L_1-norm Loss

According to Theorem 2, to minimize the error probability, it suffices to minimize the sum of the terms on the right-hand side of Eq. (3). Because the second term is a constant, we only need to minimize the first one, *i.e.*, the expected L_1-norm loss of LDL, which inspires us to apply L_1-norm loss as the learning metric for LDL. Analogous to [5], we adopt the maximum entropy model to learn the label distribution [34], which is parameterized by

$$g_{\boldsymbol{x}}^{y_j} = \frac{1}{Z_{\boldsymbol{x}}}\exp(\boldsymbol{w}_j \cdot \boldsymbol{x}),$$

where $\boldsymbol{w}_j \in \mathbb{R}^q$ is the parameter, and $Z_{\boldsymbol{x}} = \sum_j \exp(\boldsymbol{w}_j \cdot \boldsymbol{x})$ is the normalization factor. The output of the maximum entropy model satisfies the probability simplex constraint (*i.e.*, $g_{\boldsymbol{x}}^{y_j} \geq 0$ and $\sum_{j=1}^m g_{\boldsymbol{x}}^{y_j} = 1$). Next, applying L_1-norm loss as the learning metric for LDL, the problem of LDL can be formulated as the following,

$$\min_{\boldsymbol{W}} \sum_{i=1}^{n}\sum_{j=1}^{m} |g_{\boldsymbol{x}_i}^{y_j} - d_{\boldsymbol{x}_i}^{y_j}| + \frac{\lambda_1}{2}\|\boldsymbol{W}\|_{\mathrm{F}}^2 \tag{4}$$

where $\boldsymbol{W} = [\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_m]$, $\|\cdot\|_{\mathrm{F}}^2$ is the Frobenius norm, and λ_1 is the regularization parameter.

There are two advantages of L_1-norm loss. The first one is that L_1-norm loss directly relates LDL with classification. As a result, we can minimize the error probability by minimizing the expected L_1-norm loss. The second one is that both SLL and MLL can be formulated into the framework of LDL using L_1-norm loss according to Theorem 2.

3.3.2 Re-weighting LDL

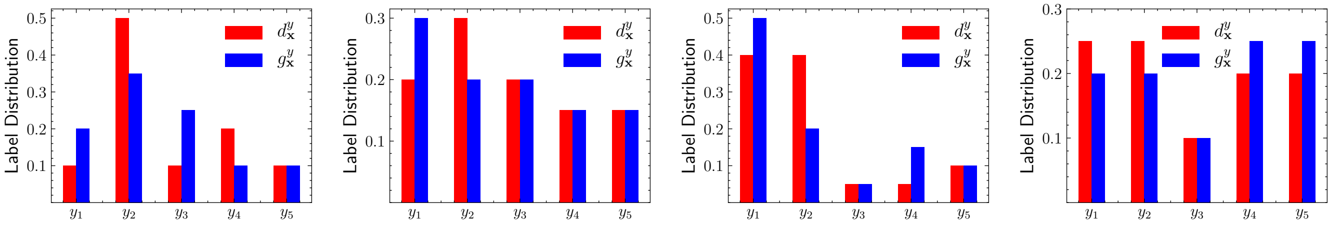
We proceed by showing in Fig. 3 that the characteristics of label distribution may result in the inconsistency. Specifically, Fig. 3a and Fig. 3b show the case of SLL, where $\boldsymbol{d_x} = [0.1, 0.5, 0.1, 0.2, 0.1]^\top$ and $\boldsymbol{g_x} = [0.2, 0.35, 0.25, 0.1, 0.1]^\top$ for Fig. 3a, and $\boldsymbol{d_x} = [0.2, 0.3, 0.2, 0.15, 0.15]^\top$ and $\boldsymbol{g_x} = [0.3, 0.2, 0.2, 0.15, 0.15]^\top$ for Fig. 3b. Besides, Fig. 3c and Fig. 3d explain the case of MLL, where $\boldsymbol{d_x} = [0.4, 0.4, 0.05, 0.05, 0.1]^\top$ and $\boldsymbol{g_x} = [0.5, 0.2, 0.05, 0.15, 0.1]^\top$ for Fig. 3c, and $\boldsymbol{d_x} = [0.25, 0.25, 0.1, 0.2, 0.2]^\top$ and $\boldsymbol{g_x} = [0.2, 0.2, 0.1, 0.25, 0.25]^\top$ for Fig. 3d. Then, we can make the following three observations:

1) For the case of Fig. 3a, the L_1-norm loss is 0.5 and $\hat{g}(\boldsymbol{x}) = g^*(\boldsymbol{x}) = y_2$. In contrast, for the case of Fig. 3b, the L_1-norm loss is 0.2, but $\hat{g}(\boldsymbol{x}) \neq g^*(\boldsymbol{x})$ ($\hat{g}(\boldsymbol{x}) = y_1$ and $g^*(\boldsymbol{x}) = y_2$).
2) For the case of Fig. 3c, the L_1-norm loss is 0.4 and $\hat{g}(\boldsymbol{x}) = g^*(\boldsymbol{x}) = \{1, 1, 0, 0, 0\}$ (top 2 labels are considered). In contrast, for the case of Fig. 3d, the L_1-norm loss is 0.2, while $\hat{g}(\boldsymbol{x}) \neq g^*(\boldsymbol{x})$ ($\hat{g}(\boldsymbol{x}) = \{1, 1, 0, 0, 0\}$ and $g^*(\boldsymbol{x}) = \{0, 0, 0, 1, 1\}$).
3) The cases of Fig. 3b and Fig. 3d have smaller L_1-norm losses, but the cases of Fig. 3a and Fig. 3c have smaller $0/1$ losses. On one hand, the cases of Fig. 3a and Fig. 3c are inferior to Fig. 3b and Fig. 3d from the view of LDL. On the other hand, the cases of Fig. 3a and Fig. 3c are superior to Fig. 3b and Fig. 3d from the perspective of classification, which validates the objective inconsistency.

In light of the above observations, for overall unevenly-distributed label distributions (like the cases of Fig. 3a and Fig. 3c), the label description degrees are mainly dominated by the top label(s). As a result, the inconsistency is less likely to occur since there is more room to maneuver. In contrast, for overall evenly-distributed label distributions (like the cases of Fig. 3b and Fig. 3d), the inconsistency is more likely to happen because the label description degrees of the top label(s) is(are) easily to be surpassed by that of other labels. That is, evenly-distributed label distributions deserve more attention than unevenly-distributed ones. Here, we use information entropy [35] to measure the *uniformity* of label distributions and re-weight instances *w.r.t.* the information entropy. Further, the labels with higher label description degrees deserve more attention, which inspires us to re-weight labels *w.r.t.* the label description degrees. Next, we propose the following weighted L_1-norm loss

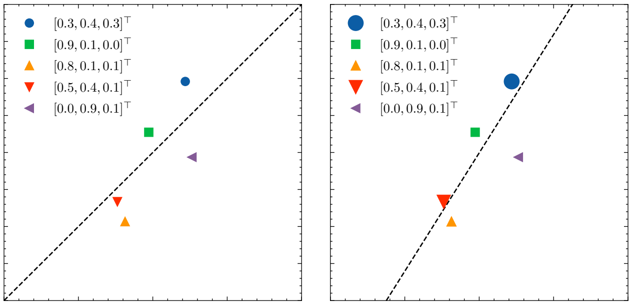
$$E_{\boldsymbol{x}} \cdot \sum_{j=1}^{m} d_{\boldsymbol{x}}^{y_j} \cdot |g_{\boldsymbol{x}}^{y_j} - d_{\boldsymbol{x}}^{y_j}| = \sum_{j=1}^{m} d_{\boldsymbol{x}}^{y_j}\, E_{\boldsymbol{x}} |g_{\boldsymbol{x}}^{y_j} - d_{\boldsymbol{x}}^{y_j}|, \tag{5}$$

where $E_{\boldsymbol{x}} = -\sum_{y\in\mathcal{Y}} d_{\boldsymbol{x}}^{y} \ln d_{\boldsymbol{x}}^{y}$ is the information entropy of $\boldsymbol{d_x}$. That is, the jth label is weighted with the weight $d_{\boldsymbol{x}}^{y_j}$, and $\boldsymbol{x}$ is assigned with the weight $E_{\boldsymbol{x}}$. Define $\omega_{i,j} = d_{\boldsymbol{x}_i}^{y_j} E_{\boldsymbol{x}_i}$.



(a) SLL: $\ell_1 = 0.5$, $\hat{g}(\mathbf{x}) = g^*(\mathbf{x})$ (b) SLL: $\ell_1 = 0.2$, $\hat{g}(\mathbf{x}) \neq g^*(\mathbf{x})$ (c) MLL:$\ell_1 = 0.4$, $\hat{g}(\mathbf{x}) = g^*(\mathbf{x})$ (d) MLL: $\ell_1 = 0.2$, $\hat{g}(\mathbf{x}) \neq g^*(\mathbf{x})$

Fig. 3. Examples to illustrate the inconsistency, where ℓ_1 denotes the L_1-norm loss. The red bar represents the ground-truth label distribution, and the blue bar denotes the learned label distribution. Fig. 3a and Fig. 3b demonstrate the case of SLL, and Fig. 3c and Fig. 3d manifest the case of MLL. The cases of Fig. 3a and Fig. 3c have small classification losses, but the cases of Fig. 3b and Fig. 3d have small L_1-norm losses.



(a) Unweighted LDL (b) Re-weighting LDL

Fig. 4. Illustration of the re-weighting LDL, where the sizes of the markers indicate the scales of the weights. Note that the instances with evenly-distributed label distributions are assigned with higher weights and have smaller losses in the re-weighting LDL.

Then, substituting Eq. (5) into Eq. (4), the re-weighting LDL is further formulated as the following,

$$\min_{\boldsymbol{W}} \sum_{i=1}^{n} \sum_{j=1}^{m} \omega_{i,j} \cdot |g_{\boldsymbol{x}_i}^{y_j} - d_{\boldsymbol{x}_i}^{y_j}| + \frac{\lambda_1}{2} \|\boldsymbol{W}\|_{\mathrm{F}}^2. \tag{6}$$

Compared with Eq. (4), where instances and labels are treated equally, the re-weighting LDL mainly focuses on the labels with higher label description degrees and the instances where the inconsistency is more likely to happen. Thereby, the objective inconsistency will be alleviated.

Fig. 4 illustrates the effect of the re-weighting LDL *w.r.t.* information entropy. The points with evenly-distributed label distributions are assigned with higher weights and have smaller losses in the re-weighting LDL.

3.3.3 LDL with Large Margin

The re-weighting LDL alleviates the objective inconsistency from the perspective of LDL. To further solve the objective inconsistency and improve the classification performance, we introduce the large margin [22] to LDL.

As Geng *et al.* [7] pointed out, the predicted label description degree can be regarded as the confidence of the corresponding label. Accordingly, we can put more confidence to the top label(s). Specifically, let $\boldsymbol{Y}_i$ denote the index set of the top label(s) for $\boldsymbol{x}_i$, and $\bar{\boldsymbol{Y}}_i$ be the complementary set of $\boldsymbol{Y}_i$. We encourage $g_{\boldsymbol{x}_i}^{y_j}$ to be larger than $g_{\boldsymbol{x}_i}^{y_l}$ by a margin ρ $(0 < \rho < 1)$ for $j \in \boldsymbol{Y}_i$ and $l \in \bar{\boldsymbol{Y}}_i$. Then, the problem (6) can be re-cast as the following,

$$\begin{aligned} \min_{\boldsymbol{W},\boldsymbol{\xi}} \; & \sum_{i,j} \omega_{i,j} \cdot |g_{\boldsymbol{x}_i}^{y_j} - d_{\boldsymbol{x}_i}^{y_j}| + \frac{\lambda_1}{2} \|\boldsymbol{W}\|_{\mathrm{F}}^2 + \lambda_2 \sum_{i=1}^{n} \sum_{j,l} \frac{\xi_{i,j,l}}{\rho} \\ \text{s.t.:} \; & \forall i \in [n],\ \forall j \in \boldsymbol{Y}_i,\ \forall l \in \bar{\boldsymbol{Y}}_i \\ & g_{\boldsymbol{x}_i}^{y_j} - g_{\boldsymbol{x}_i}^{y_l} \geq \rho - \xi_{i,j,l}, \\ & \xi_{i,j,l} \geq 0, \end{aligned} \tag{7}$$

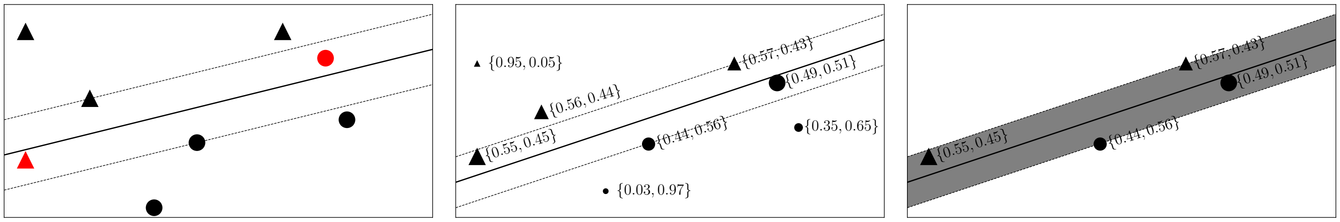
where $\xi_{i,j,l}$ is the slack variable, $[n]$ stands for the set $\{1, 2, \cdots, n\}$, and λ_2 is a trade-off parameter balancing the importance of re-weighting schemes and large margin. For SLL, $\boldsymbol{Y}_i$ has one element, *i.e.*, the index of the label with the highest label description degree of $\boldsymbol{x}_i$. For MLL, $\boldsymbol{Y}_i$ is the index set of the positive labels for $\boldsymbol{x}_i$. The second constraint of Eq. (7) encourages the predicted label description degree(s) of the top label(s) to be larger than those of other labels by a margin of ρ. As a result, the induced classifier will prefer the top label(s) as the predicted label(s), which further solves the inconsistency.

3.3.4 Advantages of Re-weighting and Large Margin

RWLM-LDL seeks a balance between re-weighting and large margin. The advantages are as follows:

1) Large margin improves the classification performance of *RWLM-LDL*. The instances outside the marginal hyper-planes have already been correctly classified by large margin classifier. Thereby, the error of *RWLM-LDL* is only determined by the instances inside the marginal hyperplanes.
2) The re-weighting further boosts the classification performance of *RWLM-LDL*. The instances inside the marginal hyper-planes tend to have evenly-distributed label distributions, which are assigned with higher weights.
3) To summarize, large margin correctly classifies the instances outside the marginal hyperplanes, and the re-weighting tends to correctly classify the instances inside the marginal hyperplanes. Thereby, the objective inconsistency is solved.

Fig. 5 visualizes the advantages of re-weighting and large margin by examples in the case of binary classification. The solid lines and the dash lines represent separating hyperplanes and marginal hyperplanes, respectively. Fig.



(a) Large margin classifier (b) The advantage of re-weighting LDL (c) The advantage of large margin classifier

Fig. 5. Visualization of the advantages of re-weighting and large margin classifier in the case of binary classification. The solid lines and the dash lines are separating hyper-planes and marginal hyper-planes, respectively. Fig. 5a illustrates a large margin classifier with two misclassified points highlighted in red. Fig. 5b showcases the advantage of re-weighting. The sizes of the markers indicate the scales of weights. Fig. 5c illuminates the advantage of large margin. The points outside the marginal hyper-planes have already been correctly classified.

5a shows a large margin classifier with two misclassified points highlighted in red. Fig. 5b illustrates the advantage of re-weighting LDL, where the label distributions of points are given, and the sizes of the markers indicate the scales of the weights. Generally, the points inside the marginal hyperplanes have higher uncertainty whose label distributions tend to be evenly distributed and are assigned with higher weights. Thereby, the separating hyperplane is adjusted to correctly position the points inside the marginal hyperplanes. Fig. 5c illuminates the advantage of large margin, where the points outside the marginal hyperplanes have already been correctly classified. As a result, the error is completely determined by the points inside the marginal hyperplanes (*i.e.*, the shaded area).

3.3.5 Optimization

It's difficult to solve the problem (7) directly due to a large number of constraints. There are $\mathcal{O}(nm^2)$ constraints, which may raise the problem of scalability. To address the computational problem, we use the Stochastic Gradient Descent (SGD) method [36] to solve the cast optimization problem. First, the problem (7) can be written equivalently as the following unconstrained optimization problem,

$$\min_{\boldsymbol{W},\boldsymbol{\xi}} \frac{\lambda_1}{2}\|\boldsymbol{W}\|_{\mathrm{F}}^2+\sum_{i}^{n}\sum_{j=1}^{m}\omega_{i,j}\cdot\left|g_{\boldsymbol{x}_i}^{y_j}-d_{\boldsymbol{x}_i}^{y_j}\right| +\lambda_2\sum_{i=1}^{n}\sum_{j\in\boldsymbol{Y}_i,l\in\bar{\boldsymbol{Y}}_i}\max(0,1-g_{\boldsymbol{x}_i}^{y_j}/\rho+g_{\boldsymbol{x}_i}^{y_l}/\rho).$$

Then, the sub-gradient is calculated to update the parameters. For a mini-batch of θ training examples indexed by $(i' : i' + \theta)$, the sub-gradient is got through

$$\begin{aligned}\triangledown_{\boldsymbol{w}}^{(i':i'+\theta)} &= \lambda_1\boldsymbol{w}+\sum_{i=i'}^{i'+\theta}\sum_{j=1}^{m}\omega_{i,j}\cdot\operatorname{sign}(g_{\boldsymbol{x}_i}^{y_j}-d_{\boldsymbol{x}_i}^{y_j})\frac{\partial g_{\boldsymbol{x}_i}^{y_j}}{\partial\boldsymbol{w}}\\ &+\frac{\lambda_2}{\rho}\sum_{i=i'}^{i'+\theta}\sum_{j\in\boldsymbol{Y}_i,l\in\bar{\boldsymbol{Y}}_i}\mathbb{I}(g_{\boldsymbol{x}_i}^{y_j}-g_{\boldsymbol{x}_i}^{y_l}<\rho)\left(\frac{\partial g_{\boldsymbol{x}_i}^{y_l}}{\partial\boldsymbol{w}}-\frac{\partial g_{\boldsymbol{x}_i}^{y_j}}{\partial\boldsymbol{w}}\right),\end{aligned}\tag{8}$$

where $\operatorname{sign}(\cdot)$ is the sign function. The gradient of the maximum entropy model is got through

$$\frac{\partial g_{\boldsymbol{x}_i}^{y_j}}{\partial\boldsymbol{w}_l}=\left[\mathbb{I}(j=l)\cdot g_{\boldsymbol{x}_i}^{y_j}-g_{\boldsymbol{x}_i}^{y_j}\cdot g_{\boldsymbol{x}_i}^{y_l}\right]\cdot\boldsymbol{x}_i.\tag{9}$$

The Adam [37] is applied to solve the problem (7). The details of the algorithm are presented in Alg. 1. First, line 4 calculates the sub-gradient (g_t). Second, lines 5 and 6 estimate biased first moment (m_t) and second raw moment (v_t), respectively. Next, lines 7 and 8 compute the biased-corrected first moment ($\hat{m}_t$) and second raw moment ($\hat{v}_t$) [37], respectively. Finally, line 9 updates the parameter $\boldsymbol{W}$.

Alg. 1 has $\mathcal{O}(\theta km^2qT)$ time complexity, where k is the maximum number of positive labels for MLL (generally, $k \ll m$ [4]), and $k = 1$ for SLL. First, the calculation of the model's output $[g_{\boldsymbol{x}_i}^{y_j}]_{i,j}$ has $\mathcal{O}(\theta mq)$ complexity for a mini-batch. Second, by Eq. (9), the computation of $\partial g_{\boldsymbol{x}_i}^{y_j}/\partial\boldsymbol{w}_l$ has $\mathcal{O}(q)$ complexity. Next, the second term on the right-hand side of Eq. (8) involves θm times gradient calculation and has $\mathcal{O}(\theta mq)$ complexity. Moreover, the third term on the left-hand side of Eq. (8) needs θkm times gradient computation at most and has $\mathcal{O}(\theta kmq)$ complexity. Thereby, the computation of Eq. (8) has $\mathcal{O}(\theta kmq)$ complexity, and the calculation of gradient *w.r.t.* $\boldsymbol{W}$ needs $\mathcal{O}(\theta km^2q)$ complexity. Given T iterations, the total time complexity is $\mathcal{O}(\theta km^2qT)$.

Algorithm 1 The *RWLM-LDL* algorithm

```
Input: Training set $S$, parameters $\lambda_1$ and $\lambda_2$, margin $\rho$,
    batch size $\theta$, step size $\alpha$, and number of iterations $T$
Output: $\boldsymbol{W}$
 1: initialize $\boldsymbol{W}_0 \leftarrow \boldsymbol{0}$, $\beta_1 \leftarrow 0.9$, $\beta_2 \leftarrow 0.999$, $m_0 \leftarrow 0$,
    $v_t \leftarrow 0$
 2: for $t = 1$ to $T$ do
 3:   generate a mini-batch indexed by $(i_t : i_t + \theta)$
 4:   $g_t \leftarrow [\triangledown_{\boldsymbol{w}_1}^{(i_t:i_t+\theta)}, \cdots, \triangledown_{\boldsymbol{w}_m}^{(i_t:i_t+\theta)}]$ by Eq. (8)
 5:   $m_t \leftarrow \beta_1 m_{t-1} + (1-\beta_1)\cdot g_t$
 6:   $v_t \leftarrow \beta_2 v_{t-1} + (1-\beta_2)\cdot g_t^2$
 7:   $\hat{m}_t \leftarrow m_t/(1-\beta_1^t)$
 8:   $\hat{v}_t \leftarrow v_t/(1-\beta_2^t)$
 9:   $\boldsymbol{W}_t \leftarrow \boldsymbol{W}_{t-1} - \alpha\cdot\hat{m}_t/(\sqrt{\hat{v}_t}+\epsilon)$
10: end for
11: return $\boldsymbol{W}_t$
```

4 Theoretical Results

This section studies the generalization and discrimination of *RWLM-LDL* by setting up upper bounds on the *error probability* (stochastic setting) and the *expected 0/1 loss* (deterministic setting), respectively. The generalization is due to LDL with L_1-norm loss, and the discrimination is credited to large margin.

RWLM-LDL adopts the maximum entropy model that can be regarded as a function combination of the softmax function and a multi-output linear regression function. Let $\mathrm{SF} : \mathbb{R}^m \to \mathbb{R}^m$ stand for the softmax function, and $\mathcal{F}$ be a family of multi-output linear functions defined by

$$\mathcal{F} := \{\boldsymbol{x} \mapsto [\boldsymbol{w}_1 \cdot \boldsymbol{x}, \boldsymbol{w}_2 \cdot \boldsymbol{x}, \ldots, \boldsymbol{w}_m \cdot \boldsymbol{x}]^T, \|\boldsymbol{w}_j\|_2 \le \Lambda_1\},$$

where $\Lambda_1 > 0$ is a constant. Then, a hypothesis set for the maximum entropy model can be defined by

$$\mathcal{G} := \{\boldsymbol{x} \mapsto \mathrm{SF} \circ f(\boldsymbol{x}) : f \in \mathcal{F}\}, \tag{10}$$

where $\circ$ is the function combination operator. Further assume that $\sup_{\boldsymbol{x}\in\mathcal{X}} \|\boldsymbol{x}\|_2 \le \Lambda_2$, where $\Lambda_2 > 0$ is a constant.

4.1 Upper Bound on Error Probability

This subsection establishes the generalization of *RWLM-LDL* by setting up an upper bound on the error probability. For simplicity, we further assume that the given label distribution function is the normalized conditional probability distribution function. Then, Eq. (3) reduces to

$$L(\hat{g}) - L(g^*) \le 2\mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}_\mathcal{X}} \left[\sum_{j=1}^m |g_{\boldsymbol{x}}^{y_j} - d_{\boldsymbol{x}}^{y_j}| \right]. \tag{11}$$

Accordingly, to bound the error probability, it suffices to bound the expected L_1-norm loss. For any $g \in \mathcal{G}$, let $\hat{R}_{\ell_1}(g)$ denote the empirical L_1-norm loss defined by

$$\hat{R}_{\ell_1}(g) = \frac{1}{n} \sum_{i=1}^n \sum_{j=1}^m |g_{\boldsymbol{x}_i}^{y_j} - d_{\boldsymbol{x}_i}^{y_j}|,$$

where the re-weighting schemes are suppressed for the convenience of analysis. Next, the following theorem shows the generalization of *RWLM-LDL*.

Theorem 3 (Upper Bound on Error Probability)**.** *For any $g \in \mathcal{G}$, let $\hat{g}$ denote the induced classifier as Eq. (1) or Eq. (2). Then, for any $\delta > 0$, with probability at least $1 - \delta$, the following bound holds for all $g \in \mathcal{G}$,*

$$L(\hat{g}) \le L(g^*) + 2\hat{R}_{\ell_1}(g) + \frac{8\sqrt{2}\Lambda_1\Lambda_2 m^2}{\sqrt{n}} + 4\sqrt{\frac{\log 1/\delta}{2n}}.$$

Theorem 3 bounds the error probability by the sum of four terms. The first one is the minimum error probability, the second one is (2×) the empirical L_1-norm loss, the third one is an upper bound on the Rademacher complexity [38], and the last one can be ignored. To achieve a small error probability, it suffices to minimize L_1-norm loss. The proof of the theorem is in the Supplementary Material.

4.2 Upper Bounds on Expected 0/1 Loss

In this subsection, we borrow the margin theory [22], [27] to derive upper bounds on the expected 0/1 loss. Specifically, $g^*(\boldsymbol{x})$ is regarded as the ground-truth label(s) for $\boldsymbol{x}$. For any $g \in \mathcal{G}$, define the empirical margin loss by

$$\hat{R}_\rho(g) = \frac{1}{n} \sum_{i=1}^n \sum_{j\in\boldsymbol{Y}_i, l\in\bar{\boldsymbol{Y}}_i} \max\left(0, 1 - g_{\boldsymbol{x}_i}^{y_j}/\rho + g_{\boldsymbol{x}_i}^{y_l}/\rho\right),$$

and define the expected 0/1 loss by

$$R(g) = \mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}_\mathcal{X}} \left[\mathbb{I}(\hat{g}(\boldsymbol{x}) \ne g^*(\boldsymbol{x}))\right],$$

where $\hat{g}$ is the induced classifier according to g as defined in Eq. (1) or Eq. (2). For MLL, the operator $\ne$ is taken over two sets, which returns 0 if they are equal and 1 otherwise.

For SLL, a margin bound on the expected 0/1 loss is established by the following theorem.

Theorem 4 (Margin Bound on Expected 0/1 Loss for SLL)**.** *Fix $\rho > 0$. For any $g \in \mathcal{G}$, let $\hat{g}$ denote the induced SLL classifier as Eq. (1). Then, for any $\delta > 0$, with probability at least $1 - \delta$, the following bound holds for all $g \in \mathcal{G}$,*

$$R(g) \le \hat{R}_\rho(g) + \frac{8\Lambda_1\Lambda_2 \exp(2\Lambda_1\Lambda_2)m^2}{\rho\sqrt{n}} + \sqrt{\frac{\log 1/\delta}{2n}}.$$

For MLL, a margin bound on the expected 0/1 loss is established by the following theorem.

Theorem 5 (Margin Bound on Expected 0/1 Loss for MLL)**.** *Fix $\rho > 0$. For any $g \in \mathcal{G}$, let $\hat{g}$ denote the induced MLL classifier as Eq. (2). Then, for any $\delta > 0$, with probability at least $1 - \delta$, the following bound holds for all $g \in \mathcal{G}$,*

$$R(g) \le \hat{R}_\rho(g) + \frac{12\Lambda_1\Lambda_2 \exp(2\Lambda_1\Lambda_2)m^2}{\rho\sqrt{n}} + \sqrt{\frac{\log 1/\delta}{2n}}.$$

Theorems 4 and 5 show that the expected 0/1 loss of *RWLM-LDL* can be upper bounded by the sum of three terms. The first one is the empirical margin loss, the second one is an upper bound on the Rademacher complexity [38], and the last one can be ignored. Accordingly, to minimize the 0/1 loss, it suffices to minimize the margin loss. The proofs are deferred to the Supplementary Material.

5 Experiments

5.1 Experimental Datasets

We conduct the experiments thoroughly on 18 real-world datasets. The characteristics of the datasets are summarized in Table 2, where r_1 and r_2 are defined in Section 5.5.3.

The first 15 datasets[2] are collected by Geng [5], among which the 1st to the 10th (from *Alpha* to *Spoem*) are collected from the biological experiments [39], the *Gene* is obtained from a research on the relation between human gene and diseases [40], the *Natural_Scene* is got by transforming the inconsistent rankings of natural scene images to compatible label distributions [6], the *SBU_3DFE* and *SJAFFE* result from two facial expression image databases *BU_3DFE* [41] and *JAFFE* [3], and the *Movie* is collected from the user ratings on movies [42]. Moreover, the 16th dataset *FBP5500*[3] is about facial beauty perception [11]. We use the trained *ResNet* [43] provided by the authors to extract 512-dimensional features. The 17th dataset *RAF_ML*[4] is a multi-label facial expression dataset, where each image is represented by 2000-dimensional Deep Bi-Manifold CNN features [10] and described by a label distribution. The last dataset *Mediamill* is a large-scale multi-label dataset [44]. We borrow the label enhancement [45] technique to recover the label distributions from the logical labels in the dataset. In the sequel, each dataset is denoted by its first three letters (Spo5" and "Spoem" are denoted by "Spo5" and "Spoe" to distinguish them from "Spo").

[2] http://palm.seu.edu.cn/xgeng/LDL/index.htm
[3] https://github.com/HCIILAB/SCUT-FBP5500-Database-Release
[4] http://www.whdeng.cn/RAF/model2.html

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2021.3082623, IEEE Transactions on Pattern Analysis and Machine Intelligence

TABLE 2
Statistics of the experimental datasets.

ID.	Dataset	n	q	m	r_1	r_2
1	Alpha	2,465	24	18	1.02	0.67
2	Cdc	2,465	24	15	1.03	0.70
3	Cold	2,465	24	4	1.30	0.69
4	Diau	2,465	24	7	1.10	0.77
5	Dtt	2,465	24	4	1.27	0.69
6	Elu	2,465	24	14	1.03	0.72
7	Heat	2,465	24	6	1.11	0.77
8	Spo	2,465	24	6	1.13	0.76
9	Spo5	2,465	24	3	1.64	0.56
10	Spoem	2,465	24	2	6.9e4	0.33
11	Gene	17,892	36	68	8.88	0.36
12	Natural_Scene	2,000	294	9	1.5e2	0.36
13	SBU_3DFE	2,500	243	6	1.19	0.74
14	SJAFFE	213	243	6	1.16	0.74
15	Movie	7,755	1,869	5	2.37	0.66
16	FBP5500	5,500	512	5	16.00	0.43
17	RAF_ML	4,908	2,000	6	3.90	0.46
18	Mediamill	43,907	120	101	1.04	0.30

5.2 SLL Predictive Experiment

We conduct the SLL experiments on the first 16 datasets (*RAF_ML* and *Mediamill* are multi-label datasets). Here, we regard the label having the largest label description degree as the ground-truth label. Moreover, *0/1 loss* is used to evaluate the performance of the comparing algorithms.

5.2.1 Methodology

We compare *RWLM-LDL* with five state-of-the-art LDL methods, including *BFGS-LDL*, *StructRF*, *RSSR-LDL21*, *LDLFs*, and *LDL-SCL*. We also compare *RWLM-LDL* with three SLL methods, including *ovrSVM*, *wSVM*, and *SAMME*, where *ovrSVM* and *wSVM* are two large margin classifiers, and *wSVM* and *SAMME* are two re-weighting methods. The details of the algorithms are as follows:

1) *SVM* [23]: This method is a multi-class SVM, which uses the *one-vs-rest* decision functions.
2) *wSVM* [46]: This method is a weighting SVM. Similar to *RWLM-LDL*, we weight each instance by the information entropy of its label distribution.
3) *SAMME* [31]: This method is a multi-class Adaboost method, which re-weights each sample *w.r.t.* the training loss of that sample.
4) *BFGS-LDL* [5]: This approach applies the maximum entropy model to learn label distribution, where KL divergence is used as the learning metric.
5) *LDL-SCL* [19]: This approach encodes label correlation as additional features and jointly learns label distribution and label correlation.
6) *RSSR-LDL21* [18]: This method applies the regularized sample self-representation technique to LDL, which is cast as an $L_{2,1}$-norm least-squares problem.
7) *StructRF* [17]: This approach is an ensemble method, which uses the structured random forest to learn label distribution and exploits structural information among different classes by clustering.
8) *LDLFs* [16]: This approach uses the differentiable decision trees to learn label distribution. Here, we use the shallow stand-alone version [16]

The parameters of the methods are set as follows. For *SVM* and *wSVM*, the implementation is based on *libsvm* [46], and the *RBF* kernel is used. For *SAMMA*, *Logistic Regression* (*LR*) is used as the base classifier since the model of *RWLM-LDL* can be viewed as a multinomial *LR*, and the number of estimators is set to 50. For *RSSR-LDL21*, the regularization parameter is tuned from the set $\{10^{-3}, \cdots, 10^{3}\}$. For other baselines, the default parameters are applied. For *RWLM-LDL*, λ_1 is tuned from $\{10^{-4}, \cdots, 1\}$, λ_2 is tuned from the candidate set $\{10^{-3}, \cdots, 10^{3}\}$, and $\rho = 0.1$. We tune the parameters of each method by ten-fold cross-validation. Then, each method with the best parameters is run for ten times random data partitions (90% for training and 10% for testing), and the average performance is reported.

5.2.2 SLL Results

Table 3 tabulates the experimental results of each approach in terms of 0/1 loss. To further compare the relative performance of *RWLM-LDL* against each comparing method, we conduct the pairwise t-test at 0.05 significance level and use •/○ to indicate whether *RWLM-LDL* is statistically superior/inferior to the comparing algorithm.

From Table 3, *RWLM-LDL* ranks first in 68.8% cases (11 out of 16) and achieves significantly superior performance against the comparing algorithms in 69.5% cases (89 wins out of 128 tests). *RWLM-LDL* achieves statistically superior or at least comparable performance against the compared LDL methods. The reason is that the compared LDL methods fail to consider the objective inconsistency, which results in sub-optimal decision functions. In contrast, *RWLM-LDL* addresses the objective inconsistency using re-weighting and large margin, which yields better classification performance. Besides, *RWLM-LDL* outperforms the compared SLL methods by a margin. The reason lies in the richer information of label distribution compared with single label [5]. Accordingly, *RWLM-LDL* learning from label distribution has better classification performance than the compared SLL methods learning from single labels. Furthermore, *wSVM* achieves better performance than *SVM*, which validates the effect of our re-weighting method since *wSVM* and *SVM* are only different in that *wSVM* uses the re-weighting.

To summarize, *RWLM-LDL* achieves competitive performances in terms of 0/1 loss. That is, *RWLM-LDL* has better classification performance in SLL.

5.3 MLL Predictive Experiment

To properly evaluate the performance of the comparing methods for MLL, we firist select 12 datasets with $m \geq 6$ and then conduct the experiments on the selected datasets.

5.3.1 Label Binarization

For *RAF_ML* and *Mediamill*, the ground-truth labels are known. For other datasets, only the label distributions are available. For comparison and evaluation, we generate multi-label using the binarization strategy proposed in [45] as follows. For each instance $\boldsymbol{x}$, first initialize an empty set $\mathcal{Y}^{+}$. Next, add into $\mathcal{Y}^{+}$ the label with the highest label

TABLE 3
SLL predictive results (mean±std) on 16 datasets, where ●/○ indicates whether RWLM-LDL is statistically superior/inferior to the comparing algorithms (pairwise t-test at 0.05 significance level), and the win/tie/lose (w./t./l.) counts are summarized in the last row.

Dataset	SVM	wSVM	SAMME	BFGS-LDL	LDL-SCL	RSSR-LDL21	StructRF	LDLFs	RWLM-LDL
Alp	0.791±0.024	**0.782±0.021**	0.787±0.025●	0.886±0.017●	0.909±0.021●	0.886±0.017●	0.871±0.024●	0.880±0.028●	**0.782±0.024**
Cdc	0.832±0.022●	0.824±0.019	0.830±0.023●	0.824±0.028	0.825±0.025●	0.824±0.026	0.826±0.032	0.823±0.019	**0.817±0.025**
Col	0.713±0.022●	0.607±0.036●	0.580±0.032●	0.577±0.032	0.578±0.034	0.580±0.032●	**0.555±0.034**	0.563±0.034	0.571±0.032
Dia	0.734±0.028●	0.684±0.048●	0.663±0.036	0.693±0.025	0.699±0.022●	0.695±0.027	0.687±0.023	0.693±0.032●	**0.658±0.036**
Dtt	0.732±0.034●	0.660±0.034●	0.645±0.041	0.643±0.038	0.643±0.037	0.632±0.042	0.625±0.026	**0.623±0.031**	0.637±0.033
Elu	0.808±0.026●	0.809±0.027●	0.810±0.026●	0.899±0.023●	0.906±0.017●	0.901±0.022●	0.915±0.020●	0.893±0.022●	**0.799±0.030**
Hea	0.721±0.032●	0.705±0.023●	0.688±0.035	0.693±0.031	0.703±0.026●	0.695±0.028●	**0.663±0.024**	0.682±0.021●	0.669±0.021
Spo	0.628±0.023●	0.559±0.032●	0.548±0.029●	0.542±0.029●	0.557±0.034●	0.548±0.029●	0.572±0.028●	0.578±0.038●	**0.540±0.029**
Spo5	0.622±0.045●	0.568±0.036●	0.543±0.031●	0.547±0.029●	0.583±0.031●	0.544±0.033●	**0.520±0.031**	0.536±0.022	0.524±0.029
Spoe	0.521±0.035●	0.516±0.033●	0.425±0.018●	0.436±0.030●	0.439±0.019●	0.438±0.029●	0.421±0.019●	0.428±0.027	**0.405±0.024**
Gen	0.969±0.005●	0.941±0.006●	0.928±0.006	0.955±0.005●	0.959±0.020●	0.953±0.002●	0.969±0.004●	0.962±0.007●	**0.927±0.006**
NAT	0.539±0.038●	0.587±0.031●	0.610±0.034●	0.596±0.035●	0.659±0.035●	0.557±0.030●	0.528±0.039●	0.735±0.067●	**0.434±0.040**
SBU	0.762±0.039●	0.688±0.041●	0.710±0.038●	0.550±0.033●	0.515±0.036	0.513±0.034	**0.494±0.030**	0.635±0.035●	0.508±0.034
SJA	0.560±0.114●	0.512±0.097●	0.748±0.119●	0.456±0.089●	0.751±0.109●	0.505±0.101	0.606±0.107●	0.503±0.079●	**0.404±0.100**
Mov	0.676±0.024●	0.437±0.017●	0.428±0.018	0.417±0.017	0.428±0.015	0.419±0.016	0.447±0.015●	0.443±0.018●	**0.416±0.016**
FBP	0.352±0.049●	0.281±0.023●	0.392±0.022●	**0.210±0.019**	0.212±0.014	0.213±0.015	0.227±0.014●	0.261±0.012●	**0.210±0.014**
w./t./l.	**15/1/0**	**14/2/0**	**11/5/0**	**9/7/0**	**11/5/0**	**9/7/0**	**9/7/0**	**11/5/0**	

description degree. Then, calculate the sum of the label description degrees of all labels in $\mathcal{Y}^+$ and denote it by H. If $H < 0.5$, add from the label set (excluding $\mathcal{Y}^+$) the label with the highest label description degree into $\mathcal{Y}^+$. The above process continues until $H \geq 0.5$. Eventually, the labels in $\mathcal{Y}^+$ are regarded as positive labels. For LDL algorithms, the predicted labels are generated by applying the label binarization to the predicted label distribution.

5.3.2 Methodology

We use five MLL metrics [4] to evaluate the performance of the comparing algorithms, including *Hamming loss*, *One error*, *Coverage*, *Ranking loss*, and *Average precision*. We compare *RWLM-LDL* with the same LDL methods as Section 5.2 and three MLL methods, including *BR-wSVM*, *RELIAB*, and *LIMO*, details of which are as follows.

1) *BR-wSVM* [4]: This method decomposes an MLL problem into m independent binary problems and then employs the $wSVM$ to each one.
2) *RELIAB* [13]: It first applies local k nearest neighbors reconstruction to estimate the *Relative Labeling-Importance* (*RLI*) of labels. Then, a predictive model is learned directly on the *RLI* by jointly minimizing KL divergence and pairwise ranking loss.
3) *LIMO* [26]: This is a large margin method for MLL, which maximizes two margins, including the label-wise margin and the instance-wise margin.

The settings of the parameters for each method are as follows. For *RWLM-LDL* and the compared LDL methods, the settings are as Section 5.2.1. For *RELIAB*, τ is selected from the candidate set $\{0.1, 0.15, \cdots, 0.5\}$, λ is chosen from the pool $\{10^{-3}, \cdots, 10\}$, and $\rho = 0.3$. For *LIMO*, λ_1 and λ_2 are tuned from the candidate set $\{10^{-3}, \cdots, 10^3\}$. We tune the parameters of each method by ten-fold cross-validation and run each method with the best parameter for ten times random data partitions (90% for training and 10% for testing). Then, the average performance is reported.

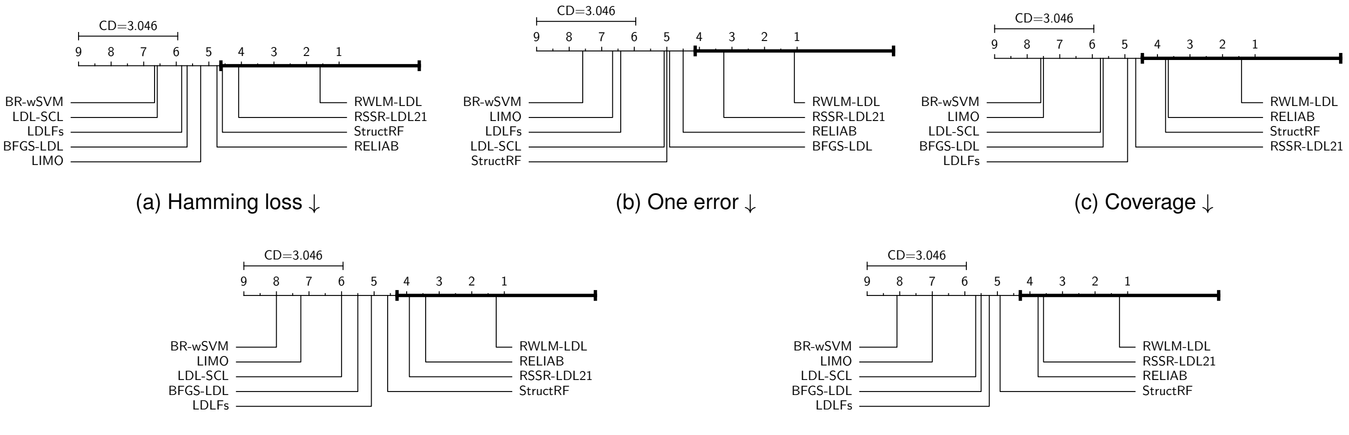
5.3.3 MLL Results

Table 4 reports the results of the comparing methods in terms of each metric on 12 datasets, where the best results are highlighted in boldface, the "↓" indicates "the smaller the better", and the "↑" means "the larger the better".

To study the relative performance among the comparing algorithms, we conduct the *Friedman* test [47] that compares multiple algorithms over several datasets. The results are summarized in Table 5. At confidence level of 0.05, the Friedman statistics on all metrics are larger than the critical value 2.0454 (9 comparing algorithms on 12 datasets). Therefore, the null hypothesis that the comparing algorithms have equal performance is rejected.

To show whether *RWLM-LDL* achieves competitive performance against other comparing methods, the *Bonferroni-Dunn* test [48] is further conducted by regarding *RWLM-LDL* as the control algorithm. For the test, the *Critical Difference* (CD) equals 3.046 at 0.05 significance level, and the performance of one comparing algorithm is significantly different from that of *RWLM-LDL* if their average ranks over the datasets differ by at least one CD. Fig. 6 shows the CD diagrams [47] in terms of each metric, where the average ranks of each comparing algorithm are marked along the axis (in decreasing order). Any method whose average rank is within one CD to that of *RWLM-LDL* is interconnected by a thick line along the axis, and the methods not connected with *RWLM-LDL* are considered to have a significantly different performance from *RWLM-LDL*.

Table 4 shows that *RWLM-LDL* ranks first in 76.7% cases and achieves the best average performance (lowest average rank) in terms of all MLL metrics. Fig. 6 shows that among the compared LDL methods, *RWLM-LDL* significantly outperforms *LDL-SCL*, *BFGS-LDL*, and *LDLFs*, and achieves statistically better or comparable performance against *StructRF* and *RSSR-LDL21* in terms of all metrics. The reason lies in that the LDL baselines have not considered the objective inconsistency. In contrast, *RWLM-LDL*



(a) Hamming loss ↓

(b) One error ↓

(c) Coverage ↓

(d) Ranking loss ↓

(e) Average precision ↑

Fig. 6. CD diagrams of the Bonferroni-Dunn tests, where RWLM-LDL is set as the control algorithm. The algorithms not connected with RWLM-LDL are considered to have significantly different performance from RWLM-LDL.

tackles the objective inconsistency by re-weighting and large margin, which results in better classification performance. *RWLM-LDL* significantly outperforms *BR-wSVM* and *LIMO* in terms of all metrics because label distribution contains richer supervision information than 0/1 label. Furthermore, *RWLM-LDL* is comparable to *RELIAB* in terms of all metrics except *hamming loss and one error* because *RELIAB* directly learns from the *RLI* [13] whose information is comparable to label distribution, and partially solves the inconsistency by optimizing pairwise ranking loss. However, *RWLM-LDL* has better mean performance than *RELIAB*.

In summary, *RWLM-LDL* achieves competitive MLL predictive performance against other comparing algorithms, which validates the effectiveness of *RWLM-LDL* in MLL.

5.4 Ablation Study

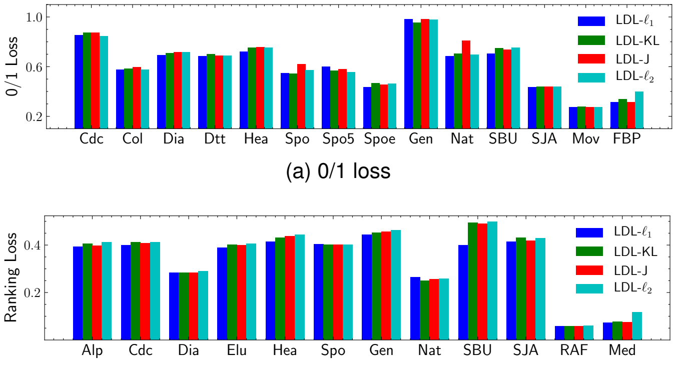
This subsection conducts ablation studies. We study the effectiveness of the three components of *RWLM-LDL*, *i.e.*, L_1-norm loss, re-weighting schemes, and larger margin. Besides, we investigate the advantages of label distribution.

Three degenerate versions of *RWLM-LDL* are derived, including (i) *LDL-ℓ_1*, which only uses L_1-norm loss ($\omega_{i,j} = 1$ and $\lambda_2 = 0$), (ii) *RW-LDL*, which only applies the weighting schemes ($\lambda_2 = 0$), and (iii) *LM-LDL*, which suppresses the re-weighting schemes ($\omega_{i,j} = 1$).

5.4.1 Effectiveness of L_1-norm Loss

The commonly used loss functions for LDL include KL divergence, Jeffrey's divergence, and L_2-norm loss [5]. To investigate the effectiveness of L_1-norm loss, we derive three variants of *LDL-ℓ_1* by replacing L_1-norm loss with KL divergence, Jeffrey's divergence, and L_2-norm loss, which are denoted by *LDL-KL*, *LDL-J*, and *LDL-ℓ_2*, respectively. Then, *LDL-ℓ_1* is compared with each of the variants.

We evaluate the performance of *LDL-ℓ_1* and the variants in terms of each metric on the experimental datasets. Due to limited space, we only present the detailed results in terms of 0/1 loss and ranking loss in Fig. 7. As shown in Fig. 7, LDL-ℓ_1 has the best mean performance. Besides, it has the sub-optimal performance for some datasets, such as *Gene* and and *Natural_Scene*. The reason is that L_1-norm loss is non-smooth, which may be hard to optimize sometimes.



(a) 0/1 loss

(b) Ranking loss

Fig. 7. Performance comparison among LDL-ℓ_1, LDL-ℓ_2, LDL-J, and LDL-KL in terms of 0/1 loss and ranking loss.

To show whether L_1-norm loss can truly bring better classification performance than other loss functions, we conduct the Wilcoxon signed-rank tests [47] for *LDL-ℓ_1* against each of the variants, which are summarized in Table 6 (from the 2nd column to the 4th column). Table 6 shows that, at significance of 0.05, *LDL-ℓ_1* achieves superior or at least comparable performance against each of the variants, which validates the advantage of L_1-norm loss. The reasons may lie in that expected L_1-norm loss bounds the error probability (Theorem 2), and L_1-norm loss is tighter than other loss functions for $\|\boldsymbol{p} - \boldsymbol{q}\|_1 \leq 2\sqrt{D_{\mathrm{KL}}(\boldsymbol{p}||\boldsymbol{q})}$ [35], $\|\boldsymbol{p} - \boldsymbol{q}\|_1 \leq 2\sqrt{D_{\mathrm{J}}(\boldsymbol{p}||\boldsymbol{q})}^5$, and $\|\boldsymbol{p} - \boldsymbol{q}\|_1 \leq \sqrt{m}\|\boldsymbol{p} - \boldsymbol{q}\|_2$, for label distribution $\boldsymbol{p}, \boldsymbol{q} \in \mathbb{R}^m$, where $D_{\mathrm{KL}}(\cdot||\cdot)$ is the KL divergence, and $D_{\mathrm{J}}(\cdot||\cdot)$ is the Jeffrey's divergence.

5.4.2 Effectiveness of Re-weighting Schemes

Since the only difference between *RW-LDL* and *LDL-ℓ_1* is that *RW-LDL* incorporates the re-weighting schemes, we compare *RW-LDL* with *LDL-ℓ_1* to validate the usefulness of

5. Recall that $D_{\mathrm{J}}(\boldsymbol{p}||\boldsymbol{q}) = \sum_j (p_i - q_i)(\ln p_i - \ln q_i) = D_{\mathrm{KL}}(\boldsymbol{p}||\boldsymbol{q}) + D_{\mathrm{KL}}(\boldsymbol{q}||\boldsymbol{p}) \geq D_{\mathrm{KL}}(\boldsymbol{p}||\boldsymbol{q})$ because $D_{\mathrm{KL}}(\boldsymbol{q}||\boldsymbol{p}) \geq 0$ [35].

TABLE 4
MLL predictive performance (mean±std.(rank)) of each comparing algorithm on 12 selected datasets in terms of five MLL metrics.

Dataset	**Hamming loss ↓**								
	BFGS-LDL	LDL-SCL	StructRF	LDLFs	RSSR-LDL21	BR-wSVM	RELIAB	LIMO	RWLM-LDL
Alp	0.431±0.008(4)	0.434±0.008(6)	0.435±0.005(7)	0.429±0.010(3)	0.428±0.007(2)	0.451±0.010(9)	0.431±0.010(5)	0.436±0.009(8)	**0.424±0.008(1)**
Cdc	0.426±0.012(4)	0.431±0.014(6)	0.430±0.007(5)	0.426±0.009(3)	0.425±0.013(2)	0.438±0.008(8)	0.486±0.005(9)	0.431±0.013(7)	**0.417±0.010(1)**
Dia	0.318±0.015(4)	0.319±0.015(5)	0.320±0.008(6)	0.320±0.014(7)	0.317±0.012(3)	0.327±0.015(8)	0.474±0.005(9)	0.316±0.010(2)	**0.310±0.013(1)**
Elu	0.416±0.013(3)	0.419±0.009(6)	0.429±0.010(7)	0.416±0.013(4)	0.416±0.012(5)	0.452±0.011(9)	**0.412±0.012(1)**	0.433±0.007(8)	0.413±0.013(2)
Hea	0.442±0.024(6)	0.436±0.023(5)	**0.415±0.019(1)**	0.427±0.018(2)	0.443±0.024(7)	0.473±0.016(9)	0.433±0.015(4)	0.454±0.015(8)	0.432±0.014(3)
Spo	0.426±0.019(7)	0.424±0.021(6)	**0.411±0.015(1)**	0.420±0.011(5)	0.419±0.019(3)	0.437±0.020(8)	0.493±0.006(9)	0.419±0.020(4)	0.418±0.022(2)
Gen	0.448±0.002(8)	0.459±0.004(9)	0.438±0.002(5)	0.445±0.004(7)	0.444±0.002(6)	0.401±0.006(4)	0.322±0.002(2)	**0.320±0.003(1)**	0.323±0.002(3)
Nat	0.223±0.011(5)	0.273±0.008(8)	**0.205±0.013(1)**	0.316±0.010(9)	0.214±0.011(4)	0.240±0.007(6)	0.210±0.007(3)	0.259±0.028(7)	0.209±0.200(2)
SBU	0.412±0.012(8)	0.385±0.010(5)	0.346±0.013(2)	0.448±0.011(9)	0.368±0.014(4)	0.405±0.015(7)	0.365±0.011(3)	0.401±0.018(6)	**0.314±0.018(1)**
SJA	0.312±0.058(4)	0.442±0.058(8)	0.337±0.028(5)	0.469±0.048(9)	0.312±0.054(3)	0.355±0.038(6)	0.294±0.035(2)	0.408±0.033(7)	**0.288±0.051(1)**
RAF	0.195±0.008(9)	0.160±0.005(8)	0.151±0.007(6)	0.132±0.007(4)	0.142±0.006(5)	0.124±0.009(2)	0.154±0.010(7)	0.124±0.010(3)	**0.112±0.011(1)**
Med	0.428±0.000(6)	0.430±0.001(7)	0.433±0.000(9)	0.431±0.001(8)	0.427±0.000(5)	0.102±0.008(4)	0.053±0.000(3)	0.041±0.001(2)	**0.040±0.001(1)**
Dataset	**One error ↓**								
	BFGS-LDL	LDL-SCL	StructRF	LDLFs	RSSR-LDL21	BR-wSVM	RELIAB	LIMO	RWLM-LDL
Alp	0.338±0.026(4)	0.351±0.029(7)	0.395±0.014(9)	0.342±0.034(5)	0.329±0.024(3)	0.378±0.031(8)	0.312±0.023(2)	0.346±0.040(6)	**0.301±0.019(1)**
Cdc	0.324±0.029(3)	0.324±0.028(2)	0.336±0.029(8)	0.325±0.024(4)	0.325±0.031(5)	0.411±0.028(9)	0.335±0.024(7)	0.331±0.026(6)	**0.321±0.023(1)**
Dia	0.164±0.017(7)	0.157±0.020(6)	0.193±0.030(9)	0.176±0.025(8)	0.153±0.024(4)	0.157±0.028(5)	0.138±0.018(3)	0.135±0.016(2)	**0.132±0.017(1)**
Elu	0.335±0.036(4)	0.336±0.041(5)	0.325±0.027(2)	0.340±0.026(7)	0.330±0.029(3)	0.403±0.038(9)	0.336±0.019(6)	0.391±0.045(8)	**0.317±0.026(1)**
Hea	0.386±0.032(6)	0.389±0.037(7)	**0.363±0.024(1)**	0.381±0.029(4)	0.377±0.027(3)	0.475±0.026(9)	0.384±0.034(5)	0.417±0.043(8)	0.376±0.025(2)
Spo	0.390±0.026(5)	0.389±0.025(4)	0.388±0.029(2)	0.394±0.031(8)	0.389±0.027(3)	0.433±0.048(9)	0.394±0.032(7)	0.393±0.036(6)	**0.372±0.026(1)**
Gen	0.582±0.016(5)	0.604±0.028(6)	0.636±0.012(8)	0.576±0.018(4)	0.561±0.017(3)	0.674±0.029(9)	0.518±0.006(2)	0.615±0.019(7)	**0.512±0.007(1)**
Nat	0.428±0.028(5)	0.468±0.030(6)	0.357±0.041(2)	0.561±0.106(8)	0.393±0.027(4)	0.525±0.034(7)	0.364±0.039(3)	0.569±0.077(9)	**0.333±0.048(1)**
SBU	0.387±0.033(5)	0.324±0.025(3)	0.303±0.031(2)	0.477±0.051(8)	0.324±0.021(3)	0.463±0.034(7)	0.324±0.020(4)	0.459±0.035(6)	**0.261±0.033(1)**
SJA	0.212±0.085(2)	0.472±0.182(8)	0.268±0.058(5)	0.502±0.104(9)	0.231±0.088(3)	0.268±0.084(6)	0.234±0.069(4)	0.402±0.121(7)	**0.202±0.085(1)**
RAF	0.090±0.014(9)	0.046±0.010(2)	0.060±0.008(5)	0.062±0.009(6)	0.050±0.007(3)	0.057±0.012(4)	0.085±0.014(8)	0.062±0.013(7)	**0.043±0.009(1)**
Med	0.139±0.004(4)	0.163±0.003(5)	0.197±0.004(7)	0.186±0.015(6)	0.134±0.004(2)	0.329±0.074(9)	0.139±0.004(3)	0.210±0.043(8)	**0.134±0.006(1)**
Dataset	**Coverage ↓**								
	BFGS-LDL	LDL-SCL	StructRF	LDLFs	RSSR-LDL21	BR-wSVM	RELIAB	LIMO	RWLM-LDL
Alp	0.842±0.005(6)	0.843±0.006(7)	0.840±0.004(5)	0.839±0.005(3)	0.840±0.004(4)	0.869±0.007(9)	0.837±0.006(2)	0.849±0.006(8)	**0.835±0.005(1)**
Cdc	0.833±0.005(5)	0.834±0.006(6)	**0.825±0.007(1)**	0.828±0.006(4)	0.834±0.006(7)	0.838±0.006(8)	0.827±0.007(3)	0.845±0.011(9)	0.826±0.008(2)
Dia	0.634±0.012(6)	0.635±0.015(7)	0.628±0.013(4)	0.630±0.017(5)	0.638±0.012(9)	0.635±0.013(8)	0.620±0.007(2)	0.620±0.007(3)	**0.614±0.010(1)**
Elu	0.811±0.008(4)	0.812±0.008(5)	0.818±0.006(7)	**0.809±0.008(1)**	0.811±0.008(3)	0.837±0.006(9)	0.813±0.007(6)	0.825±0.012(8)	**0.809±0.007(1)**
Hea	0.661±0.011(6)	0.659±0.014(4)	**0.641±0.010(1)**	0.651±0.012(2)	0.662±0.014(7)	0.684±0.006(9)	0.660±0.011(5)	0.682±0.016(8)	0.659±0.012(3)
Spo	0.616±0.013(6)	0.615±0.013(5)	**0.604±0.011(1)**	0.614±0.010(4)	0.609±0.010(3)	0.627±0.013(8)	0.621±0.014(7)	0.634±0.029(9)	0.608±0.009(2)
Gen	0.914±0.002(6)	0.917±0.003(7)	0.905±0.002(2)	0.909±0.003(4)	0.912±0.002(5)	0.935±0.003(9)	0.906±0.002(3)	0.931±0.005(8)	**0.905±0.002(1)**
Nat	0.312±0.016(4)	0.338±0.015(6)	**0.280±0.022(1)**	0.381±0.023(8)	0.320±0.021(5)	0.359±0.008(7)	0.297±0.016(3)	0.382±0.022(9)	0.285±0.017(2)
SBU	0.589±0.015(8)	0.561±0.011(5)	0.531±0.012(3)	0.606±0.011(9)	0.549±0.016(4)	0.563±0.017(6)	0.516±0.017(2)	0.574±0.013(7)	**0.490±0.021(1)**
SJA	0.507±0.049(4)	0.792±0.084(9)	0.534±0.043(5)	0.670±0.040(8)	0.491±0.046(3)	0.563±0.035(6)	0.487±0.048(2)	0.599±0.040(7)	**0.480±0.057(1)**
RAF	0.297±0.009(9)	0.266±0.007(2)	0.287±0.008(8)	0.282±0.007(6)	0.273±0.008(3)	0.276±0.008(4)	0.286±0.008(7)	0.278±0.008(5)	**0.265±0.008(1)**
Med	0.174±0.002(4)	0.198±0.006(6)	0.207±0.003(7)	0.191±0.007(5)	0.170±0.002(3)	0.265±0.008(8)	0.169±0.002(2)	0.275±0.011(9)	**0.161±0.003(1)**
Dataset	**Ranking loss ↓**								
	BFGS-LDL	LDL-SCL	StructRF	LDLFs	RSSR-LDL21	BR-wSVM	RELIAB	LIMO	RWLM-LDL
Alp	0.398±0.009(5)	0.403±0.011(6)	0.404±0.005(7)	0.393±0.013(3)	0.393±0.009(4)	0.429±0.014(9)	0.391±0.010(2)	0.404±0.011(8)	**0.387±0.012(1)**
Cdc	0.406±0.014(6)	0.409±0.012(7)	0.405±0.012(5)	0.402±0.011(3)	0.405±0.014(4)	0.417±0.010(9)	0.397±0.017(2)	0.413±0.012(8)	**0.393±0.013(1)**
Dia	0.280±0.014(7)	0.281±0.017(8)	0.278±0.015(5)	0.277±0.015(4)	0.279±0.012(6)	0.285±0.013(9)	0.269±0.012(2)	0.272±0.015(3)	**0.266±0.010(1)**
Elu	0.387±0.013(3)	0.390±0.015(6)	0.395±0.011(7)	0.387±0.012(5)	0.386±0.014(2)	0.435±0.009(9)	0.387±0.012(4)	0.408±0.009(8)	**0.384±0.015(1)**
Hea	0.424±0.023(7)	0.417±0.021(5)	**0.390±0.019(1)**	0.406±0.020(2)	0.423±0.025(6)	0.462±0.021(9)	0.414±0.020(4)	0.438±0.019(8)	0.412±0.018(3)
Spo	0.404±0.024(6)	0.403±0.023(5)	**0.388±0.017(1)**	0.401±0.015(4)	0.399±0.022(3)	0.422±0.025(9)	0.408±0.025(7)	0.409±0.025(8)	0.397±0.021(2)
Gen	0.430±0.003(5)	0.443±0.006(7)	0.432±0.003(6)	0.422±0.005(3)	0.424±0.003(4)	0.478±0.006(9)	0.414±0.003(2)	0.469±0.006(8)	**0.413±0.004(1)**
Nat	0.195±0.010(4)	0.219±0.009(6)	0.163±0.014(2)	0.268±0.027(9)	0.197±0.014(5)	0.241±0.008(7)	0.174±0.011(3)	0.264±0.011(8)	**0.161±0.014(1)**
SBU	0.374±0.016(6)	0.340±0.014(5)	0.300±0.016(2)	0.419±0.013(9)	0.323±0.016(4)	0.396±0.021(7)	0.303±0.021(3)	0.401±0.022(8)	**0.264±0.020(1)**
SJA	0.260±0.057(4)	0.723±0.177(9)	0.283±0.040(5)	0.447±0.057(8)	0.244±0.059(3)	0.310±0.042(6)	0.240±0.059(2)	0.408±0.040(7)	**0.233±0.065(1)**
RAF	0.092±0.009(9)	0.060±0.007(2)	0.079±0.007(7)	0.075±0.007(6)	0.066±0.007(3)	0.069±0.008(4)	0.081±0.009(8)	0.072±0.006(5)	**0.059±0.007(1)**
Med	0.050±0.001(4)	0.057±0.002(6)	0.063±0.001(7)	0.056±0.003(5)	0.048±0.001(3)	0.093±0.003(9)	0.047±0.001(2)	0.085±0.003(8)	**0.044±0.001(1)**
Dataset	**Average precision ↑**								
	BFGS-LDL	LDL-SCL	StructRF	LDLFs	RSSR-LDL21	BR-wSVM	RELIAB	LIMO	RWLM-LDL
Alp	0.647±0.008(5)	0.642±0.010(7)	0.640±0.005(8)	0.651±0.013(4)	0.652±0.007(3)	0.628±0.011(9)	0.656±0.008(2)	0.644±0.011(6)	**0.660±0.010(1)**
Cdc	0.654±0.010(6)	0.653±0.009(7)	0.656±0.009(4)	0.658±0.007(3)	0.655±0.010(5)	0.638±0.008(9)	0.660±0.012(2)	0.650±0.009(8)	**0.662±0.010(1)**
Dia	0.811±0.010(7)	0.812±0.011(4)	0.808±0.013(9)	0.812±0.009(6)	0.812±0.008(5)	0.810±0.009(8)	0.819±0.010(2)	0.818±0.009(3)	**0.820±0.007(1)**
Elu	0.678±0.012(3)	0.676±0.013(6)	0.673±0.009(7)	0.677±0.008(5)	0.679±0.013(2)	0.638±0.010(9)	0.678±0.010(4)	0.656±0.008(8)	**0.681±0.013(1)**
Hea	0.699±0.014(7)	0.702±0.015(5)	**0.719±0.013(1)**	0.709±0.013(2)	0.701±0.015(6)	0.668±0.010(9)	0.702±0.015(4)	0.685±0.014(8)	0.705±0.013(3)
Spo	0.720±0.014(6)	0.721±0.013(5)	**0.728±0.010(1)**	0.721±0.012(4)	0.725±0.013(3)	0.701±0.019(9)	0.718±0.015(8)	0.718±0.018(7)	0.726±0.012(2)
Gen	0.411±0.004(5)	0.401±0.005(7)	0.402±0.004(6)	0.418±0.006(3)	0.417±0.005(4)	0.370±0.004(9)	0.431±0.004(2)	0.376±0.007(8)	**0.434±0.004(1)**
Nat	0.686±0.017(5)	0.655±0.016(6)	0.732±0.020(2)	0.591±0.050(7)	0.701±0.016(4)	0.587±0.015(9)	0.720±0.022(3)	0.587±0.017(8)	**0.736±0.025(1)**
SBU	0.692±0.012(6)	0.722±0.010(5)	0.750±0.013(2)	0.651±0.011(9)	0.733±0.010(4)	0.674±0.017(7)	0.749±0.014(3)	0.673±0.016(8)	**0.780±0.016(1)**
SJA	0.805±0.035(3)	0.587±0.071(9)	0.775±0.031(5)	0.643±0.038(8)	0.814±0.042(2)	0.745±0.040(6)	0.804±0.039(4)	0.678±0.038(7)	**0.812±0.050(1)**
RAF	0.905±0.009(9)	0.938±0.007(2)	0.922±0.006(7)	0.925±0.006(6)	0.933±0.006(3)	0.930±0.007(4)	0.914±0.008(8)	0.926±0.006(5)	**0.940±0.006(1)**
Med	0.707±0.004(4)	0.684±0.004(5)	0.644±0.003(7)	0.666±0.011(6)	0.718±0.004(2)	0.521±0.029(9)	0.718±0.004(3)	0.585±0.020(8)	**0.724±0.005(1)**

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2021.3082623, IEEE Transactions on Pattern Analysis and Machine Intelligence

TABLE 5
Results of the Friedman test. The Friedman statistics in terms of each metric and the critical value at 0.05 significance level are given.

MLL Metrics	Friedman Statistics	Critical Value
Hamming loss	30.7778	
One error	49.4288	
Coverage	47.9166	2.0454
Ranking loss	53.1778	
Avg. precision	51.0444	

the re-weighting schemes. Following the same evaluation protocol with Section 5.4.1, we evaluate the performance of *RW-LDL* and *LDL-*ℓ_1. To show whether the re-weighting schemes yield better classification performance, we conduct the Wilcoxon signed-rank tests [47] for *RW-LDL* against *LDL-*ℓ_1, which are reported in Table 6 (the 5th column). Table 6 shows that *RW-LDL* is superior or at least comparable to *LDL-*ℓ_1 in terms of all metrics, which justifies the usefulness of the re-weighting schemes.

5.4.3 Effectiveness of Large Margin

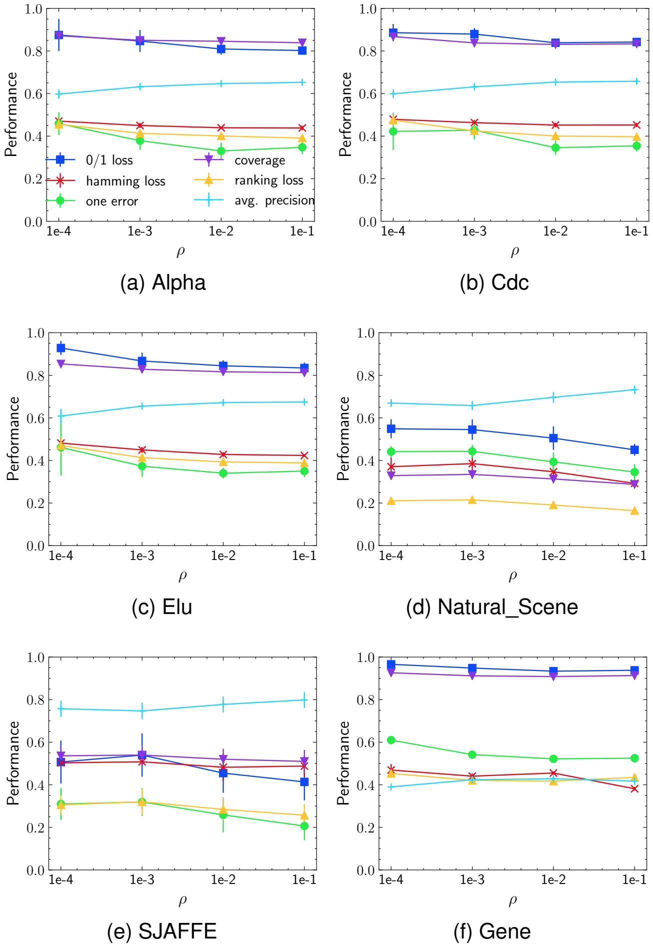
RWLM-LDL (*LM-LDL*) is only different from *RW-LDL* (*LDL-*ℓ_1) in that large margin is used in *RWLM-LDL* (*LM-LDL*). We compare *RWLM-LDL* (*LM-LDL*) against *RW-LDL* (*LDL-*ℓ_1) to show the effectiveness of large margin. Similar to Section 5.4.1, we evaluate the performance of these variants. To investigate whether large margin can truly improve classification performance, we conduct the Wilcoxon signed-rank tests [47] for *RWLM-LDL* against *RW-LDL* and *LM-LDL* against *LDL-*ℓ_1, which are summarized in Table 6 (the last two columns). Table 6 shows that *RWLM-LDL* has statistically superior performance against *RW-LDL*, and *LM-LDL* significantly outperforms *LDL-*ℓ_1, which demonstrates the effectiveness of large margin.

5.4.4 Advantage of Label Distribution

Label distribution can represent multi-label. For an instance with c positive labels, we can represent it as a label distribution: a degree of $1/c$ for each positive label and 0 for each negative label (*e.g.*, $\{1, 0, 1, 0, 0, 0\}$ can be equivalently rewritten as $\{0.5, 0, 0.5, 0, 0, 0\}$). We can run LDL methods on multi-label as follows: represent multi-label as label distribution and run LDL methods. Notice that *RAF_ML* has both the ground-truth label distribution and mulit-label. To show the advantage of label distribution, we run *BFGS-LDL*, *LDLFs*, and *RWLM-LDL* on the label distribution and multi-label of *RAF_MLL*. The results are shown in Table 7, where *BFGS-ML*, *LDLFs-ML*, and *RWLM-ML* denote *BFGS-LDL*, *LDLFs*, and *RWLM-LDL* on multi-label, respectively. Table 7 shows that the methods on label distribution have better performance than those on multi-label, which manifests the advantage of label distribution. The reasons lie in that label distribution has more information than 0/1 label and directly models the relative importance of labels.

5.5 Further Analysis

5.5.1 Parameter Sensitivity Analysis



(a) Alpha (b) Cdc

(c) Elu (d) Natural_Scene

(e) SJAFFE (f) Gene

Fig. 8. Influence of ρ in terms of 0/1 loss and five MLL metrics.

This subsection analyzes the influence of the parameters, including λ_1 (the regularization parameter), λ_2 (the trade-off parameter), and ρ (the margin).

To study the influence of ρ, we run *RWLM-LDL* with ρ from the candidate set $\{10^{-4}, \cdots, 10^{-1}\}$. Fig. 8 presents the results on several datasets. We can see from Fig. 8 that *RWLM-LDL* achieves satisfying performance with $\rho = 0.1$. Additionally, to investigate the influence of λ_1 and λ_2, we set $\rho = 0.1$ and run *RWLM-LDL* with λ_1 and λ_2 selecting from the candidate set $\{10^{-5}, \cdots, 10^{5}\}$. Fig. 9 presents the results of the grid-search for λ_1 and λ_2 on *SJAFFE* in terms of each metric. We can see from Fig. 9 that *RWLM-LDL* with $\lambda_1 = 0.0001$ has a satisfying performance. Moreover, *RWLM-LDL* is robust *w.r.t.* λ_2, which can be simply set to 1.

5.5.2 Convergence

To study the convergence of *RWLM-LDL*, Fig. 10 plots the objective function values *w.r.t.* the number of iterations on *Alpha* for SLL and MLL. As can be seen from Fig. 10, *RWLM-LDL* converges fast, and the objective function approaches a stable value after about 200 iterations, which validates the efficiency of the optimization method.

5.5.3 When Does RWLM-LDL Work Well?

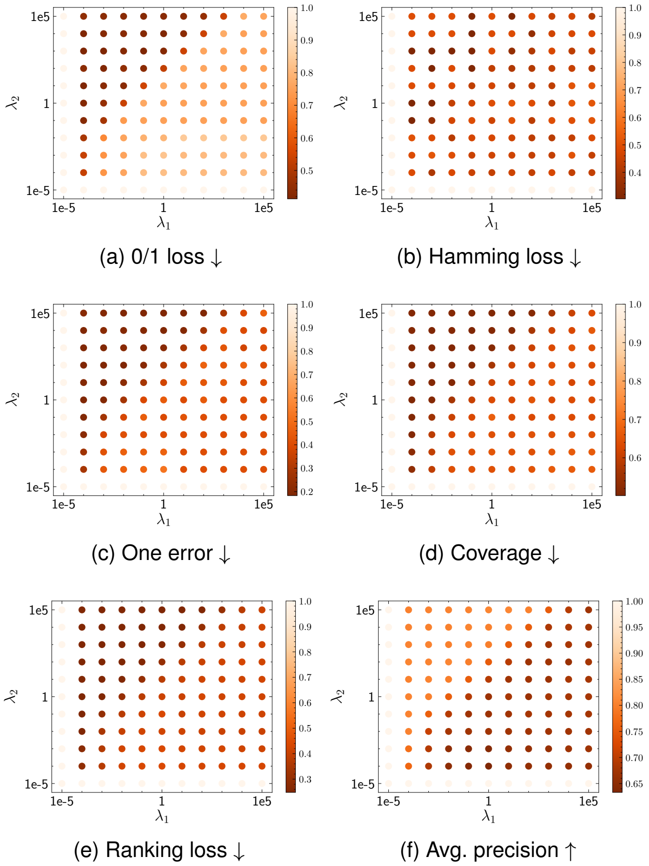
For each dataset, define $r_1 = \max_i E_{\boldsymbol{x}_i} / \min_i E_{\boldsymbol{x}_i}$ that is the ratio between the maximum and the minimum entropy of the label distributions, and define $r_2 = \text{avg. Ent.}/m \ln m$

TABLE 6
Results (win/tie/lose[p-value]) of the Wilcoxon signed-rank test in terms of 0/1 loss and five MLL metrics (at 0.05 confidence level).

Metric	LDL-ℓ_1 **against**			RW-LDL **against**	RWLM-LDL **against**	LM-LDL **against**
	LDL-ℓ_2	LDL-J	LDL-KL	LDL-ℓ_1	RW-LDL	LDL-ℓ_1
0/1 loss	**tie**[1.20e-1]	**win**[6.13e-3]	**win**[4.94e-2]	**win**[3.87e-2]	**win**[4.45e-3]	**win**[2.71e-3]
Hamming loss	**win**[1.50e-2]	**tie**[5.97e-2]	**win**[2.29e-2]	**win**[2.81e-2]	**win**[6.04e-3]	**win**[7.65e-3]
One error	**tie**[9.26e-2]	**tie**[5.15e-1]	**tie**[2.03e-1]	**tie**[5.96e-2]	**win**[9.63e-3]	**tie**[8.44e-2]
Coverage	**win**[4.14e-2]	**tie**[1.36e-1]	**tie**[1.58e-1]	**win**[2.29e-2]	**win**[2.22e-3]	**win**[2.29e-2]
Ranking loss	**win**[7.65e-3]	**win**[2.81e-2]	**win**[4.99e-2]	**win**[2.22e-3]	**win**[2.22e-3]	**win**[2.87e-3]
Average precision	**win**[6.04e-3]	**win**[1.21e-2]	**win**[1.21e-2]	**tie**[7.12e-2]	**win**[2.22e-3]	**win**[6.04e-3]

TABLE 7
Performance (mean±std) comparison for the algorithms learning the label distribution and multi-label on *RAF_ML*.

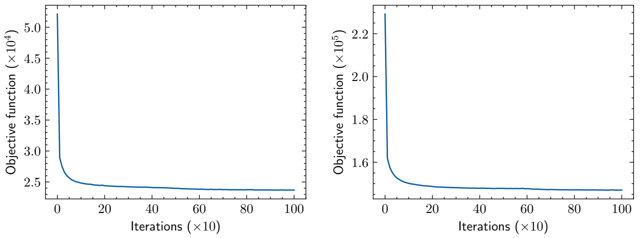
Method	Hamming loss	One error	Coverage	Ranking loss	Average precision
BFGS-ML	0.228±0.006	0.116±0.015	0.300±0.008	0.099±0.008	0.895±0.008
BFGS-LDL	**0.195**±**0.008**	**0.090**±**0.014**	**0.297**±**0.009**	**0.092**±**0.009**	**0.905**±**0.009**
LDLFs-ML	**0.124**±**0.010**	0.071±0.013	0.286±0.010	0.079±0.007	0.921±0.006
LDLFs	0.132±0.00	**0.062**±**0.009**	**0.282**±**0.007**	**0.075**±**0.007**	**0.933**±**0.006**
RWLM-ML	0.139±0.008	0.051±0.010	0.269±0.007	0.063±0.007	0.935±0.007
RWLM-LDL	**0.112**±**0.011**	**0.043**±**0.009**	**0.265**±**0.008**	**0.059**±**0.007**	**0.940**±**0.006**



(a) 0/1 loss ↓ (b) Hamming loss ↓ (c) One error ↓ (d) Coverage ↓ (e) Ranking loss ↓ (f) Avg. precision ↑

Fig. 9. Influence of λ_1 and λ_2 in terms of each metric on *SJAFFE*.

that is the ratio between the average entropy of the label distributions and the possible maximum entropy (*i.e.*, uniform distribution). Note that r_1 indicates how large the gap among the entropy of the label distributions is, and r_2 reveals how evenly the label distributions are distributed ($r_2 = 1$ means uniform distribution). The statistics of r_1 and r_2 for each dataset are listed in Table 2.



(a) SLL case (b) MLL case

Fig. 10. Convergence of *RWLM-LDL* on *Alpha* in SLL and MLL cases.

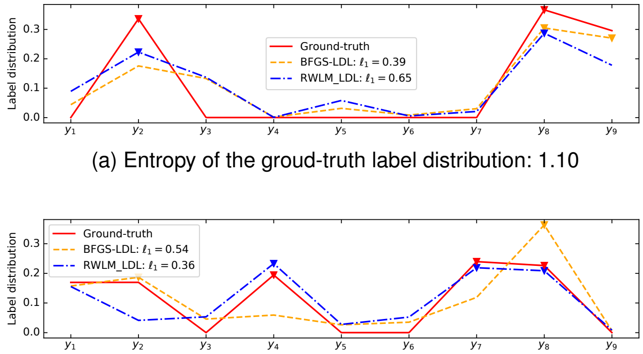
According to Tables 3 and 4, we observe that *RWLM-LDL* works less efficiently on the datasets with small r_1 and large r_2, such as *Heat* and *Spo*. The usefulness of the re-weighting schemes is partially suppressed when the gap among the entropy of the label distributions is small (small r_1). The effectiveness of large margin is suppressed when the label distributions are overall evenly-distributed (large r_2) since there will be more points inside the marginal hyperplanes, as discussed in Section 3.3.4. Instead, *RWLM-LDL* achieves much better performance on the datasets with large r_1 and small r_2, such as *Spoem* and *Gene*, because the re-weighting schemes and large margin work more efficiently.

To summarize, *RWLM-LDL* works well particularly on the datasets where the gap among the entropy of the label distributions is large, and the label distributions are unevenly-distributed.

5.5.4 Why Does RWLM-LDL Work Well?

Fig. 11 shows two examples from *Natural_Scene* in MLL, where the ground-truth label distributions, as well as the ones predicted by *BFGS-LDL* and *RWLM-LDL*, are presented. From Fig. 11, two observations can be made:

1) The ground-truth label distribution of Fig. 11b has higher entropy than that of Fig. 11a. *RWLM-LDL*



(a) Entropy of the groud-truth label distribution: 1.10

(b) Entropy of the ground-truth label distribution: 1.65

Fig. 11. Examples from *Natural_Scene*. The ground-truth label distribution of Fig. 11b has higher entropy, which implies that the objective inconsistency is more likely to occur. RWLM-LDL captures the entropy and has a smaller loss for Fig. 11b. Besides, RWLM-LDL mainly focuses on the top labels.

captures the entropy and has a smaller L_1-norm loss for Fig. 11b. In contrast, *BFGS-LDL* neglects the entropy and has a smaller L_1-norm loss for Fig. 11a.

2) *RWLM-LDL* mainly focuses on the label description degrees of the top labels and successfully keeps the rankings of the top labels. In contrast, *BFGS-LDL* neglects the rankings of the top labels for the sake of learning the whole label distributions.

According to the preceding observations, label distribution has rich information, which at least includes the relative importance of labels, the rankings of labels, and the uniformity of the label distribution (by the entropy). Traditional LDL methods only consider the relative importance of labels and ignore others. In contrast, *RWLM-LDL* captures the uniformity of label distributions by re-weighting *w.r.t.* the entropy of label distributions and keeps the rankings of the top labels by large margin and re-weighting *w.r.t.* label description degrees. To summarize, re-weighting and large margin are helpful to sufficiently exploit the rich information of label distribution, which explains why *RWLM-LDL* works well.

6 Conclusion

Although LDL has been applied to varieties of real classification tasks, it faces the challenge of objective inconsistency, which leads to the performance deterioration of LDL.

This paper addresses the inconsistency. We establish the relation between LDL and classification that the expected L_1-norm loss of LDL bounds the classification error probability. We then propose a new LDL method named *RWLM-LDL* that employs three components, including L_1-norm loss, re-weighting schemes, and large margin. *RWLM-LDL* is shown to have generalization and discrimination. In the experiments, we show that *RWLM-LDL* has a competitive performance against the state-of-the-art LDL methods and SLL/MLL methods. Furthermore, ablation study and further analysis explain the effectiveness of *RWLM-LDL*.

In the future study, we will explore the followings:

1) How to apply the re-weighing schemes to the relation between LDL and classification and establish a tighter bound.
2) How to leverage the large margin to derive a tighter bound on the error probability, and how to leverage the re-weighting schemes to derive a tighter bound on the expected 0/1 loss.
3) Replace the maximum entropy model with a deep model, and use AutoML techniques to train a high-quality model automatically.

Acknowledgments

This research was supported by the National Key Research and Development Plan of China (No. 2017YFB1002801), the National Science Foundation of China (62076063, 62076062), the Collaborative Innovation Center of Novel Software Technology and Industrialization, and the Collaborative Innovation Center of Wireless Communications Technology.

References

[1] B. Gao, C. Xing, C. Xie, J. Wu, and X. Geng, “Deep label distribution learning with label ambiguity,” *IEEE Trans. Image Process.*, vol. 26, no. 6, pp. 2825–2838, June 2017.

[2] M. Zhang and Z. Zhou, “ML-KNN: A lazy learning approach to multi-label learning,” *Pattern Recognit.*, vol. 40, no. 7, pp. 2038 – 2048, July 2007.

[3] M. Lyons, S. Akamatsu, M. Kamachi, and J. Gyoba, “Coding facial expressions with gabor wavelets,” in *Proc. 3th IEEE Int. Conf. Auto. Face Gesture Recogn.*, Apr. 1998, pp. 200–205.

[4] M. Zhang and Z. Zhou, “A review on multi-label learning algorithms,” *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 8, pp. 1819–1837, Aug. 2014.

[5] X. Geng, “Label distribution learning,” *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 7, pp. 1734–1748, July 2016.

[6] X. Geng and L. Luo, “Multilabel ranking with inconsistent rankers,” in *Proc. 2014 IEEE Conf. Comput. Vis. Pattern Recogn. (CVPR’14)*, June 2014, pp. 3742–3747.

[7] X. Geng, C. Yin, and Z. Zhou, “Facial age estimation by learning from label distributions,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 10, pp. 2401–2412, Oct. 2013.

[8] X. Wen, B. Li, H. Guo, Z. Liu, G. Hu, M. Tang, and J. Wang, “Adaptive variance based label distribution learning for facial age estimation,” in *Proc. 16th Eur. Conf. Comput. Vis. (ECCV’20)*, Aug. 2020, pp. 379–395.

[9] X. Geng, X. Qian, Z. Huo, and Y. Zhang, “Head pose estimation based on multivariate label distribution,” *IEEE Trans. Pattern Anal. Mach. Intell.*, 2020.

[10] S. Li and W. Deng, “Blended emotion in-the-wild: Multi-label facial expression recognition using crowdsourced annotations and deep locality feature learning,” *Int. J. Comput. Vis.*, vol. 127, no. 6, pp. 884–906, June 2019.

[11] L. Liang, L. Lin, L. Jin, D. Xie, and M. Li, “SCUT-FBP5500: A diverse benchmark dataset for multi-paradigm facial beauty prediction,” in *Proc. 24th Int. Conf. Pattern Recogn.*, Aug. 2018, pp. 1598–1603.

[12] X. Wu, N. Wen, J. Liang, Y. Lai, D. She, M. Cheng, and J. Yang, “Joint acne image grading and counting via label distribution learning,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV’19)*, Oct. 2019, pp. 10 641–10 650.

[13] M. Zhang, Q. Zhang, J. Fang, Y. Li, and X. Geng, “Leveraging implicit relative labeling-importance information for effective multi-label learning,” *IEEE Trans. Knowl. Data Eng.*, vol. 33, no. 5, pp. 2057–2070, May 2021.

[14] B. Gao, H. Zhou, J. Wu, and X. Geng, “Age estimation using expectation of label distribution learning,” in *Proc. 27th Int. Joint Conf. Artif. Intell. (IJCAI’18)*, July 2018, pp. 712–718.

[15] J. Wang and X. Geng, “Classification with label distribution learning,” in *Proc. 28th Int. Joint Conf. Artif. Intell. (IJCAI’19)*, July 2019, pp. 3712–3718.

[16] W. Shen, K. Zhao, Y. Guo, and A. L. Yuille, “Label distribution learning forests,” in *Proc. 31st Int. Conf. Neural Inf. Process. Syst. (NIPS)’17*, Dec. 2017, pp. 834–843.

[17] M. Chen, X. Wang, B. Feng, and W. Liu, "Structured random forest for label distribution learning," *Neurocomputing*, vol. 320, pp. 171–182, Dec. 2018.

[18] W. Yang, C. Li, and H. Zhao, "Label distribution learning by regularized sample self-representation," *Math. Probl. Eng.*, vol. 2018, Apr. 2018.

[19] X. Jia, Z. Li, X. Zheng, W. Li, and S. Huang, "Label distribution learning with label correlations on local samples," *IEEE Trans. Knowl. Data Eng.*, vol. 33, no. 4, pp. 1619–1631, Apr. 2021.

[20] J. Wang and X. Geng, "Theoretical analysis of label distribution learning," in *Proc. 33th AAAI Conf. Artif. Intell. (AAAI'19)*, Feb. 2019, pp. 5256–5263.

[21] V. Vapnik and A. Y. Chervonenkis, "A note on one class of perceptrons," *Automat. Rem. Control*, vol. 25, no. 1, pp. 821–837, 1964.

[22] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, Sep. 1995.

[23] C. Hsu and C. Lin, "A comparison of methods for multiclass support vector machines," *IEEE Trans. Neural Netw.*, vol. 13, no. 2, pp. 415–425, Apr. 2002.

[24] T. Zhang and Z. Zhou, "Optimal margin distribution machine," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 6, pp. 1143–1156, June 2020.

[25] A. Elisseeff and J. Weston, "A kernel method for multi-labelled classification," in *Proc. 14th Int. Conf. Neural Inf. Process. Syst. (NIPS'01)*, Jan. 2001, pp. 681–687.

[26] X. Wu and Z. Zhou, "A unified view of multi-label performance measures," in *Proc. Int. Conf. Mach. Learn. (ICML'17)*, Aug. 2017, pp. 3780–3788.

[27] M. Mohri, A. Rostamizadeh, and A. Talwalkar, *Foundations of Machine Learning*. The MIT Press, 2012.

[28] H. Kahn and A. W. Marshall, "Methods of reducing sample size in monte carlo computations," *J. Oper. Res. Soc.*, vol. 1, no. 5, pp. 263–278, 1953.

[29] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *J. Artif. Intell. Res.*, vol. 16, pp. 321–357, June 2002.

[30] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *J. Comput. Syst. SCI.*, vol. 55, no. 1, pp. 119–139, Aug. 1997.

[31] T. Hastie, S. Rosset, J. Zhu, and H. Zou, "Multi-class AdaBoost," *Stat. Interface*, vol. 2, no. 3, pp. 349–360, Jan. 2009.

[32] L. Devroye, L. Györfi, and G. Lugosi, *A probabilistic theory of pattern recognition*. Springer Science & Business Media, 1996, vol. 31.

[33] W. Gao and Z. Zhou, "On the consistency of multi-label learning," *Artif. Intell*, vol. 199-200, pp. 22–44, June 2013.

[34] A. L. Berger, S. A. Della Pietra, and V. J. Della Pietra, "A maximum entropy approach to natural language processing," *Comput. Linguist.*, vol. 22, no. 1, pp. 39–71, Mar. 1996.

[35] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. NY, USA: Wiley-Interscience, 2006.

[36] L. Bottou, F. E. Curtis, and J. Nocedal, "Optimization methods for large-scale machine learning," *SIAM Rev.*, vol. 60, no. 2, pp. 223–311, May 2018.

[37] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Repr. (ICLR'15)*, May 2015.

[38] P. L. Bartlett and S. Mendelson, "Rademacher and gaussian complexities: Risk bounds and structural results," *J. Mach. Learn. Res.*, vol. 3, pp. 463–482, Mar. 2003.

[39] M. Eisen, P. Spellman, P. Brown, and D. Botstein, "Cluster analysis and display of genome-wide expression patterns," in *Proc. Nat. Acad. Sci. U.S.A.*, vol. 95, Dec. 1998, pp. 14 863–14 868.

[40] J. Yu, D. Jiang, K. Xiao, Y. Jin, J. Wang, and X. Sun, "Discriminate the falsely predicted protein-coding genes in aeropyrum pernix k1 genome based on graphical representation," *MATCH Commun. Math. Comput. Chem.*, vol. 67, no. 3, pp. 845–866, Jan. 2012.

[41] L. Yin, X. Wei, Y. Sun, J. Wang, and M. J. Rosato, "A 3D facial expression database for facial behavior research," in *Proc. 7th Int. Conf. Autom. Face Gesture Recogn.*, Apr. 2006, pp. 211–216.

[42] X. Geng and P. Hou, "Pre-release prediction of crowd opinion on movies by label distribution learning," in *Proc. 24th Int. Conf. Artif. Intell. (IJCAI'15)*, July 2015, pp. 3511–3517.

[43] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. 2016 IEEE Conf. Comput. Vis. Pattern Recogn. (CVPR'16)*, June 2016, pp. 770–778.

[44] C. G. M. Snoek, M. Worring, J. C. van Gemert, J. M. Geusebroek, and A. W. M. Smeulders, "The challenge problem for automated detection of 101 semantic concepts in multimedia," in *Proc. 14th ACM Int. Conf. Multimedia (MM'06)*, Oct. 2006, pp. 421–430.

[45] N. Xu, Y. Liu, and X. Geng, "Label enhancement for label distribution learning," *IEEE Trans. Knowl. Data Eng.*, vol. 33, no. 4, pp. 1632–1643, Apr. 2021.

[46] C. Chang and C. Lin, "Libsvm: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, pp. 1–27, May 2011.

[47] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *J. Mach. Learn. Res.*, vol. 7, pp. 1–30, Jan. 2006.

[48] O. J. Dunn, "Multiple comparisons among means," *J. AM. STAT. ASSOC.*, vol. 56, no. 293, pp. 52–64, Mar. 1961.

Jing Wang is currently pursuing Ph.D. degree with the School of Computer Science and Engineering, Southeast University. He received the B.Sc. degree in Computer Science from Suzhou University of Science and Technology, China, in 2013, and the M.Sc. degree in Computer Science from the Northeastern University, China, in 2015. His research interests include pattern recognition and machine learning.

Xin Geng (M'13-SM'21) is currently a professor and the dean of School of Computer Science and Engineering at Southeast University, China. He received the B.Sc. (2001) and M.Sc. (2004) degrees in computer science from Nanjing University, China, and the Ph.D. (2008) degree in computer science from Deakin University, Australia. His research interests include machine learning, pattern recognition, and computer vision. He has published over 70 refereed papers in these areas, including those published in prestigious journals and top international conferences. He has been an Associate Editor of IEEE T-MM, FCS and MFC, a Steering Committee Member of PRICAI, a Program Committee Chair for conferences such as PRICAI' 18, VALSE'13, etc., an Area Chair for conferences such as CVPR, ACMMM, PRCV, CCPR, and a Senior Program Committee Member for conferences such as IJCAI, AAAI, ECAI, etc. He is a Distinguished Fellow of IETI and a Member of IEEE.

Hui Xue received the B.Sc. degree in Mathematics from Nanjing Norm University in 2002. In 2005, she received the M.Sc. degree in Mathematics from Nanjing University of Aeronautics & Astronautics (NUAA). And she also received the Ph.D. degree in Computer Application Technology at NUAA in 2008. Since 2009, as an Associate Professor, she has been with the school of Computer Science and Engineering at Southeast University. Her research interests include pattern recognition and machine learning.