Classification with Label Distribution Learning

Jing Wang and **Xin Geng** *

MOE Key Laboratory of Computer Network and Information Integration
School of Computer Science and Engineering, Southeast University, Nanjing 210096, China
{wangjing91, xgeng}@seu.edu.cn

Abstract

Label Distribution Learning (LDL) is a novel learning paradigm, aim of which is to minimize the distance between the model output and the ground-truth label distribution. We notice that, in real-word applications, the learned label distribution model is generally treated as a classification model, with the label corresponding to the highest model output as the predicted label, which unfortunately prompts an inconsistency between the training phrase and the test phrase. To solve the inconsistency, we propose in this paper a new Label Distribution Learning algorithm for Classification (LDL4C). Firstly, instead of KL-divergence, absolute loss is applied as the measure for LDL4C. Secondly, samples are re-weighted with information entropy. Thirdly, large margin classifier is adapted to boost discrimination precision. We then reveal that theoretically LDL4C seeks a balance between generalization and discrimination. Finally, we compare LDL4C with existing LDL algorithms on 17 real-word datasets, and experimental results demonstrate the effectiveness of LDL4C in classification.

1 Introduction

Learning with ambiguity, especially label ambiguity [Gao *et al.*, 2017], has become one of the hot topics among the machine learning communities. Traditional supervised learning paradigms include single-label learning (SLL) and multi-label learning (MLL) [Zhang and Zhou, 2014], which label each instance with one or multiple labels. MLL assumes that each instance is associated with multiple labels, which compared with SLL takes the label ambiguity into consideration somewhat. Essentially, both SLL and MLL consider the relation between instance and label to be binary, i.e., whether or not the label is relevant with the instance. However, there are a variety of real-world tasks that instances are involved with labels with different importance degree, e.g., image annotation [Zhou and Zhang, 2006], emotion recognition [Zhou *et al.*, 2015], age estimation [Geng *et al.*, 2013]. Consequently, a soft label instead of a hard one seems to be a reasonable solution. Inspired by this, recently a novel learning paradigm, called Label Distribution Learning (LDL) [Geng, 2016], is proposed. LDL tackles label ambiguity with the definition of label description degree. Formally speaking, given an instance $\mathbf{x}$, LDL assigns each $y \in \mathcal{Y}$ a real value $d_{\mathbf{x}}^{y}$ (label description degree), which indicates the importance of y to $\mathbf{x}$. To make the definition proper, [Geng, 2016] suggests that $d_{\mathbf{x}}^{y} \in [0, 1]$ and $\sum_{y \in \mathcal{Y}} d_{\mathbf{x}}^{y} = 1$, and the real value function d is called the label distribution function. Since LDL extends the supervision from binary to label distribution, which is more applicable for real-world scenarios.

Due to the utility of dealing with ambiguity explicitly, LDL has been extensively applied in varieties of real-world problems. According to the source of label distribution, applications can be mainly classified into three classes. The first one includes emotion recognition [Zhou *et al.*, 2015], pre-release rating prediction on movies [Geng and Hou, 2015], et. al, where the label distribution is from the data. Applications of the second class include head pose estimation [Huo and Geng, 2017], crowd counting [Zhang *et al.*, 2015], et. al, label distribution of which are generated by pre-knowledge. Representative applications of the third one include beauty sensing [Ren and Geng, 2017], label enhancement [Xu *et al.*, 2018], where the label distribution is learned from the data. Notice that the aforementioned applications of LDL fall into the scope of classification, and we find that a learned LDL model is generally treated as a classification model, with the label corresponding to the highest model output as the prediction, which unfortunately draws an inconsistency between the aim of the training phrase and the goal of test phrase of LDL. In the training phrase, the aim of LDL is to minimize the distance between the model output and the ground-truth label distribution, while in the test phrase, the object is to minimize the 0/1 error.

We tackle the inconsistency in this paper. In order to alleviate the inconsistency, we come up with three improvements, i.e., absolute loss, re-weighting with entropy information, and large margin. The proposal of applying absolute loss and introducing large margin are inspired by theory arguments, and re-weighting samples with information entropy is based on observing the gap between the metric of LDL and that of the corresponding classification model. Since improvements are mainly originated from well-established

*Corresponding author.

theory tools, we are able to provide theoretical analysis of the proposed method. Consequently, the proposed algorithm LDL4C is well-designed and theoretically sound.

The main contributions of this paper are summarized as followings,

1) We propose three key components for improving classification performance for LDL, i.e, absolute loss, reweighting samples with information entropy, and large margin.
2) We establish upper bounds for 0/1 error and error probability of the proposed algorithms. Theoretical and empirical studies show that LDL4C enjoys generalization and discrimination.

The rest of the paper is organized as follows. Firstly, related works are briefly reviewed. Secondly details of the proposed method are presented. Then we give the theoretical analysis of the proposed method. Next experimental results are reported. Finally, we conclude the paper.

2 Related Work

Existing studies on LDL are primarily concentrated on designing new algorithms for LDL, and many algorithms have already been designed for LDL. [Geng, 2016] suggests three strategies for designing LDL algorithms. The first one is Problem Transformation (PT), which generates SL dataset according to the label distribution and then learns the transformed dataset with SL algorithms. Algorithms of the first strategy include PT-SVM and PT-Bayes, which apply SVM and Bayes classifier respectively. The second one is Algorithm Adaptation (AA), algorithms of which adapt existing learning algorithms to deal with label distribution straightly. Two representative algorithms are AA-kNN and AA-BP. For AA-kNN, mean of label distribution of k nearest neighbors is calculated as the predicted label distribution, and for AA-BP, one-hidden-layer neural network with multi-output is adapted to minimize the sum-squared loss of output of the network compared with the ground-truth label distribution. The last one is Specialized Algorithm (SA). This category of algorithms take characteristics of LDL into consideration. Two representative approaches of SA are IIS-LDL and BFGS-LDL, which apply the maximum entropy model to learn the label distribution. Besides, [Geng and Hou, 2015] treats LDL as a regression problem, and proposes LDL-SVR, which embraces SVR to deal with the label distribution. Furthermore, [Shen *et al.*, 2017] proposes LDLF, which extends random forest to learn label distribution. In addition, [Gao *et al.*, 2017] provides the first deep LDL model DLDL. Notice that compared with the classic LDL algorithms, LDLF and DLDL support end-to-end training, which are suitable for computer vision tasks. Notice that none of aforementioned algorithms ever consider the inconsistency as we previously discussed.

There are few work on inconsistency between the training and the test phrase of LDL. [Gao *et al.*, 2018] firstly recognizes the inconsistency in the application of age estimation, and designs a lightweight network to jointly learn the age distribution and the ground-truth age to bridge the gap. However the method is only suitable for real-valued label space and no theory guarantee is provided. Besides, one recent work [Wang and Geng, 2019] provides a theoretical analysis of the relation between the risk of LDL (absolute loss) and error probability of the corresponding classification function, discovering that LDL with absolute loss dominates classification. However, the major motivation of the paper is to understanding LDL theoretically, and no algorithm design to improve classification precision is provided.

3 The Proposed Method

3.1 Preliminary

Let $\mathcal{X}$ be the input space, and $\mathcal{Y} = \{y_1, y_2, \ldots, y_m\}$ be the label space. Denote the instance by $\mathbf{x}$. The label distribution function $d \in \mathbb{R}^{\mathcal{X}\times\mathcal{Y}}$ is defined as $d : \mathcal{X} \times \mathcal{Y} \mapsto \mathbb{R}$, satisfying the constraints $d_{\mathbf{x}}^y > 0$ and $\sum_{y\in\mathcal{Y}} d_{\mathbf{x}}^y = 1$, where $d_{\mathbf{x}}^y = d(\mathbf{x}, y)$ for convenience of notations. Given a training set $S = \{(\mathbf{x}_1, [d_{\mathbf{x}_1}^{y_1}, \ldots, d_{\mathbf{x}_1}^{y_m}]), \ldots, (\mathbf{x}_n, [d_{\mathbf{x}_n}^{y_1}, \ldots, d_{\mathbf{x}_n}^{y_m}])\}$, the goal of LDL is to learn the unknown function d, and the object of LDL4C is to make a prediction.

3.2 Classification via LDL

In real-world applications we generally regard a learned LDL model as a classification mode. Formally speaking, denote the learned LDL function by h and the corresponding classification function by $\hat{h}$, for a given instance $\mathbf{x}$, then we have

$$\hat{h}(\mathbf{x}) = \arg\max_{y\in\mathcal{Y}} h_{\mathbf{x}}^y,$$

i.e, the prediction is the label with the highest output. Intuitively if h is near the ground-truth label distribution function d, then the corresponding classification function $\hat{h}$ is close to the Bayes classification (we assume d is the conditional probability distribution function), thereby LDL is definitely related with classification. The mathematical tool which quantifies the relation between LDL and classification is the plug-in decision theorem [Devroye *et al.*, 1996]. Let h^* be the Bayes classification function, i.e.,

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

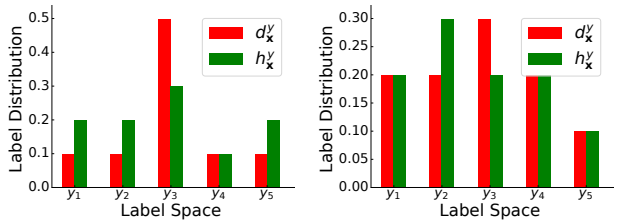
then we have

Theorem 1. [Devroye *et al.*, 1996; Wang and Geng, 2019] *The error probability difference between $\hat{h}$ and h^* satisfies*

$$\mathbb{P}(\hat{h}(\mathbf{x}) \neq y) - \mathbb{P}(h^*(\mathbf{x}) \neq y) \leq \mathbb{E}_{\mathbf{x}}\left[\sum_{y\in\mathcal{Y}} |h_{\mathbf{x}}^y - d_{\mathbf{x}}^y|\right].$$

The theorem says that if h is close to d in terms of absolute loss, then error probability of $\hat{h}$ is close to that of h^*. Theoretically LDL with absolute loss is directly relevant with classification. Also absolute loss is tighter than KL-divergence, since $|\mathbf{p} - \mathbf{q}| \leq 2\sqrt{\mathrm{KL}(\mathbf{p}, \mathbf{q})}$, for $\mathbf{p}, \mathbf{q} \in \mathbb{R}^m$ [Cover and Thomas, 2012]. Accordingly we propose to use absolute loss in LDL for classification, instead of KL-divergence. Similar with [Geng, 2016], we use maximum entropy model to learn the label distribution. Maximum entropy mode is defined as

$$h_{\mathbf{x}}^{y_j} = \frac{1}{Z}\exp(\mathbf{w}_j \cdot \mathbf{x}), \tag{1}$$



(a) $\ell_1 = 0.5, \ell_{\text{ep}} = 0.5$ (b) $\ell_1 = 0.2, \ell_{\text{ep}} = 0.8$

Figure 1: An example to illustrate the gap between absolute loss for LDL and error probability for classification. The red bar represents the ground-truth label distribution, and the green one represents the learned label distribution.

where $Z = \sum_j \exp(\mathbf{w}_j \cdot \mathbf{x})$ is the normalization factor. Absolute loss is applied as the measure for LDL, and consequently the problem can be formulated as

$$\min_{\mathbf{W}} \sum_{i=1}^{n} \sum_{j=1}^{m} \left| h_{\mathbf{x}_i}^{y_j} - d_{\mathbf{x}_i}^{y_j} \right| + \frac{C}{2} \sum_{j=1}^{m} ||\mathbf{w}_j||^2, \tag{2}$$

where $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_m]$ is the set of parameters.

3.3 Re-weighting with Information Entropy

After taking a second look at the Theorem (1), we will find out that model (2) actually optimizes the upper bound for error probability. This really brings a gap between absolute loss for LDL and 0/1 loss for classification. Fig. 1 is an illustrative example to demonstrate the gap, where ℓ_1 denotes absolute loss and ℓ_{ep} denotes error probability defined as Eq. (15). As we can see from the figure, (b) is superior to (a) from the perspective of absolute loss. However (b) is inferior to (a) in terms of error probability. For (a), y_3 has the highest label distribution both for d and h, thus prediction of $\hat{h}$ coincides with that of the Bayes classification function h^*. For (b), y_3 has the highest label distribution for d and y_2 has the highest label distribution for h, thus the corresponding classification result is different with that of the Bayes classification. Essentially for label distribution which is multimodal (like (b) of Fig.1), the corresponding classification function is much more sensitive to the loss of LDL. While, for label distribution which is unimodal (like (a) of Fig.1), there is more room to manoeuvre. Information entropy seems to be a reasonable tool to quantize this sensitiveness, since a multimodal distribution generally brings higher information entropy compared with a unimodal one. Precisely, for the label distribution with higher information entropy (i.e., multimodal), the corresponding classification model is much more sensitive to the LDL loss, and vice verse. In other words, samples with high information entropy deserves more attention, which reminds us to re-weight samples with information entropy. Reweighted with information entropy will penalize large loss for samples with high information entropy, and leave more room for samples with low information entropy. Recall the definition of entropy information , for $\mathbf{x}$,

$$E_{\mathbf{x}} = -\sum_{y \in \mathcal{Y}} d_{\mathbf{x}}^{y} \ln d_{\mathbf{x}}^{y}.$$

Accordingly the problems is then formulated as

$$\min_{\mathbf{W}} \sum_{i=1}^{n} E_{\mathbf{x}_i} \sum_{j=1}^{m} \left| h_{\mathbf{x}_i}^{y_j} - d_{\mathbf{x}_i}^{y_j} \right| + \frac{C}{2} \sum_{j=1}^{m} ||\mathbf{w}_j||^2. \tag{3}$$

3.4 LDL with Large Margin

To further boost the classification precision, we borrow the margin theory. Formally speaking, let the model output corresponding to the Bayes prediction has a large margin over other labels, which pushes the corresponding classification model consistent with the Bayes classification model. Let y_i^* be the Bayes prediction for $\mathbf{x}_i$, i.e., $y_i^* = h^*(\mathbf{x}_i)$. Then we assume a margin ρ between $h_{\mathbf{x}_i}^{y_i^*}$ and $\max_{y \in \{\mathcal{Y} - y_i^*\}} h_{\mathbf{x}_i}^{y}$, and the problem can be formulated as

$$\begin{aligned} \min_{\mathbf{W}} & \sum_{i=1}^{n} E_{\mathbf{x}_i} \sum_{j=1}^{m} \left| h_{\mathbf{x}_i}^{y_j} - d_{\mathbf{x}_i}^{y_j} \right| + C_1 \sum_{i=1}^{n} \xi_i + \frac{C_2}{2} \sum_{j=1}^{m} ||\mathbf{w}_j||^2, \\ \text{s.t.}: \quad & h_{\mathbf{x}_i}^{y_i^*} - \max_{y \in \{\mathcal{Y} - y_i^*\}} h_{\mathbf{x}_i}^{y} > \rho - \xi_i, \\ & \xi_i \geq 0, \quad \forall i \in [n], \end{aligned} \tag{4}$$

where C_1, C_2 are the balance coefficients for margin loss and regularization respectively. Large margin turns out to be a reasonable assumption since the ultimate goal of LDL4C is to pursuit the Bayes classifier. Overall, model (4) seeks a balance between LDL and large margin classifier.

3.5 Optimization

To start, for $\mathbf{x}$, define

$$\alpha = h_{\mathbf{x}}^{y_i^*} - \max_{y \in \{\mathcal{Y} - y_i^*\}} h_{\mathbf{x}}^{y}, \tag{5}$$

and ρ-margin loss as $\ell^{\rho}(\alpha) = \max\{0, 1 - \frac{\alpha}{\rho}\}$. By introducing margin loss, model (4) can be re-formed as

$$\min_{\mathbf{W}} \sum_{i=1}^{n} E_{\mathbf{x}_i} \sum_{j=1}^{m} \left| h_{\mathbf{x}_i}^{y_j} - d_{\mathbf{x}_i}^{y_j} \right| + C_1 \sum_{i=1}^{n} \ell^{\rho}(\alpha_i) + \frac{C_2}{2} \sum_{j=1}^{m} ||\mathbf{w}_j||^2. \tag{6}$$

which can be optimized efficiently by gradient-based method (ℓ^{ρ} is differential). We optimize the target function by BFGS [Nocedal and Wright, 2006]. Define ϕ as the step function,

$$\phi(x) = \begin{cases} 1 & \text{if } x \geq 0, \\ 0 & \text{otherwise}, \end{cases}$$

and denote the target function by T, gradient of which is obtained through

$$\begin{aligned} \frac{\partial T}{\partial \mathbf{w}_k} = & \sum_{i,j} E_{\mathbf{x}_i} \text{sign}(h_{\mathbf{x}_i}^{y_j} - d_{\mathbf{x}_i}^{y_j}) \frac{\partial h_{\mathbf{x}_i}^{y_j}}{\mathbf{w}_k} + C_2 \mathbf{w}_k + \\ & C_1 \sum_{i=1}^{n} \phi(\rho - \alpha_i) \left(\frac{\partial \max_{y \in \{\mathcal{Y} - y_i^*\}} h_{\mathbf{x}_i}^{y})}{\mathbf{w}_k} - \frac{\partial h_{\mathbf{x}_i}^{y_i^*}}{\mathbf{w}_k} \right). \end{aligned}$$

Moreover, gradient of h is got through

$$\frac{\partial h_{\mathbf{x}}^{y_j}}{\partial \mathbf{w}_k} = \left[\mathbf{1}_{\{j=k\}} \frac{\exp(\mathbf{w}_k \cdot \mathbf{x})}{\sum_i \exp(\mathbf{w}_i \cdot \mathbf{x}_i)} - \left(\frac{\exp(\mathbf{w}_j \cdot \mathbf{x})}{\sum_i \exp(\mathbf{w}_i \cdot \mathbf{x})} \right)^2 \right] \mathbf{x}_i.$$

4 Theoretical Results

In this section, we will demonstrate that, theoretically LDL4C seeks a trade-off between generalization and discrimination, where generalization is due to the LDL process, and discrimination is credited to the large margin classifier.

4.1 Generalization

Here by *generalization* we mean that the error probability of LDL4C converges to that of the Bayes classifier, by developing an upper bound for error probability of LDL4C. The basic steps are providing an upper bound for risk of LDL4C (with absolute loss) firstly, and then converting the risk bound into error probability bound by Theorem 1.

Notice that maximum entropy model, i.e., Eq. (1) can be regarded as function combination of softmax function and multi-output linear regression function. Formally, let SF be the softmax function, $\mathcal{F}$ be a family of functions for multi-output linear regression. Denote the family of functions for the maximum entropy model by $\mathcal{H}$, then $\mathcal{H} = \{\mathbf{x} \mapsto \text{SF} \circ f(\mathbf{x}) : f \in \mathcal{F}\}$. To bound the risk of LDL4C, it suffices to derive an upper bound on the Rademacher complexity [Bartlett and Mendelson, 2003] of the model.

Theorem 2. *Define $\mathcal{F}$ as $\mathcal{F} = \{\mathbf{x} \mapsto [\mathbf{w}_1 \cdot \mathbf{x}, \ldots, \mathbf{w}_m \cdot \mathbf{x}]^T : ||\mathbf{w}_j|| \leq 1\}$. Rademacher complexity of $\ell_1 \circ \mathcal{H}$ satisfies*

$$\hat{\mathcal{R}}_n(\ell_1 \circ \mathcal{H}) \leq \frac{2\sqrt{2}m^2 \max_{i\in[n]} ||\mathbf{x}_i||_2}{\sqrt{n}}. \tag{7}$$

Proof. Firstly, according to [Wang and Geng, 2019], $\ell_1 \circ$SF is shown to be $2m$-Lipschitz. And according to [Maurer, 2016], with $\ell_1 \circ$ SF being $2m$-Lipschitz, then

$$\hat{\mathcal{R}}_n(\ell_1 \circ \mathcal{H}) \leq 2\sqrt{2}m \sum_{j=1}^{m} \hat{\mathcal{R}}_n(\mathcal{F}_i \circ S), \tag{8}$$

where $\mathcal{F}_j = \{\mathbf{x} \mapsto \mathbf{w}_j \cdot \mathbf{x} : ||\mathbf{w}_j|| \leq 1\}$. Moreover, according to [Kakade *et al.*, 2009], Rademacher complexity of $\mathcal{F}_j$ satisfies

$$\hat{\mathcal{R}}_n(\mathcal{F}_j) \leq \frac{\max_{i\in[n]} ||\mathbf{x}_i||_2}{\sqrt{n}}. \tag{9}$$

Finally substitute Eq. (9) into Eq. (8), and we finish the proof for Theorem (2). □

Then data-dependent error probability for LDL4C is as following,

Theorem 3. *Define $\hat{\mathcal{H}} = \{\mathbf{x} \mapsto \max_{y\in\mathcal{Y}} h_{\mathbf{x}}^y : h \in \mathcal{H}\}$. Then for any $\delta > 0$, with probability at least $1-\delta$, for any $\hat{h} \in \hat{\mathcal{H}}$, such that*

$$\mathbb{P}(\hat{h}(\mathbf{x}) \neq y) - \mathbb{P}(h^*(\mathbf{x}) \neq y) \leq \sum_{i}^{n}\sum_{j}^{m} |h_{\mathbf{x}_i}^{y_j} - d_{\mathbf{x}_i}^{y_j}| + \frac{4\sqrt{2}m^2 \max_{i\in[n]} ||\mathbf{x}_i||_2}{\sqrt{n}} + 6\sqrt{\frac{\log 2/\delta}{2n}}. \tag{10}$$

Proof. Firstly, according to [Bartlett and Mendelson, 2003; Mohri *et al.*, 2012], and $\sum_y |d_{\mathbf{x}}^y - h_{\mathbf{x}}^y| \leq 2$ (triangle inequality), data-dependent risk bound for LDL4C is as

$$\mathbb{E}_{\mathbf{x}}\left[\sum_{j=1}^{m} |h_{\mathbf{x}}^{y_j} - d_{\mathbf{x}}^{y_j}|\right] \leq \sum_{i}^{n}\sum_{j}^{m} |h_{\mathbf{x}_i}^{y_j} - d_{\mathbf{x}_i}^{y_j}| + \frac{4\sqrt{2}m^2 \max_{i\in[n]} ||\mathbf{x}_i||_2}{\sqrt{n}} + 6\sqrt{\frac{\log 2/\delta}{2n}}. \tag{11}$$

Secondly combine Theorem (1) and Eq. (3), which completes the proof for the theorem. □

4.2 Discrimination

In this part, we will borrow margin theory to demonstrate that LDL4C enjoys discrimination. By discrimination, we mean the ability to output the same prediction as the Bayes prediction. According to [Bartlett and Mendelson, 2003], with margin loss, we have

Theorem 4. *Define $\tilde{\mathcal{H}}$ as $\tilde{\mathcal{H}} = \{(\mathbf{x}, y^*) \mapsto h(\mathbf{x}, y^*) - \max_{y\in\{\mathcal{Y}-y^*\}} h(\mathbf{x}, y) : h \in \mathcal{H}\}$. Fix $\rho > 0$. Then, for any $\delta > 0$, with probability at least $1-\delta$, for all $h \in \mathcal{H}$, such that*

$$\mathbb{E}\left[\ell^\rho(\alpha)\right] \leq \sum_{i=1}^{n} \ell^\rho(\alpha_i) + 2\hat{\mathcal{R}}_n(\ell^\rho \circ \tilde{\mathcal{H}}) + 3\sqrt{\frac{\log 2/\delta}{2n}}. \tag{12}$$

Since ℓ^ρ satisfies $1/\rho$-Lipschitzness, we have $\hat{\mathcal{R}}_n(\ell^\rho \circ \tilde{\mathcal{H}}) \leq \frac{1}{\rho}\hat{\mathcal{R}}_n(\tilde{\mathcal{H}})$. And according to [Mohri *et al.*, 2012], $\hat{\mathcal{R}}_n(\tilde{\mathcal{H}}) \leq 2m\hat{\mathcal{R}}_n(\Pi_1(\mathcal{H}))$, where $\Pi_1(\mathcal{H})$ is defined as

$$\Pi_1(\mathcal{H}) = \{\mathbf{x} \mapsto h(\mathbf{x}, y) : y \in \mathcal{Y}, h \in \mathcal{H}\}.$$

Since $\mathbf{1}_{\{h(\mathbf{x})\neq y^*\}} < \ell^\rho(\alpha)$, thus we have

$$\mathbb{E}\left[\mathbf{1}_{h(\mathbf{x})\neq y^*}\right] \leq \sum_{i=1}^{n} \ell^\rho(\alpha_i) + \frac{4m\hat{\mathcal{R}}_n(\Pi_1(\mathcal{H}))}{\rho} + 3\sqrt{\frac{\log\frac{2}{\delta}}{2n}}.$$

And $\hat{\mathcal{R}}_n(\Pi_1(\mathcal{H}))$ can be bounded as

$$\hat{\mathcal{R}}_n(\Pi_1(\mathcal{H})) \leq \frac{(m-1)\gamma\exp(\gamma)}{\sqrt{n}}, \tag{13}$$

where $\gamma = 2\max_{i\in[n]} ||\mathbf{x}_i||$, which leads to

Theorem 5. *Fix $\rho > 0$. Then for any $\delta > 0$, with probability at least $1-\delta$, for all $h \in \mathcal{H}$ such that*

$$\mathbb{E}\left[\mathbf{1}_{h(\mathbf{x})\neq y^*}\right] \leq \sum_{i=1}^{n} \ell^\rho(\alpha_i) + \frac{4(m-1)m\gamma\exp(\gamma)}{\rho\sqrt{n}} + 3\sqrt{\frac{\log\frac{2}{\delta}}{2n}}. \tag{14}$$

5 Experiments

5.1 Experimental Configuration

Real-word datasets. The experiments are extensively conducted on seventeen datasets totally, among which fifteen are from [Geng, 2016], and M^{2}B is from [Nguyen *et al.*, 2012], and SCUT-FBP is from [Xie *et al.*, 2015]. Note that M^{2}B and SCUT-FBP are transformed to label distribution as [Ren and Geng, 2017]. Datasets are from a variety of fields, and statistics about the datasets are listed as Table 1.

Dataset	# Examples	# Features	# Labels
SJAFFE	213	243	6
M^2B	1240	250	5
SCUT-FBP	1500	300	5
Natural Scene	2000	294	9
Yeast-alpha	2465	24	18
Yeast-cdc	2465	24	15
Yeast-cold	2465	24	4
Yeast-diau	2465	24	7
Yeast-dtt	2465	24	4
Yeast-elu	2465	24	14
Yeast-heat	2465	24	6
Yeast-spo	2465	24	6
Yeast-spo5	2465	24	3
Yeast-spoem	2465	24	2
SBU_3DFE	2500	243	6
Movie	7755	1869	5
Human Gene	17892	36	68

Table 1: Statistics of 17 real-world datasets used in the experiments.

Dataset	LDL	LDL+ℓ_1	LDL+RW	LDL+LM
SJAFFE	.437	.417	**.394**	.426
M^2B	.486	.486	**.482**	.485
SCUT-FBP	.467	.470	**.465**	.467
Natural Scene	.597	.420	**.421**	.425
Yeast-alpha	.891	**.787**	**.787**	**.787**
Yeast-cdc	.823	**.820**	.821	.821
Yeast-cold	.580	**.576**	**.576**	**.576**
Yeast-diau	.694	**.665**	.668	.669
Yeast-dtt	.629	.630	**.627**	**.627**
Yeast-heat	.702	.676	**.675**	.677
Yeast-spo	.548	**.547**	**.547**	**.547**
Yeast-spoem	.435	**.400**	.416	.422
Movie	.416	.416	**.410**	.414

Table 2: Experimental results of comparing BFGS-LDL with BFGS-LDL + { ℓ_1, RW, LM } in terms of 0/1 loss. The best results are highlighted in bold face.

Comparing algorithms. We compare LDL4C with six popular LDL algorithms, i.e., PT-Bayes, PT-SVM, AA-BP, AA-kNN, LDL-SVR [Geng and Hou, 2015], BFGS-LDL, and two state-of-the-art LDL algorithms, i.e., StructRF [Chen *et al.*, 2018] and LDL-SCL [Zheng *et al.*, 2018]. For LDL4C, the balance parameter C_1 and C_2 are selected from $\{0.001, 0.01, 0.1, 1, 10, 100\}$ and ρ is chosen from $\{0.001, 0.01, 0.1\}$ by cross validation. Moreover for AA-BP, the number of hidden-layer neurons is set to 64, and for AA-kNN, the number of nearest neighbors k is selected from $\{3, 5, 7, 9, 11\}$. Furthermore, for BFGS-LDL, we follow the same settings with [Geng, 2016]. Besides, the insensitive parameter ϵ is set to 0.1 as suggested by [Geng and Hou, 2015], and rbf kernel is used for both PT-SVM and LDL-SVR. For StructRF, depth is set to 20 and number of trees is set to 50, and sampling ratio is set to 0.8 as suggested by [Chen *et al.*, 2018]. In addition, for LDL-SCL, $\lambda_1 = 0.001$, $\lambda_2 = 0.001$, $\lambda_3 = 0.001$ and number of clusters $m = 6$ as given in [Zheng *et al.*, 2018]. Finally, all algorithms are examined on 17 datasets with 10-fold cross validation, and average performance is reported.

Evaluation metrics. We test algorithms in terms of two measures, 0/1 loss and error probability. For 0/1 loss, we regard the label with the highest label distribution as the ground-truth label. Given an instance $\mathbf{x}$ with prediction y, the error probability is defined as

$$\ell_{\text{ep}}(\mathbf{x}, y) = 1 - d_{\mathbf{x}}^{y}. \tag{15}$$

5.2 Experimental Results

Firstly, to validate the three key components (i.e., absolute loss, re-weighting with information entropy, and large margin) for boosting classification precision, we add each of the components into BFGS-LDL and compare 0/1 loss with BFGS-LDL. The performance is reported in Table 2. Due to the page limitation, we only report part of the results. Here "LDL" is short for BFGS-LDL. "LDL+ℓ_1" represents BFGS-LDL with absolute loss, and "LDL+RW" represents BFGS-LDL with information entropy re-weighting, and "LDL+LM" stands for BFGS-LDL with large margin. As we can see from the table, armed with three key components, the corresponding LDL model can achieve better classification performance compared with the original BFGS-LDL model. Secondly we conduct extensive experiments of LDL4C and comparing methods on 17 real-world datasets in terms of 0/1 loss and error probability, and the experimental results are presented as in Table 3 and Table 4 respectively. The pairwise t-test at 0.05 significance level is conducted, with the best performance marked in bold face. From the tables, we can get that:

- In terms of 0/1 loss, LDL4C rank 1st among 13 datasets, and is significantly superior to other comparing methods in 70% cases, which validates the ability of LDL4C for performing classification.
- In terms of error probability, LDL4C ranks 1st among 12 datasets, and achieves significant better performance in 46% cases, which discloses that LDL4C has better generalization.

6 Conclusions

This paper tackles the inconsistency between the training and test phrase of LDL when applied in applications. In the training phrase of LDL, aim of which is to learn the given label distribution, i.e., minimizing (maximizing) distance (similarity) between the model output and the given label distribution. However, for applications of LDL, a learned LDL model is generally treated as a classification model, with the label corresponding to the highest output as the prediction. The proposed algorithm LDL4C alleviates the inconsistency with three key improvements. The first one is inspired by the plug-in decision Theorem 1, which states that absolute loss is directly related with classification error, thereby absolute loss is applied as the loss function for LDL4C. The second one stems from noticing the gap between absolute loss and the classification error, which suggests re-weighting samples with information entropy to pay more attention to multimodal distribution. The last one is due to margin theory, which introduces a margin between $h_{\mathbf{x}}^{y^*}$ and $\max_{y \in \{\mathcal{Y} - y^*\}} h_{\mathbf{x}}^{y}$ to enforce the corresponding classification model approaching the Bayes decision rule. Furthermore, we explore the effectiveness of LDL4C theoretically and empirically. Theoretically, LDL4C enjoy both generalization and discrimination, and empirically extensive comparing experiments clearly manifest the advantage of LDL4C in classification.

Dataset	PT-Bayes	PT-SVM	AA-BP	AA-kNN	LDL-SVR	StructRF	LDL-SCL	BFGS-LDL	LDL4C
SJAFFE	.783±.016•	.785±.003•	.807±.003•	.483±.012•	.686±.004•	.569±.011•	.789±.002•	.437±.018	**.394±.010**
M^{2}B	.645±.003•	.552±.003•	.487±.001	.484±.001	.495±.002	.483±.002	.498±.001	.486±.002	**.481±.001**
SCUT-FBP	.893±.002•	.517±.001	.463±.001	**.457±.001**	.467±.001	.467±.001	.502±.001•	.467±.000	.465±.001
Natural Scene	.801±.002•	.649±.000•	.667±.001•	.521±.001•	.568±.001•	.520±.001•	.696±.001•	.597±.002•	**.420±.001**
Yeast-alpha	.945±.000•	.940±.000•	.886±.000•	.882±.000•	.925±.000•	.882±.001•	.903±.000•	.891±.000•	**.787±.001**
Yeast-cdc	.935±.000•	.926±.000•	.825±.001	.830±.001•	.858±.000•	.824±.001	.825±.000	.823±.001	**.818±.001**
Yeast-cold	.730±.000•	.733±.001•	.580±.001	**.570±.001**○	.609±.000•	.585±.001•	.586±.001•	.580±.001	.575±.001
Yeast-diau	.866±.000•	.848±.001•	.711±.001•	.688±.000•	.694±.001•	.678±.000	.703±.001•	.694±.001•	**.665±.000**
Yeast-dtt	.743±.001•	.740±.001•	.641±.001	.637±.001	.651±.001•	.622±.001	.636±.001	.629±.001	**.627±.001**
Yeast-elu	.930±.000•	.924±.000•	.917±.000•	.867±.001•	.892±.000•	.875±.000•	.903±.000•	.899±.000•	**.803±.000**
Yeast-heat	.824±.001•	.827±.000•	.707±.001•	.692±.001	.710±.001•	.680±.001	.717±.001•	.703±.001•	**.675±.001**
Yeast-spo	.826±.001•	.817±.001•	.548±.000	.565±.001•	.616±.000•	.566±.001•	.584±.000•	.548±.000	**.547±.000**
Yeast-spo5	.667±.001•	.637±.000•	.543±.001	.542±.000	.559±.001	**.524±.001**	.622±.000•	.559±.001•	.534±.001
Yeast-spoem	.482±.000•	.479±.001•	.441±.001•	.410±.001	.409±.000	**.401±.002**	.439±.002•	.435±.002•	**.401±.000**
SBU_3DFE	.818±.001•	.810±.001•	.675±.001•	.616±.001•	.633±.002•	**.516±.001**○	.514±.001○	.582±.001	.569±.001
Movie	.819±.000•	.677±.000 •	.423±.001•	.427±.000•	.415±.000•	.410±.000	.454±.000•	.416±.000•	**.409±.000**
Human Gene	.986±.000•	.981±.000•	.951±.001•	.966±.000•	.977±.000•	.965±.000•	.976±.000•	.955±.000•	**.927±.000**

Table 3: Experimental results (mean ± std.) of LDL4C and comparing algorithms in terms of 0/1 loss on 17 real-word datasets. In addition, •/○ indicates whether LDL4C is statistically superior/inferior to the comparing algorithms.

Dataset	PT-Bayes	PT-SVM	AA-BP	AA-kNN	LDL-SVR	StructRF	LDL-SCL	BFGS-LDL	LDL4C
SJAFFE	.8080±3e-4•	.8305±4e-4•	.8188±2e-4•	.7693±6e-4•	.7951±1e-4•	.7766±5e-4•	.8309±2e-4•	.7583±6e-4	**.7573±4e-4**
M^{2}B	.6438±9e-4•	.5441±4e-4	.5436±4e-4	.5424±4e-4	.5465±3e-4	.5427±6e-4	.5494±4e-4	.5419±6e-4	**.5358±6e-4**
SCUT-FBP	.8905±6e-4•	.5414±2e-4	.5410±2e-4	.5379±3e-4	.5420±2e-4	.5486±3e-4•	.5494±4e-4	.5426±2e-4	**.5414±2e-4**
Natural Scene	.8013±9e-4•	.6478±1e-4	.6620±3e-4•	.6658±2e-4•	.6400±1e-4	**.6200±2e-4**○	.6648±2e-4•	.6471±2e-4	.6450±3e-4
Yeast-alpha	.9444±0e-4•	.9442±0e-4•	.9427±0e-4	.9428±0e-4	.9435±0e-4•	.9429±0e-4•	.9429±0e-4•	**.9426±0e-4**	**.9426±0e-4**
Yeast-cdc	.9329±0e-4•	.9320±0e-4•	**.9287±0e-4**	.9290±0e-4	.9298±0e-4•	.9288±0e-4	.9289±0e-4•	**.9287±0e-4**	**.9287±0e-4**
Yeast-cold	.7465±0e-4•	.7402±0e-4•	.7297±0e-4	.7297±0e-4	.7350±0e-4•	**.7291±0e-4**	.7304±0e-4•	.7297±0e-4	.7296±0e-4
Yeast-diau	.8524±0e-4•	.8478±0e-4•	.8432±0e-4	**.8424±0e-4**	.8432±0e-4	**.8424±0e-4**	.8427±0e-4	.8428±0e-4	.8428±0e-4
Yeast-dtt	.7489±0e-4•	.7486±0e-4•	.7421±0e-4•	.7415±0e-4	.7429±0e-4•	.7430±0e-4•	.7418±0e-4	.7416±0e-4	**.7412±0e-4**
Yeast-elu	.9285±0e-4•	.9277±0e-4•	.9261±0e-4	.9261±0e-4	.9266±0e-4•	**.9260±0e-4**	.9262±0e-4	.9261±0e-4	**.9260±0e-4**
Yeast-heat	.8299±0e-4•	.8309±0e-4	.8238±0e-4	.8237±0e-4	.8250±0e-4	.8234±0e-4	.8250±0e-4•	.8234±0e-4	**.8233±0e-4**
Yeast-spo	.8324±0e-4•	.8165±0e-4•	.8101±0e-4	.8109±0e-4	.8137±0e-4•	.8105±0e-4	.8115±0e-4	.8101±0e-4	**.8000±0e-4**
Yeast-spo5	.6601±0e-4•	.6551±0e-4	.6511±0e-4	.6473±0e-4	.6487±0e-4	**.6430±0e-4**○	.6533±0e-4	.6527±0e-4	.6526±0e-4
Yeast-spoem	.4909±0e-4	.4756±1e-4	.4717±1e-4	.4689±0e-4	.4720±0e-4	**.4670±1e-4**	.4713±1e-4	.4705±0e-4	.4700±0e-4
SBU_3DFE	.7991±1e-4•	.8093±1e-4•	.8041±1e-4•	.7875±0e-4•	.7937±0e-4•	.7745±0e-4	.7769±0e-4•	.7746±0e-4	**.7734±0e-4**
Movie	.8179±0e-4•	.6815±0e-4•	.6760±0e-4•	.6781±0e-4•	.6760±0e-4•	.6745±0e-4	.6844±0e-4•	.6755±0e-4•	**.6743±0e-4**
Human Gene	.9848±0e-4•	.9834±0e-4•	.9824±0e-4•	.9831±0e-4•	.9822±0e-4•	.9626±0e-4•	.9822±0e-4•	**.9816±0e-4**	**.9816±0e-4**

Table 4: Experimental results (mean ± std.) of LDL4C and comparing algorithms in terms of error probability on 17 real-word datasets.In addition, •/○ indicates whether LDL4C is statistically superior/inferior to the comparing algorithms.

Acknowledgments

This research was supported by the National Key Research & Development Plan of China (No. 2017YFB1002801), the National Science Foundation of China (61622203), the Collaborative Innovation Center of Novel Software Technology and Industrialization, and the Collaborative Innovation Center of Wireless Communications Technology.

A Proof for Eq. (13)

Proof. Recall the definition of Rademacher complexity

$$\begin{aligned}\hat{\mathcal{R}}(\Pi_1(\mathcal{H})) &= \mathbb{E}_\sigma\left[\frac{1}{n}\sup_{\mathbf{w}_y,\mathbf{W}}\sum_{i=1}^{n}\frac{\exp(\mathbf{x}_i\cdot\mathbf{w}_y)}{\sum_j\exp(\mathbf{w}_j\cdot\mathbf{x}_i)}\sigma_i\right]\\ &= \mathbb{E}_\sigma\left[\frac{1}{n}\sup_{\mathbf{w}_y,\mathbf{W}}\sum_{i=1}^{n}\frac{\sigma_i}{\sum_j \mathrm{e}^{(\mathbf{w}_j-\mathbf{w}_y)\cdot\mathbf{x}_i}}\right]\\ &\le \mathbb{E}_\sigma\left[\frac{1}{n}\sup_{\mathbf{w}_y,\mathbf{W}}\sum_{i,j\neq y}\mathrm{e}^{(\mathbf{w}_j-\mathbf{w}_y)\cdot\mathbf{x}_i}\sigma_i\right]\\ &\le \sum_{j\neq y}\frac{1}{n}\mathbb{E}_\sigma\left[\sup_{\mathbf{w}_y,\mathbf{w}_j}\sum_{i=1}^{n}\mathrm{e}^{(\mathbf{w}_j-\mathbf{w}_y)\cdot\mathbf{x}_i}\sigma_i\right],\end{aligned}$$

where $\sigma_1,\sigma_2,\ldots,\sigma_n$ are n independent random variables with $\mathbb{P}(\sigma_i=-1)=\mathbb{P}(\sigma_i=1)=1/2$, and $\mathbf{W}=[\mathbf{w}_1,\mathbf{w}_2,\ldots,\mathbf{w}_m]$ with $||\mathbf{w}_j||<1$, and the third inequality is according to the 1-Lipschitzness of function $1/x$ for $x>1$. And for each $j\neq y$,

$$\begin{aligned}\mathbb{E}_\sigma\left[\sup_{y,j}\sum_{i=1}^{n}\frac{\sigma_i}{n}\mathrm{e}^{(\mathbf{w}_j-\mathbf{w}_y)\mathbf{x}_i}\right] &\le \mathrm{e}^{\gamma}\mathbb{E}_\sigma\left[\sup_{y,j}\sum_{i=1}^{n}\frac{\sigma_i(\mathbf{w}_j-\mathbf{w}_y)\mathbf{x}_i}{n}\right]\\ &\le \frac{\gamma\mathrm{e}^{\gamma}}{\sqrt{n}},\end{aligned}$$

where the first inequality is according to the e^a-Lipschitzness of function e^x for $x<a$, and the second one is according to [Kakade *et al.*, 2009] by noticing that $||\mathbf{w}_j-\mathbf{w}_y||<2$, which concludes the proof for Eq. (13). □

References

[Bartlett and Mendelson, 2003] Peter L. Bartlett and Shahar Mendelson. Rademacher and gaussian complexities: Risk bounds and structural results. *J. Mach. Learn. Res.*, 3:463–482, March 2003.

[Chen *et al.*, 2018] Mengting Chen, Xinggang Wang, Bin Feng, and Wenyu Liu. Structured random forest for la-

bel distribution learning. *Neurocomputing*, 320:171 – 182, 2018.

[Cover and Thomas, 2012] Thomas M Cover and Joy A Thomas. *Elements of information theory*. John Wiley & Sons, 2012.

[Devroye *et al.*, 1996] Luc Devroye, László Györfi, and Gábor Lugosi. *A probabilistic theory of pattern recognition*, volume 31. Springer Science & Business Media, 1996.

[Gao *et al.*, 2017] Binbin Gao, Chao Xing, Chenwei Xie, Jianxin Wu, and Xin Geng. Deep label distribution learning with label ambiguity. *IEEE Transactions on Image Processing*, 26(6):2825–2838, June 2017.

[Gao *et al.*, 2018] Binbin Gao, Hongyu Zhou, Jianxin Wu, and Xin Geng. Age estimation using expectation of label distribution learning. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*, pages 712–718. International Joint Conferences on Artificial Intelligence Organization, 7 2018.

[Geng and Hou, 2015] Xin Geng and Peng Hou. Pre-release prediction of crowd opinion on movies by label distribution learning. In *Proceedings of the 24th International Conference on Artificial Intelligence*, IJCAI'15, pages 3511–3517, 2015.

[Geng *et al.*, 2013] Xin Geng, Chao Yin, and Zhihua Zhou. Facial age estimation by learning from label distributions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(10):2401–2412, Oct 2013.

[Geng, 2016] Xin Geng. Label distribution learning. *IEEE Transactions on Knowledge and Data Engineering*, 28(7):1734–1748, July 2016.

[Huo and Geng, 2017] Zengwei Huo and Xin Geng. Ordinal zero-shot learning. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, pages 1916–1922, 2017.

[Kakade *et al.*, 2009] Sham M Kakade, Karthik Sridharan, and Ambuj Tewari. On the complexity of linear prediction: Risk bounds, margin bounds, and regularization. In *Advances in Neural Information Processing Systems, NIPS'09*, pages 793–800, 2009.

[Maurer, 2016] Andreas Maurer. A vector-contraction inequality for rademacher complexities. In *Algorithmic Learning Theory*, pages 3–17, 2016.

[Mohri *et al.*, 2012] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of machine learning*. MIT press, 2012.

[Nguyen *et al.*, 2012] Tam V. Nguyen, Si Liu, Bingbing Ni, Jun Tan, Yong Rui, and Shuicheng Yan. Sense beauty via face, dressing, and/or voice. In *Proceedings of the 20th ACM International Conference on Multimedia*, MM '12, pages 239–248, New York, USA, 2012. ACM.

[Nocedal and Wright, 2006] Jorge Nocedal and Stephen Wright. *Numerical optimization*. Springer Science & Business Media, 2006.

[Ren and Geng, 2017] Yi Ren and Xin Geng. Sense beauty by label distribution learning. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, pages 2648–2654, 2017.

[Shen *et al.*, 2017] Wei Shen, Kai Zhao, Yilu Guo, and Alan L. Yuille. Label distribution learning forests. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'30, pages 834–843. Curran Associates, Inc., 2017.

[Wang and Geng, 2019] Jing Wang and Xin Geng. Theoretical analysis of label distribution learning (in press). In *AAAI Conference on 33th AAAI Conference on Artificial Intelligence, AAAI'19*, 2019.

[Xie *et al.*, 2015] Duorui Xie, Lingyu Liang, Lianwen Jin, Jie Xu, and Mengru Li. Scut-fbp: A benchmark dataset for facial beauty perception. *2015 IEEE International Conference on Systems, Man, and Cybernetics*, Oct 2015.

[Xu *et al.*, 2018] Ning Xu, An Tao, and Xin Geng. Label enhancement for label distribution learning. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*, pages 2926–2932. International Joint Conferences on Artificial Intelligence Organization, 7 2018.

[Zhang and Zhou, 2014] Minling Zhang and Zhihua Zhou. A review on multi-label learning algorithms. *IEEE Transactions on Knowledge and Data Engineering*, 26(8):1819–1837, Aug 2014.

[Zhang *et al.*, 2015] Zhaoxiang Zhang, Wang Mo, and Geng Xin. Crowd counting in public video surveillance by label distribution learning. *Neurocomputing*, 166(1):151–163, 2015.

[Zheng *et al.*, 2018] Xiang Zheng, Xiuyi Jia, and Weiwei Li. Label distribution learning by exploiting sample correlations locally, 2018.

[Zhou and Zhang, 2006] Zhihua Zhou and Minling Zhang. Multi-instance multi-label learning with application to scene classification. In *Proceedings of the 19th International Conference on Neural Information Processing Systems*, NIPS'06, pages 1609–1616, Cambridge, MA, USA, 2006. MIT Press.

[Zhou *et al.*, 2015] Ying Zhou, Hui Xue, and Xin Geng. Emotion distribution recognition from facial expressions. In *Proceedings of the 23rd ACM International Conference on Multimedia*, MM '15, pages 1247–1250, New York, NY, USA, 2015. ACM.