Learning with Multiple Complementary Labels

Lei Feng [*1] **Takuo Kaneko** [*2 3] **Bo Han** [4 3] **Gang Niu** [3] **Bo An** [1] **Masashi Sugiyama** [3 2]

Abstract

A *complementary label* (CL) simply indicates an incorrect class of an example, but learning with CLs results in multi-class classifiers that can predict the correct class. Unfortunately, the problem setting only allows a *single* CL for each example, which notably limits its potential since our labelers may easily identify *multiple CLs* (MCLs) to one example. In this paper, we propose a novel problem setting to allow MCLs for each example and two ways for learning with MCLs. In the first way, we design two *wrappers* that *decompose* MCLs into many single CLs, so that we could use any method for learning with CLs. However, the supervision information that MCLs hold is conceptually diluted after decomposition. Thus, in the second way, we derive an *unbiased risk estimator*; minimizing it processes each set of MCLs as a whole and possesses an *estimation error bound*. We further improve the second way into minimizing properly chosen upper bounds. Experiments show that the former way works well for learning with MCLs but the latter is even better.

1. Introduction

Ordinary machine learning tasks generally require massive data with accurate supervision information, while it is expensive and time-consuming to collect the data with high-quality labels. To alleviate this problem, the researchers have studied various weakly supervised learning frameworks (Zhou, 2018), including *semi-supervised learning* (Chapelle et al., 2006; Li & Liang, 2019; Miyato et al., 2018; Niu et al., 2013; Zhu & Goldberg, 2009), *positive-unlabeled learning* (du Plessis et al., 2014; 2015; Elkan & Noto, 2008; Kiryo et al., 2017; Sakai et al., 2017; 2018), *noisy-label learning* (Han et al., 2018a;b; Menon et al., 2015; Wei et al., 2020; Xia et al., 2019), *partial label learning* (Cour et al., 2011; Feng & An, 2018; 2019a;b; Zhang & Yu, 2015), *positive-confidence learning* (Ishida et al., 2018), *similar-unlabeled learning* (Bao et al., 2018), and *unlabeled-unlabeled classification* (Lu et al., 2019; 2020).

Here, we consider another weakly supervised classification framework called *complementary-label learning* (Ishida et al., 2017; 2019; Yu et al., 2018). In complementary-label learning, each training example is supplied with a *complementary label* (CL), which specifies one of the classes that the example does *not* belong to. Compared with ordinary labels, it is obviously easier to collect CLs. Recently, complementary-label learning has been applied to online learning (Kaneko et al., 2019) and medical image segmentation (Rezaei et al., 2019). In addition, another potential application of learning with CLs would be data privacy. For example, collecting some survey data may require extremely private questions (Ishida et al., 2017; 2019). It may be difficult for us to directly obtain the true answer (label) to the question. Nonetheless, it would be mentally less demanding if we ask the respondent to provide some incorrect answers. Besides, the respondent may provide multiple incorrect answers, rather than exactly one. In this case, *multiple complementary labels* (MCLs) would be more widespread than a single CL.

In this paper, we propose a novel problem setting (Section 3.1) that allows MCLs for each example, and provide a real-world motivation (Section 3.2). Although existing complementary-label learning approaches (Ishida et al., 2017; 2019; Yu et al., 2018) have provided solid theoretical foundations and achieved promising performance, they are all restricted to the case where each example is associated with a single CL. To learn with MCLs, we first design two wrappers (Section 4.1) that decompose each example with MCLs into multiple examples, each with a single CL, in different manners. With the two wrappers, we are able to use arbitrary ordinary complementary-label learning approaches for learning with MCLs. However, the derived data with many single CLs may not match the assumed data distribution for complementary-label learning (Ishida et al., 2017; 2019). In addition, the supervision information would be

[*]Equal contribution [⋆]Work done when LF was an intern at RIKEN AIP and TK belonged to The University of Tokyo and RIKEN AIP. [1]School of Computer Science and Engineering, Nanyang Technological University, Singapore [2]The University of Tokyo [3]RIKEN Center for Advanced Intelligence Project [4]Department of Computer Science, Hong Kong Baptist University. Correspondence to: Lei Feng <feng0093@e.ntu.edu.sg>.

Proceedings of the 37^{th} International Conference on Machine Learning, Online, PMLR 119, 2020.

conceptually diluted after decomposition.

In order to solve the above problems, we further propose an unbiased risk estimator (Section 4.2) for learning with MCLs, which processes each set of MCLs as a whole. Our risk estimator is conceptually consistent, and builds a prototype baseline for the new problem setting that may inspire more specially designed methods for this new setting in the future. Then, we theoretically derive an estimation error bound, which guarantees that the empirical risk minimizer converges to the true risk minimizer with high probability as the number of training data approaches infinity. Furthermore, we improve the risk estimator into minimizing properly chosen upper bounds for practical implementation (Section 4.3), and we show that they bring benefits to gradient update. Experimental results show that the wrappers work well for learning with MCLs while the (improved) risk estimator is even better on various benchmark datasets.

2. Related Work

In this section, we introduce some notations and briefly review the formulations of multi-class classification and complementary-label learning.

2.1. Multi-Class Classification

Suppose the feature space is $\mathcal{X} \in \mathbb{R}^d$ with d dimensions and the label space is $\mathcal{Y} = \{1, 2, \ldots, k\}$ with k classes, the instance $\boldsymbol{x} \in \mathcal{X}$ with its class label $y \in \mathcal{Y}$ is sampled from an unknown probability distribution with density $p(\boldsymbol{x}, y)$. Ordinary multi-class classification aims to induce a learning function $f(\boldsymbol{x}) : \mathbb{R}^d \to \mathbb{R}^k$ that minimizes the classification risk:

$$R(f) = \mathbb{E}_{p(\boldsymbol{x},y)}\big[\mathcal{L}\big(f(\boldsymbol{x}), y\big)\big], \tag{1}$$

where $\mathcal{L}\big(f(\boldsymbol{x}), y\big)$ is a multi-class loss function. The predicted label is given as $\hat{y} = \operatorname{argmax}_{y\in\mathcal{Y}} f_y(\boldsymbol{x})$, where $f_y(\boldsymbol{x})$ is the y-th coordinate of $f(\boldsymbol{x})$.

2.2. Complementary-Label Learning

Suppose the dataset for complementary-label learning is denoted by $\{(\boldsymbol{x}_i, \bar{y}_i)\}_{i=1}^n$, where $\bar{y}_i \in \mathcal{Y}$ is a complementary label of $\boldsymbol{x}_i$, and each complementarily labeled example is sampled from $\bar{p}(\boldsymbol{x}, \bar{y})$. Ishida et al. (2017; 2019) assumed that $\bar{p}(\boldsymbol{x}, \bar{y})$ is expressed as:

$$\bar{p}(\boldsymbol{x}, \bar{y}) = \tfrac{1}{k-1} \textstyle\sum_{y\neq\bar{y}} p(\boldsymbol{x}, y). \tag{2}$$

This assumption implies that all other labels except the correct label are chosen to be the complementary label with uniform probabilities. This is reasonable as we do not have extra labeling information expect a complementary label. Under this assumption, it was proved by Ishida et al. (2017) that an unbiased estimator of the original classification risk can be obtained from only complementarily labeled data, when the loss function satisfies certain conditions. Specifically, they used the multi-class loss functions with the one-versus-all strategy and the pairwise comparison strategy (Zhang, 2004):

$$\bar{\mathcal{L}}_{\mathrm{OVA}}\big(f(\boldsymbol{x}), \bar{y}\big) = \tfrac{1}{k-1} \textstyle\sum_{y'\neq\bar{y}} \ell\big(f_{y'}(\boldsymbol{x})\big) + \ell\big(-f_{\bar{y}}(\boldsymbol{x})\big),$$
$$\bar{\mathcal{L}}_{\mathrm{PC}}\big(f(\boldsymbol{x}), \bar{y}\big) = \textstyle\sum_{y'\neq\bar{y}} \ell\big(f_{y'}(\boldsymbol{x}) - f_{\bar{y}}(\boldsymbol{x})\big),$$

where $\ell(z)$ is a binary loss function that satisfies $\ell(z) + \ell(-z) = 1$, such as sigmoid loss $\ell_{\mathrm{S}}(z) = \frac{1}{1+e^z}$ and ramp loss $\ell_{\mathrm{R}}(z) = \frac{1}{2}\max(0, \min(2, 1-z))$.

Later, another different assumption was used by Yu et al. (2018). They assumed that all other labels except the correct label are chosen to be the complementary label with different probabilities, and proposed to estimate the class transition probability matrix for model training. Although they showed that the minimizer of their learning objective coincides with the minimizer of the original classification risk, they did not provide an unbiased risk estimator.

Recently, a more general unbiased risk estimator (Ishida et al., 2019) was proposed, which does not rely on specific losses or models. Their formulation is as follows:

$$\bar{\mathcal{L}}_{\mathrm{FREE}}\big(f(\boldsymbol{x}), \bar{y}\big) = \sum_{y=1}^{k} \mathcal{L}\big(f(\boldsymbol{x}), y\big) - (k-1)\mathcal{L}\big(f(\boldsymbol{x}), \bar{y}\big). \tag{3}$$

For this formulation, they showed that due to the negative term, the empirical risk could be unbounded below, which leads to over-fitting. In order to alleviate this issue, the authors further proposed modified versions by using the max operator and the gradient ascent strategy.

In summary, although the above methods have provided solid theoretical foundations and achieved promising performance for complementary-label learning, they are all restricted to the case where each example is associated with a single CL. In this paper, we propose a novel problem setting that allows MCLs for each example.

3. Multiple Complementary Labels

In this section, we first introduce our problem setting where each example is associated with MCLs, and then provide a corresponding real-world motivation.

3.1. Data Generation Process

Suppose the given dataset for learning with MCLs is represented as $\bar{\mathcal{D}} = \{(\boldsymbol{x}_i, \bar{Y}_i)\}_{i=1}^n$, where $\bar{Y}_i$ is a set of complementary labels for the instance $\boldsymbol{x}_i$. It is obvious that learning with MCLs is a generalization of complementary-label learning that learns with a single CL. Specifically, if

$\bar{Y}_i$ contains only one complementary label with probability 1, we obtain a complementary-label learning problem. In addition, if $\bar{Y}_i$ contains $k-1$ complementary labels where k denotes the total number of classes, we obtain an ordinary multi-class classification problem. It is easy to know that for all i, $\bar{Y}_i$ cannot be the empty set nor the full label set, hence $\bar{Y}_i \in \bar{\mathcal{Y}}$ where $\bar{\mathcal{Y}} = \{2^{\mathcal{Y}} - \varnothing - \mathcal{Y}\}$ and $|\bar{\mathcal{Y}}| = 2^k - 2$.

For the generation process of each example with MCLs, we assume that it relies on the size of the set of MCLs. Let us represent the size of the complementary label set by a random variable s, and assume s is sampled from a distribution $p(s)$. In this way, we assume that each training example $(\boldsymbol{x}_i, \bar{Y}_i)$ is drawn from the following data distribution:

$$\bar{p}(\boldsymbol{x}, \bar{Y}) = \sum_{j=1}^{k-1} p(s=j)\bar{p}(\boldsymbol{x}, \bar{Y} \mid s=j), \tag{4}$$

where

$$\bar{p}(\boldsymbol{x}, \bar{Y} \mid s=j) := \begin{cases} \frac{1}{\binom{k-1}{j}} \sum_{y \notin \bar{Y}} p(\boldsymbol{x}, y), & \text{if } |\bar{Y}| = j, \\ 0, & \text{otherwise.} \end{cases}$$

It is clear that when $p(s=1)=1$, our introduced distribution reduces to the assumed distribution (e.g., Eq. (2)) in ordinary complementary-label learning approaches (Ishida et al., 2017; 2019). Then, we show that $\bar{p}(\boldsymbol{x}, \bar{Y})$ is a valid probability distribution by the following theorem.

Theorem 1. *The following equality holds:*

$$\int_{\bar{\mathcal{Y}}} \int_{\mathcal{X}} \bar{p}(\boldsymbol{x}, \bar{Y}) \mathrm{d}\boldsymbol{x} \, \mathrm{d}\bar{Y} = 1. \tag{5}$$

The proof is provided in Appendix A.1.

3.2. Real-World Motivation

Here, we present a real-world motivation for the assumed data distribution.

Since directly choosing the correct label is hard for labelers, it would be easier if a labeling system can randomly choose a label set and ask labelers whether the correct label is included in the proposed label set or not. Given a pattern $\boldsymbol{x}$, suppose the labeling system first randomly samples the size s of the proposed label set from $p(s)$, and then randomly and uniformly chooses a specific label set with size s from $\bar{\mathcal{Y}}$. In this way, the collected label sets that do not include the correct label precisely follow the same distribution as Eq. (4). We will demonstrate this fact in the following.

We start by considering the case where the labeling system has already sampled the size s of the proposed label set. Then we have the following lemma.

Lemma 1. *Given the sampled size s of the proposed label set, for any pattern $\boldsymbol{x}$ with its correct label y and any label set $\bar{Y}$ with size s (i.e., $|\bar{Y}| = s$), the following equality holds:*

$$p(y \in \bar{Y} \mid \boldsymbol{x}, s) = \frac{s}{k}. \tag{6}$$

The proof is provided in Appendix A.2.

Theorem 2. *In the above setting, the distribution of collected data where the correct label y ($y \in \mathcal{Y}$) is not included in the label set $\bar{Y}$ ($\bar{Y} \in \bar{\mathcal{Y}}$) is the same as Eq. (4), i.e.,*

$$p(\boldsymbol{x}, \bar{Y} \mid y \notin \bar{Y}) = \bar{p}(\boldsymbol{x}, \bar{Y}). \tag{7}$$

The proof is provided in Appendix A.3.

4. Learning with Multiple Complementary Labels

In this section, we first present two wrappers that enable us to use any ordinary complementary-label learning approach for learning with MCLs. Then, we present an unbiased risk estimator for learning with MCLs as a whole, and establish an estimation error bound.

4.1. Wrappers

Since ordinary complementary-label learning approaches cannot directly deal with MCLs, it would be natural to ask whether there exist some strategies that can enable us to use any existing complementary-label learning approach for learning with MCLs.

Motivated by this, we propose two wrappers that decompose each example with MCLs into multiple examples, each with a single CL. Specifically, suppose a training example with MCLs is given as $(\boldsymbol{x}_i, \bar{Y}_i)$ where $\bar{Y}_i = \{\bar{y}_1, \bar{y}_2\}$. Then ordinary complementary label learning approaches may learn from $(\boldsymbol{x}_i, \bar{y}_1)$ and $(\boldsymbol{x}_i, \bar{y}_2)$. According to whether decomposition is after shuffling the training set, there are two decomposition strategies (wrappers) when we optimize a loss function by a stochastic optimization algorithm:

Decomposition after Shuffle. Given the shuffled training set with MCLs, in each mini-batch, we decompose each example into multiple examples, each with a single CL.

Decomposition before Shuffle. Given the training set with MCLs, we drive a new training set by decomposing each example into multiple examples, each with a single CL. Then, we shuffle the derived training set.

Both the above decomposition strategies enable us to use arbitrary ordinary complementary-label learning approaches for learning with MCLs. However, the derived training data with many single CLs may not match the originally assumed data distribution (i.e., Eq. (2)) for complementary-label learning, since these CLs are completely derived from

Table 1. Supervision information for a set of MCLs (with size s).

Setting	#TP	#FP	Supervision Purity
Many single CLs	s	$(k-2)s$	$1/(k-1)$
A set of MCLs	1	$k-s-1$	$1/(k-s)$

MCLs while the data distribution with MCLs is relevant to the size of each set of MCLs. As a consequence, the learning consistency would no longer be guaranteed even if the complementary-label learning approach inside the wrappers is originally risk-consistent or classifier-consistent.

Moreover, since ordinary complementary-label learning approaches can only learn with a single CL for each example at a time and treat each example independently, the supervision information for each set of MCLs would be conceptually diluted. We demonstrate this issue by Table 1. As shown in Table 1, there are two settings according to whether to decompose a set of MCLs into many single CLs or not. Since all the non-complementary labels have the possibility to be the correct label, we specially count how many times the correct label serves as a non-complementary label (denoted by #TP), and how many times the other labels except the correct label serve as a non-complementary label (denoted by #FP). Then the supervision purity is calculated by (#TP)/(#TP+#FP).

Clearly, the wrappers follow the setting where a set of MCLs is decomposed into many single CLs. If the size of the set of MCLs is s, then #TP equals s, since the correct label would serve as a non-complementary label for s times after decomposition, and the other labels except the correct label would serve as a non-complementary label for $(k-s-1)s+s(s-1)=(k-2)s$ times, hence the supervision purity would be $s/(s+(k-2)s)=1/(k-1)$. However, for the setting where the set of MCLs is not decomposed, we can easily know that the correct label serves as a non-complementary label once, and the other labels expect the correct label serve as a non-complementary label for $k-s-1$ times, hence the supervision purity is $1/(k-s)$. These observations clearly show that the supervision information is diluted after decomposing MCLs ($s \geqslant 2$), which also motivate us to take a set of MCLs as a whole set. In the following, we will introduce our proposed unbiased risk estimator, which is able to learn with MCLs as a whole.

4.2. Unbiased Risk Estimator

The above example has shown that the supervision information is diluted after decomposition. The basic reason lies in that ordinary complementary-label learning approaches are designed by only considering the data distribution with a single CL, i.e., $\overline{p}(\boldsymbol{x},\overline{y})$. However, the data distribution with MCLs $\overline{p}(\boldsymbol{x},\bar{Y})$ becomes much different, and the wrappers fail to capture such distribution because they do not treat MCLs as a whole for each example. To solve this problem, we propose an unbiased estimator of the original classification risk for learning with MCLs as a whole.

We first relate the data distribution with ordinary labels to that with MCLs by the following lemma.

Lemma 2. *The following equality holds:*

$$p(\boldsymbol{x},y)=1-\sum_{j=1}^{k-1}\Big(\frac{k-1}{j}\sum\nolimits_{\bar{Y}\in\bar{\mathcal{Y}}_j^y}\overline{p}(\boldsymbol{x},\bar{Y},s=j)\Big),$$

where $\bar{\mathcal{Y}}_j^y$ is the set of all the possible label sets with size j that include a specific label $y\in\mathcal{Y}$, i.e.,

$$\bar{\mathcal{Y}}_j^y:=\{\bar{Y}\in\bar{\mathcal{Y}}\mid y\in\bar{Y},|\bar{Y}|=j\}.$$

The proof is provided in Appendix B.1.

Based on Lemma 2, we derive an unbiased estimator of the ordinary classification risk Eq. (1) by the following theorem.

Theorem 3. *The ordinary classification risk Eq. (1) can be equivalently expressed as*

$$R(f)=\sum_{j=1}^{k-1}p(s=j)\bar{R}_j(f), \tag{8}$$

where

$$\bar{R}_j(f):=\mathbb{E}_{\overline{p}(\boldsymbol{x},\bar{Y}|s=j)}[\bar{\mathcal{L}}_j\big(f(\boldsymbol{x}),\bar{Y}\big)], \tag{9}$$

and

$$\bar{\mathcal{L}}_j\big(f(\boldsymbol{x}),\bar{Y}\big):=\sum\nolimits_{y\notin\bar{Y}}\mathcal{L}\big(f(\boldsymbol{x}),y\big)-\frac{k-1-j}{j}\sum\nolimits_{y'\in\bar{Y}}\mathcal{L}\big(f(\boldsymbol{x}),y'\big). \tag{10}$$

The proof is provided in Appendix B.2.

It is easy to verify that Eq. (8) reduces to Eq. (3) when $p(s=1)=1$. Which means, our approach is a generalization of Ishida et al. (2019). Furthermore, according to Corollary 2 in Ishida et al. (2019), our approach is also a generalization of Ishida et al. (2017).

Given the dataset with MCLs $\bar{\mathcal{D}}=\{(\boldsymbol{x}_i,\bar{Y}_i)\}_{i=1}^n$, we can empirically approximate $p(s=j)$ by n_j/n where n_j denotes the number of examples whose complementary label set size is j. By further taking into account Eqs. (8)-(10), we can obtain the following empirical approximation of the unbiased risk estimator introduced in Theorem 3:

$$\widehat{R}(f)=\frac{1}{n}\sum_{i=1}^{n}\Big(\sum\nolimits_{y\notin\bar{Y}_i}\mathcal{L}\big(f(\boldsymbol{x}_i),y\big)-\frac{k-1-|\bar{Y}_i|}{|\bar{Y}_i|}\sum\nolimits_{y'\in\bar{Y}_i}\mathcal{L}\big(f(\boldsymbol{x}_i),y'\big)\Big). \tag{11}$$

Estimation Error Bound. Here, we derive an estimation error bound for the proposed unbiased risk estimator

based on *Rademacher complexity* (Bartlett & Mendelson, 2002). Let $\mathcal{F} \subset \{f : \mathbb{R}^d \to \mathbb{R}^k\}$ be the hypothesis class, $\widehat{f} := \arg\min_{f\in\mathcal{F}} \widehat{R}(f)$ be the empirical risk minimizer, and $f^\star = \arg\min_{f\in\mathcal{F}} R(f)$ be the true risk minimizer. Besides, we define the functional space $\mathcal{G}_y$ for the label $y \in \mathcal{Y}$ as $\mathcal{G}_y = \{g : \boldsymbol{x} \to f_y(\boldsymbol{x}) \mid f \in \mathcal{F}\}$. Then, we have the following theorem.

Theorem 4. *Assume the loss function $\mathcal{L}(f(\boldsymbol{x}), y)$ is ρ-Lipschitz with respect to $f(\boldsymbol{x})$ $(0 < \rho < \infty)$ for all $y \in \mathcal{Y}$. Let $C_{\mathcal{L}} = \sup_{\boldsymbol{x}\in\mathcal{X}, f\in\mathcal{F}, y\in\mathcal{Y}} \mathcal{L}(f(\boldsymbol{x}), y)$ and $\mathfrak{R}_n(\mathcal{G}_y)$ be the Rademacher complexity of $\mathcal{G}_y$ given the sample size n. Then, for any $\delta > 0$, with probability at least $1-\delta$,*

$$R(\widehat{f}) - R(f^\star) \leqslant \sum_{j=1}^{k-1} p(s=j)\Big(\frac{4\sqrt{2}\rho k(k-1)}{j}\sum_{y=1}^{k}\mathfrak{R}_{n_j}(\mathcal{G}_y) + \frac{C_j}{\sqrt{n_j}}\Big),$$

where $C_j = (4k-4j-2)C_{\mathcal{L}}\sqrt{\frac{\log\frac{2(k-1)}{\delta}}{2}}$ for all $j \in \{1,\ldots,k-1\}$ and n_j denotes the number of examples whose complementary label set size is j.

The definition of Rademacher complexity and the proof of Theorem 4 are provided in Appendix C. Theorem 4 shows that the empirical risk minimizer converges to the true risk minimizer with high probability as the number of training data approaches infinity. It is worth noting that this bound is not only related to the Redemacher complexity of the function class, but also s and k. This observation accords with our intuition that the learning task will be harder if the number of classes k increases or the size of the complementary label set s decreases.

4.3. Practical Implementation

In this section, we present the practical implementation of our proposed formulation and improvements of the used loss functions. As described above, we have provided a general unbiased risk estimator that is able to use arbitrary loss functions. There arises a question: Can all loss functions work well in our approach? Unfortunately, the answer is negative.

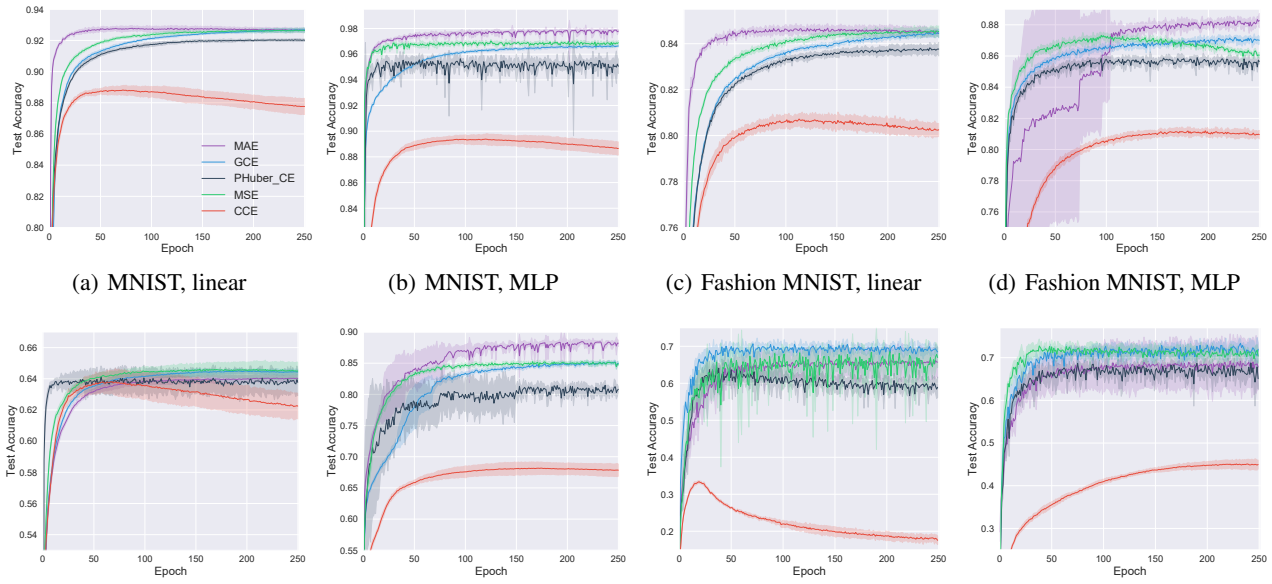
The original classification risk estimator in Eq. (1) includes an expectation over a non-negative loss $\mathcal{L} : \mathbb{R}^k \times [k] \to \mathbb{R}_+$, hence the expected risk and the empirical approximation are both lower-bounded by zero. However, our proposed risk estimator in Theorem 3 contains a negative term. Although the expected risk estimator is unbiased, the empirical estimator may become unbounded below if the used loss function is unbounded, thereby leading to over-fitting. Similar issues have also been shown by Ishida et al. (2019); Kiryo et al. (2017). The above analysis suggests that a bounded loss is probably better than an unbounded loss, in our empirical risk estimator (i.e., Eq. (11)).

To demonstrate the above conjecture, we would like to insert bounded and unbounded losses into Eq. (11), for comparison studies. Note that we assume that the softmax function is absorbed in each loss, and denote by $p_{\boldsymbol{\theta}}(y|\boldsymbol{x}) = \exp(f_y(\boldsymbol{x}))/(\sum_{j=1}^{k}\exp(f_j(\boldsymbol{x})))$ the predicted probability of the instance $\boldsymbol{x}$ belonging to class y, where $\boldsymbol{\theta}$ denotes the parameters of the model f. In this way, we list the compared loss functions as follows.

- Categorical Cross Entropy (CCE):
$$\mathcal{L}_{\text{CCE}}(f(\boldsymbol{x}), y) = -\log p_{\boldsymbol{\theta}}(y|\boldsymbol{x}).$$
- Mean Absolute Error (MAE):
$$\mathcal{L}_{\text{MAE}}(f(\boldsymbol{x}), y) = 2 - 2p_{\boldsymbol{\theta}}(y|\boldsymbol{x}).$$
- Mean Square Error (MSE):
$$\mathcal{L}_{\text{MSE}}(f(\boldsymbol{x}), y) = 1 - 2p_{\boldsymbol{\theta}}(y|\boldsymbol{x}) + \sum_{j=1}^{k} p_{\boldsymbol{\theta}}(j|\boldsymbol{x})^2.$$
- Generalized Cross Entropy (GCE) (Zhang & Sabuncu, 2018):
$$\mathcal{L}_{\text{GCE}}(f(\boldsymbol{x}), y) = (1 - p_{\boldsymbol{\theta}}(y|\boldsymbol{x})^q)/q,$$
where $q \in (0, 1]$ is a user-defined hyper-parameter. We set $q = 0.7$, as suggested by Zhang & Sabuncu (2018).
- Partially Huberised Cross Entropy (PHuber-CE) (Menon et al., 2020):
$$\mathcal{L}_{\text{PHuber-CE}}(f(\boldsymbol{x}), y) = \begin{cases} -\log p_{\boldsymbol{\theta}}(y|\boldsymbol{x}), \text{if } p_{\boldsymbol{\theta}}(y|\boldsymbol{x}) \geqslant \frac{1}{\tau}, \\ -\tau p_{\boldsymbol{\theta}}(y|\boldsymbol{x}) + \log\tau + 1, \text{ else}, \end{cases}$$
where $\tau > 0$ is a user-defined hyper-parameter. We set $\tau = 10$, because it works well in Menon et al. (2020).

The detailed derivations of the above loss functions and their bounds are provided in Appendix D. Among these losses, CCE is unbounded while the others are bounded. We will experimentally demonstrate (Figure 1) that by inserting the above losses into Eq. (11), bounded loss is significantly better than unbounded loss. Furthermore, we conduct a deeper analysis of MAE because MAE has the special property that MAE is not only bounded, but also satisfies the symmetric condition (Ghosh et al., 2017), i.e., $\sum_{y=1}^{k}\mathcal{L}_{\text{MAE}}\big(f(\boldsymbol{x}), y\big) = 2k-2$, which means the sum of the losses over all classes is a constant for arbitrary examples. However, is MAE good enough? Previous studies (Wang et al., 2019; Zhang & Sabuncu, 2018) have already shown that MAE suffers from the optimization issue, which would affect its practical performance. To alleviate this problem, we further improve MAE by proposing two upper-bound surrogate loss functions. Specifically, by using MAE in Eq. (11), we obtain

$$\begin{aligned}\widehat{R}(f) =& \frac{k-1}{|\bar{Y}_i|}\sum\nolimits_{y\notin\bar{Y}_i}\mathcal{L}_{\text{MAE}}\big(f(\boldsymbol{x}_i), y\big) \\ =& \frac{2k-2}{|\bar{Y}_i|}\mathcal{L}'_{\text{MAE}}\big(f(\boldsymbol{x}_i), \bar{Y}_i\big) + Z_i, \end{aligned} \tag{12}$$



(a) MNIST, linear (b) MNIST, MLP (c) Fashion MNIST, linear (d) Fashion MNIST, MLP

(e) Kuzushiji MNIST, linear (f) Kuzushiji MNIST, MLP (g) CIFAR-10, ResNet (h) CIFAR-10, DenseNet

Figure 1. Experimental results of different loss functions for different datasets and models. Dark colors show the mean accuracy of 5 trials and light colors show the standard deviation.

where $\mathcal{L}'_{\mathrm{MAE}}\big(f(\boldsymbol{x}_i), \bar{Y}_i\big) := 1 - \sum_{j \notin \bar{Y}_i} p_{\boldsymbol{\theta}}(j|\boldsymbol{x}_i)$, and Z_i is a constant independent of $f(\boldsymbol{x}_i)$. It is clear that minimizing $\mathcal{L}'_{\mathrm{MAE}}\big(f(\boldsymbol{x}_i), \bar{Y}_i\big)$ is equivalent to minimizing $\sum_{y \notin \bar{Y}_i} \mathcal{L}_{\mathrm{MAE}}\big(f(\boldsymbol{x}, y)\big)$.

Based on this fact, we further introduce two upper-bound surrogate loss functions of $\mathcal{L}'_{\mathrm{MAE}}$:

$$\mathcal{L}_{\mathrm{EXP}}(f(\boldsymbol{x}_i), \bar{Y}_i) = \exp\Big(-\sum\nolimits_{j \notin \bar{Y}_i} p_{\boldsymbol{\theta}}(j|\boldsymbol{x}_i)\Big),$$
$$\mathcal{L}_{\mathrm{LOG}}(f(\boldsymbol{x}_i), \bar{Y}_i) = -\log\Big(\sum\nolimits_{j \notin \bar{Y}_i} p_{\boldsymbol{\theta}}(j|\boldsymbol{x}_i)\Big).$$

One can easily verify that $\mathcal{L}'_{\mathrm{MAE}}$ is upper bounded by $\mathcal{L}_{\mathrm{EXP}}$ and $\mathcal{L}_{\mathrm{LOG}}$ using the two inequalities $1 - z \leqslant \exp(-z)$ and $1 - z \leqslant -\log z$, respectively. By replacing $\mathcal{L}'_{\mathrm{MAE}}$ by $\mathcal{L}_{\mathrm{LOG}}$ and $\mathcal{L}_{\mathrm{LOG}}$ in Eq. (12), we obtain two new methods for learning with MCLs. We explain the advantage of $\mathcal{L}_{\mathrm{EXP}}$ and $\mathcal{L}_{\mathrm{LOG}}$ over $\mathcal{L}'_{\mathrm{MAE}}$ by closely examining their gradients:

$$\frac{\partial \mathcal{L}'_{\mathrm{MAE}}}{\partial \boldsymbol{\theta}} = \begin{cases} -\nabla_{\boldsymbol{\theta}} p_{\boldsymbol{\theta}}(j|\boldsymbol{x}_i), & \text{if } j \notin \bar{Y}_i, \\ 0, & \text{else}, \end{cases}$$
$$\frac{\partial \mathcal{L}_{\mathrm{EXP}}}{\partial \boldsymbol{\theta}} = \begin{cases} -\nabla_{\boldsymbol{\theta}} p_{\boldsymbol{\theta}}(j|\boldsymbol{x}_i) \cdot w_{\mathrm{EXP}}, & \text{if } j \notin \bar{Y}_i, \\ 0, & \text{else}, \end{cases}$$
$$\frac{\partial \mathcal{L}_{\mathrm{LOG}}}{\partial \boldsymbol{\theta}} = \begin{cases} -\nabla_{\boldsymbol{\theta}} p_{\boldsymbol{\theta}}(j|\boldsymbol{x}_i) \cdot w_{\mathrm{LOG}}, & \text{if } j \notin \bar{Y}_i, \\ 0, & \text{else}, \end{cases}$$

where $w_{\mathrm{EXP}} = \exp\big(-\sum_{j \notin \bar{Y}_i} p_{\boldsymbol{\theta}}(j|\boldsymbol{x}_i)\big)$ and $w_{\mathrm{LOG}} = \big(\sum_{j \notin \bar{Y}_i} p_{\boldsymbol{\theta}}(j|\boldsymbol{x}_i)\big)^{-1}$. From their gradients, we can clearly observe that $\mathcal{L}'_{\mathrm{MAE}}$ basically treats each example equally, while $\mathcal{L}_{\mathrm{EXP}}$ and $\mathcal{L}_{\mathrm{LOG}}$ give more weights to difficult examples. Concretely, if $\sum_{j \notin \bar{Y}_i} p_{\boldsymbol{\theta}}(j|\boldsymbol{x}_i)$ is small, both w_{EXP} and w_{LOG} would be large. In other words, $\mathcal{L}_{\mathrm{EXP}}$ and $\mathcal{L}_{\mathrm{LOG}}$ pay more attention to hard examples whose sum of the predicted confidences of all the non-complementary labels is small.

5. Experiments

In this section, we conduct extensive experiments to evaluate the performance of our proposed approaches including the two wrappers, the unbiased risk estimator with various loss functions and the two upper-bound surrogate loss functions.

Datasets. We use five widely-used benchmark datasets MNIST (LeCun et al., 1998), Kuzushiji-MNIST (Clanuwat et al., 2018), Fashion-MNIST (Xiao et al., 2017), 20Newsgroups (Lang, 1995), and CIFAR-10 (Krizhevsky et al., 2009), and four datasets from the UCI repository (Blake & Merz, 1998). We use four base models including linear model, MLP model (d-500-k), ResNet (34 layers) (He et al., 2016), and DenseNet (22 layers) (Huang et al., 2017). The detailed descriptions of these datasets with the corresponding base models are provided in Appendix E.1. To generate MCLs, we instantiate $p(s) = \binom{k-1}{s}/(2^k - 2)$, $\forall s \in \{1, \cdots, k-1\}$, which means $p(s)$ represents the ratio of the number of label sets whose size is s to the number of all possible label sets. For each instance $\boldsymbol{x}$, we first randomly sample s from $p(s)$, and then uniformly and randomly sample a complementary label set $\bar{Y}$ with size s (i.e., $p(\bar{Y}) = 1/\binom{k-1}{s}$).

Approaches. We absorb five ordinary complementary-label learning approaches in the two wrappers (introduced

Table 2. Classification accuracy (mean±std) of each algorithm on the four UCI datasets using a linear model for 5 trials. The best performance among all the approaches is highlighted in boldface. In addition, •/◦ indicates whether the performance of our approach (the best of EXP and LOG) is statistically superior/inferior to the comparing algorithm on each dataset (paired t-test at 0.05 significance level).

Approach		Yeast	Texture	Dermatology	Synthetic Control
Upper-bound Losses	EXP	54.94±1.56%•	97.51±0.09%•	98.89±0.37%	27.87±5.13%•
	LOG	**60.11±1.93%**	98.88±0.43%	**99.46±1.14%**	**90.73±4.41%**
Bounded Losses	MAE	33.07±0.37%•	85.29±7.93%•	85.39±2.58%•	23.50±2.44%•
	MSE	58.17±1.52%•	97.59±0.16%•	97.84±1.21%•	34.20±8.69%•
	GCE	57.56±1.56%•	97.25±0.31%•	97.53±1.81%•	23.67±3.10%•
	Phuber-CE	55.54±1.03%•	94.89±3.28%•	95.14±2.41%•	24.71±3.18%•
Unbounded Loss	CCE	49.50±3.58%•	92.08±1.15%•	83.19±3.65%•	63.47±6.91%•
Decomposition before Shuffle	GA	27.91±5.02%•	90.93±1.34%•	36.05±9.79%•	18.12±1.74%•
	NN	32.73±3.59%•	96.29±0.39%•	61.49±6.83%•	55.12±4.43%•
	FREE	35.50±2.79%•	94.36±1.08%•	86.30±5.62%•	76.95±3.26%•
	PC	53.89±3.53%•	92.68±0.81%•	96.27±3.07%•	72.63±5.86%•
	Forward	58.15±1.54%•	**98.95±0.17%**	99.37±0.85%	38.77±6.06%•
Decomposition after Shuffle	GA	28.21±1.53%•	83.66±2.27%•	42.05±7.94%•	25.46±1.28%•
	NN	36.04±2.24%•	93.91±0.40%•	62.54±9.19%•	59.80±5.14%•
	FREE	43.47±1.36%•	93.94±0.72%•	86.22±6.07%•	73.33±2.17%•
	PC	54.58±2.57%•	94.19±1.21%•	95.73±3.33%•	69.53±9.01%•
	Forward	59.46±1.16%	97.65±0.32%•	99.03±1.33%	43.57±5.83%•
Partial Label Convex Formulation	CLPL	55.39±1.21%•	92.07±0.88%•	99.42±0.54%	63.57±5.46%•

Table 3. Classification accuracy (mean±std) of each algorithm on the four benchmark datasets using a linear model for 5 trials. The best performance among all the approaches is highlighted in boldface. In addition, •/◦ indicates whether the performance of our approach (the best of EXP and LOG) is statistically superior/inferior to the comparing algorithm on each dataset (paired t-test at 0.05 significance level).

Approach		MNIST	Kuzushiji	Fashion	20Newsgroups
Upper-bound Losses	EXP	**92.67±0.11%**	64.23±0.33%•	**84.56±0.25%**	81.72±0.39%•
	LOG	92.58±0.09%•	**68.89±0.25%**	84.42±0.16%	**84.06±0.57%**
Bounded Losses	MAE	92.66±0.12%	64.03±0.19%•	84.50±0.16%	79.68±1.40%•
	MSE	92.64±0.13%	64.51±0.55%•	84.53±0.20%	81.55±0.52%•
	GCE	92.66±0.12%	64.44±0.17%•	84.44±0.15%	81.78±0.60%•
	Phuber-CE	92.02±0.07%•	63.81±0.75%•	83.76±0.22%•	73.52±1.04%•
Unbounded Loss	CCE	88.23±0.19%•	62.27±0.84%•	80.25±0.29%•	63.78±0.79%•
Decomposition before Shuffle	GA	85.51±0.26%•	55.61±0.24%•	78.64±0.33%•	76.64±0.62%•
	NN	88.09±0.16%•	60.54±0.23%•	80.68±0.07%•	76.00±0.37%•
	FREE	89.35±0.14%•	65.21±0.45%•	81.22±0.11%•	68.34±0.72%•
	PC	88.21±0.23%•	62.76±0.40%•	80.60±0.18%•	66.91±1.20%•
	Forward	92.57±0.05%•	63.51±0.22%•	84.38±0.20%	74.69±1.14%•
Decomposition after Shuffle	GA	83.16±0.22%•	56.31±0.42%•	73.37±0.10%•	66.14±0.79%•
	NN	88.79±0.26%•	63.19±0.12%•	79.77±0.14%•	66.35±0.53%•
	FREE	89.02±0.22%•	64.18±0.18%•	80.11±0.04%•	66.16±0.60%•
	PC	87.76±0.17%•	61.64±0.38%•	80.58±0.17%•	65.64±0.81%•
	Forward	92.54±0.04%•	63.69±0.14%•	84.37±0.17%•	71.98±3.41%•
Partial Label Convex Formulation	CLPL	81.85±0.27%•	55.31±0.23%•	77.26±0.10%•	81.48±0.45%•

in Section 4.1): GA, NN, and Free (Ishida et al., 2019), PC (Ishida et al., 2017), and Forward (Yu et al., 2018).We also use an unbounded loss CCE and four bounded losses MAE, MSE, GCE (Zhang & Sabuncu, 2018), and PHuber-CE (Menon et al., 2020) in our empirical estimator Eq. (11). Besides, two upper-bound loss functions LOG and EXP are also inserted into Eq. (12). In addition, we also compare with a representative partial label learning approach CLPL (Cour et al., 2011). For all the approaches, we adopt the same base model for fair comparison. Learning rate and weight decay are selected from $\{10^{-6}, 10^{-5}, \cdots, 10^{-1}\}$. We implement our approach using PyTorch[1], and use the Adam (Kingma & Ba, 2015) optimization method with mini-batch size set to 256 and epoch number set to 250. Hyper-parameters for all the approaches are selected so as to maximize the accuracy on a validation set (10% of the training set) of complementarily labeled data. All the experiments are conducted on NVIDIA Tesla V100 GPUs.

[1] www.pytorch.org

Table 4. Classification accuracy (mean±std) of each algorithm on the five benchmark datasets using neural networks for 5 trials. The best performance among all the approaches is highlighted in boldface. In addition, •/◦ indicates whether the performance of our approach (the best of EXP and LOG) is statistically superior/inferior to the comparing algorithm on each dataset (paired t-test at 0.05 significance level).

Approach		MNIST	Kuzushiji	Fashion	CIFAR-10 R	CIFAR-10 D	20Newsgroups
Upper-bound Losses	EXP	97.80±0.06%	**88.25±0.28%**	88.07±0.19%•	72.49±0.84%•	75.53±0.58%	77.22±1.22%
	LOG	**97.86±0.13%**	88.24±0.08%	**88.36±0.26%**	**75.38±0.34%**	**75.80±0.62%**	**79.46±0.94%**
Bounded Losses	MAE	97.81±0.04%	88.11±0.40%	88.13±0.23%	65.57±4.08%•	68.24±5.84%•	49.83±4.01%•
	MSE	96.84±0.08%•	84.97±0.23%•	86.14±0.04%•	63.58±1.19%•	70.89±0.81%•	72.19±0.59%•
	GCE	96.62±0.08%•	85.02±0.26%•	87.03±0.20%•	68.40±1.05%•	71.54±0.83%•	74.96±0.47%•
	Phuber-CE	95.00±0.36%•	80.66±0.32%•	85.52±0.18%•	59.64±1.21%•	66.49±0.67%•	62.63±2.32%•
Unbounded Loss	CCE	88.64±0.50%•	67.86±1.01%•	80.97±0.23%•	18.01±0.63%•	44.94±1.20%•	54.96±0.38%•
Decomposition before Shuffle	GA	96.36±0.05%•	84.35±0.22%•	85.59±0.30%•	69.05±0.83%•	65.38±1.40%•	79.06±0.57%
	NN	96.70±0.08%•	82.21±0.36%•	86.29±0.10%•	63.85±0.74%•	64.80±1.28%•	76.81±0.44%•
	FREE	88.55±0.38%•	70.32±0.80%•	81.17±0.36%•	32.02±1.69%•	39.22±0.43%•	61.22±1.24%•
	PC	92.74±0.17%•	73.18±0.59%•	83.32±0.28%•	43.16±2.21%•	49.53±1.18%•	65.15±2.05%•
	Forward	97.67±0.04%•	87.65±0.24%•	88.08±0.24%•	71.92±1.09%•	71.30±1.16%•	77.19±%0.76•
Decomposition after Shuffle	GA	92.08±0.22%•	74.64±0.67%•	79.73±0.19%•	53.12±0.97%•	56.51±0.89%•	63.37±1.16%•
	NN	92.47±0.14%•	73.88±0.63%•	82.99±0.13%•	36.79±0.78%•	53.78±0.92%•	65.15±0.73%•
	FREE	88.99±0.39%•	70.09±0.74%•	81.74±0.23%•	15.16±2.22%•	47.45±0.98%•	50.86±1.56%•
	PC	92.94±0.05%•	68.60±1.32%•	82.46±0.26%•	33.16±0.92%•	52.23±0.88%•	64.32±0.86%•
	Forward	97.49±0.08%•	86.47±0.39%•	87.56±0.14%•	72.16±0.97%•	75.23±1.02%	79.35±0.82%

Loss Comparison. Figure 1 shows the mean and standard deviation of test accuracy of 5 trials, for bounded loss functions MAE, MSE, GCE, PHuber-CE, and unbounded loss function CCE used in our empirical risk estimator Eq. (11). We also record the mean and standard deviation of training accuracy (the training set is evaluated with ordinary labels) of 5 trials, and put the results in Appendix E.2. As can be seen from Figure 1, all the bounded losses are significantly better than the unbounded loss CCE in our formulation. This observation clearly accords with our discussion on the over-fitting issue in Section 4.3. In addition, MAE achieves comparable performance compared with other bounded losses in most cases, while it is sometimes inferior to other bounded losses due to its optimization issue (Zhang & Sabuncu, 2018). Both the advantage and disadvantage of MAE motivate us to use the upper-bound loss functions EXP and LOG for improving the classification performance.

Performance Comparison. Table 2, Table 3, and Table 4 show the experimental results of different approaches using a linear model or neural networks on the four UCI datasets and the other five benchmark datasets. In table 4, "CIFAR-10 R" and "CIFAR-10 D" mean that we use ResNet and DenseNet on CIFAR-10. Note that CLPL is a convex approach for partial label learning, which is specially designed with a linear model. Hence CLPL does not appear in Table 4. From the three tables, we can find that equipped with the two wrappers "Decomposition before Shuffle" and "Decomposition after Shuffle", ordinary complementary-label learning approaches work well for learning with MCLs. However, they are significantly outperformed by the upper-bound losses in most cases, which also achieve the best performance among all the approaches on various benchmark datasets. In addition, we also study the case where the size of each complementary label set s is fixed at j (i.e., $p(s = j) = 1$) while increasing j from 1 to $k - 2$. The corresponding experimental results are provided in Appendix E.3, which show that the classification accuracy of our approaches increases as j increases. This observation is clearly in accordance with our derived estimation error bound (Theorem 4), as the estimation error would decrease if j increases.

6. Conclusion

In this paper, we propose a novel problem setting called *learning with multiple complementary labels* (MCLs), which is a generation of *complementary-label learning* (Ishida et al., 2017; 2019; Yu et al., 2018). To solve this learning problem, we first design two wrappers that enable us to use arbitrary complementary-label learning approaches for learning with MCLs. However, we find that the supervision information that MCLs hold is conceptually diluted after decomposition. Therefore, we further propose an unbiased risk estimator for learning with MCLs, which processes each set of MCLs as a whole. Then, we theoretically derive an estimation error bound, which guarantees the learning consistency. Although our risk estimator does not rely on specific models or loss functions, we show that bounded loss is generally better than unbounded loss in our empirical risk estimator. In addition, we improve the risk estimator into minimizing properly chosen upper bounds for practical implementation. Extensive experiments demonstrate the effectiveness of the proposed approaches.

Acknowledgements

This research was supported by the National Research Foundation, Singapore under its AI Singapore Programme (AISG Award No: AISG-RP-2019-0013), National Satellite of Excellence in Trustworthy Software Systems (Award No: NSOE-TSS2019-01), and NTU. BH was partially supported by the Early Career Scheme (ECS) through the Research Grants Council of Hong Kong under Grant No.22200720, HKBU Tier-1 Start-up Grant and HKBU CSD Start-up Grant. GN and MS were supported by JST AIP Acceleration Research Grant Number JPMJCR20U3, Japan.

References

Bao, H., Niu, G., and Sugiyama, M. Classification from pairwise similarity and unlabeled data. In *ICML*, pp. 452–461, 2018.

Bartlett, P. L. and Mendelson, S. Rademacher and gaussian complexities: Risk bounds and structural results. *JMLR*, 3(11):463–482, 2002.

Blake, C. L. and Merz, C. J. Uci repository of machine learning databases, 1998. URL http://archive.ics.uci.edu/ml/index.php.

Chapelle, O., Scholkopf, B., and Zien, A. *Semi-Supervised Learning*. MIT Press, 2006.

Clanuwat, T., Bober-Irizar, M., Kitamoto, A., Lamb, A., Yamamoto, K., and Ha, D. Deep learning for classical japanese literature. *arXiv preprint arXiv:1812.01718*, 2018.

Cour, T., Sapp, B., and Taskar, B. Learning from partial labels. *JMLR*, 12(5):1501–1536, 2011.

du Plessis, M. C., Niu, G., and Sugiyama, M. Analysis of learning from positive and unlabeled data. In *NeurIPS*, pp. 703–711, 2014.

du Plessis, M. C., Niu, G., and Sugiyama, M. Convex formulation for learning from positive and unlabeled data. In *ICML*, pp. 1386–1394, 2015.

Elkan, C. and Noto, K. Learning classifiers from only positive and unlabeled data. In *KDD*, pp. 213–220, 2008.

Feng, L. and An, B. Leveraging latent label distributions for partial label learning. In *IJCAI*, pp. 2107–2113, 2018.

Feng, L. and An, B. Partial label learning with self-guided retraining. In *AAAI*, pp. 3542–3549, 2019a.

Feng, L. and An, B. Partial label learning by semantic difference maximization. In *IJCAI*, pp. 2294–2300, 2019b.

Ghosh, A., Kumar, H., and Sastry, P. Robust loss functions under label noise for deep neural networks. In *AAAI*, 2017.

Halko, N., Martinsson, P.-G., and Tropp, J. A. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Review*, 53(2):217–288, 2011.

Han, B., Yao, J., Niu, G., Zhou, M., Tsang, I., Zhang, Y., and Sugiyama, M. Masking: A new perspective of noisy supervision. In *NeurIPS*, pp. 5836–5846, 2018a.

Han, B., Yao, Q., Yu, X., Niu, G., Xu, M., Hu, W., Tsang, I., and Sugiyama, M. Co-teaching: Robust training of deep neural networks with extremely noisy labels. In *NeurIPS*, pp. 8527–8537, 2018b.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *CVPR*, pp. 770–778, 2016.

Huang, G., Liu, Z., Van Der Maaten, L., and Weinberger, K. Q. Densely connected convolutional networks. In *CVPR*, pp. 4700–4708, 2017.

Ishida, T., Niu, G., Hu, W., and Sugiyama, M. Learning from complementary labels. In *NeurIPS*, pp. 5644–5654, 2017.

Ishida, T., Niu, G., and Sugiyama, M. Binary classification for positive-confidence data. In *NeurIPS*, pp. 5917–5928, 2018.

Ishida, T., Niu, G., Menon, A. K., and Sugiyama, M. Complementary-label learning for arbitrary losses and models. In *ICML*, pp. 2971–2980, 2019.

Kaneko, T., Sato, I., and Sugiyama, M. Online multiclass classification based on prediction margin for partial feedback. *arXiv preprint arXiv:1902.01056*, 2019.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. In *ICLR*, 2015.

Kiryo, R., Niu, G., du Plessis, M. C., and Sugiyama, M. Positive-unlabeled learning with non-negative risk estimator. In *NeurIPS*, pp. 1674–1684, 2017.

Krizhevsky, A., Hinton, G., et al. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009.

Lang, K. Newsweeder: Learning to filter netnews. In *ICML*, 1995.

LeCun, Y., Bottou, L., Bengio, Y., Haffner, P., et al. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Li, Y.-F. and Liang, D.-M. Safe semi-supervised learning: a brief introduction. *Frontiers of Computer Science*, 13(4): 669–676, 2019.

Lu, N., Niu, G., Menon, A. K., and Sugiyama, M. On the minimal supervision for training any binary classifier from only unlabeled data. In *ICLR*, 2019.

Lu, N., Zhang, T., Niu, G., and Sugiyama, M. Mitigating overfitting in supervised classification from two unlabeled datasets: A consistent risk correction approach. In *AISTATS*, 2020.

Maurer, A. A vector-contraction inequality for rademacher complexities. In *ALT*, pp. 3–17, 2016.

McDiarmid, C. On the method of bounded differences. In *Surveys in Combinatorics*, 1989.

Menon, A., Van Rooyen, B., Ong, C. S., and Williamson, B. Learning from corrupted binary labels via class-probability estimation. In *ICML*, pp. 125–134, 2015.

Menon, A. K., Rawat, A. S., Reddi, S. J., and Kumar, S. Can gradient clipping mitigate label noise? In *ICLR*, 2020.

Miyato, T., Maeda, S.-i., Koyama, M., and Ishii, S. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *TPAMI*, 41(8): 1979–1993, 2018.

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

Niu, G., Jitkrittum, W., Dai, B., Hachiya, H., and Sugiyama, M. Squared-loss mutual information regularization: A novel information-theoretic approach to semi-supervised learning. In *ICML*, pp. 10–18, 2013.

Rezaei, M., Yang, H., and Meinel, C. Recurrent generative adversarial network for learning imbalanced medical image semantic segmentation. *Multimedia Tools and Applications*, pp. 1–20, 2019.

Sakai, T., du Plessis, M. C., Niu, G., and Sugiyama, M. Semi-supervised classification based on classification from positive and unlabeled data. In *ICML*, pp. 2998–3006, 2017.

Sakai, T., Niu, G., and Sugiyama, M. Semi-supervised auc optimization based on positive-unlabeled learning. *MLJ*, 107(4):767–794, 2018.

Wang, X., Kodirov, E., Hua, Y., and Robertson, N. M. Improving mae against cce under label noise. *arXiv preprint arXiv:1903.12141*, 2019.

Wei, H., Feng, L., Chen, X., and An, B. Combating noisy labels by agreement: A joint training method with co-regularization. In *CVPR*, June 2020.

Xia, X., Liu, T., Wang, N., Han, B., Gong, C., Niu, G., and Sugiyama, M. Are anchor points really indispensable in label-noise learning? In *NeurIPS*, pp. 6835–6846, 2019.

Xiao, H., Rasul, K., and Vollgraf, R. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

Yu, X., Liu, T., Gong, M., and Tao, D. Learning with biased complementary labels. In *ECCV*, pp. 68–83, 2018.

Zhang, M.-L. and Yu, F. Solving the partial label learning problem: An instance-based approach. In *IJCAI*, pp. 4048–4054, 2015.

Zhang, T. Statistical analysis of some multi-category large margin classification methods. *JMLR*, 5(10):1225–1251, 2004.

Zhang, Z. and Sabuncu, M. Generalized cross entropy loss for training deep neural networks with noisy labels. In *NeurIPS*, pp. 8778–8788, 2018.

Zhou, Z. A brief introduction to weakly supervised learning. *National Science Review*, 5(1):44–53, 2018.

Zhu, X. and Goldberg, A. B. Introduction to semi-supervised learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 3(1):1–130, 2009.

A. Proofs about the Problem Setting

A.1. Proofs of Theorem 1

Firstly, we define the set of all the possible label sets whose size is j as

$$\bar{\mathcal{Y}}_j := \{Y \mid Y \in \bar{\mathcal{Y}}, |Y| = j\}.$$

Then, by the definition of $\bar{p}(\boldsymbol{x}, \bar{Y})$, we can obtain

$$\begin{aligned}
\int_{\bar{\mathcal{Y}}} \int_{\mathcal{X}} \bar{p}(\boldsymbol{x}, \bar{Y}) \mathrm{d}\boldsymbol{x}\, \mathrm{d}\bar{Y} &= \int \textstyle\sum_{\bar{Y} \in \bar{\mathcal{Y}}} \bar{p}(\boldsymbol{x}, \bar{Y}) \mathrm{d}\boldsymbol{x} \\
&= \int \textstyle\sum_{\bar{Y} \in \bar{\mathcal{Y}}} \sum_{j=1}^{k-1} \Big(\bar{p}(\boldsymbol{x}, \bar{Y} \mid s = j) p(s = j) \Big) \mathrm{d}\boldsymbol{x} \\
&= \int \sum_{j=1}^{k-1} \textstyle\sum_{\bar{Y} \in \bar{\mathcal{Y}}_j} \Big(\bar{p}(\boldsymbol{x}, \bar{Y} \mid s = j) p(s = j) \Big) \mathrm{d}\boldsymbol{x} \qquad (\because \bar{\mathcal{Y}}_j := \{\bar{Y} \mid \bar{Y} \in \bar{\mathcal{Y}}, |\bar{Y}| = j\}) \\
&= \int \sum_{j=1}^{k-1} \textstyle\sum_{\bar{Y} \in \bar{\mathcal{Y}}_j} \left(\frac{1}{\binom{k-1}{j}} \sum_{y \notin \bar{Y}} p(\boldsymbol{x}, y) p(s = j) \right) \mathrm{d}\boldsymbol{x} \qquad (\because \text{the definition of } \bar{p}(\boldsymbol{x}, \bar{Y} \mid s = j)) \\
&= \int \sum_{j=1}^{k-1} \left(\frac{1}{\binom{k-1}{j}} \frac{\binom{k}{j}(k-j)}{k} \sum_{y=1}^{k} p(\boldsymbol{x}, y) p(s = j) \right) \mathrm{d}\boldsymbol{x} \qquad \left(\because |\bar{\mathcal{Y}}_j| = \tbinom{k}{j} \right) \\
&= \int \sum_{j=1}^{k-1} p(\boldsymbol{x}) p(s = j) \mathrm{d}\boldsymbol{x} \\
&= 1,
\end{aligned}$$

which concludes the proof of Theorem 1. □

A.2. Proof of Lemma 1

Let us consider the case where the correct label y is a specific label i ($i \in \{1, 2, \cdots, k\}$), then we have

$$\begin{aligned}
p(y \in \bar{Y}, y = i \mid \boldsymbol{x}, s) =& p(y \in \bar{Y} \mid y = i, \boldsymbol{x}, s) p(y = i \mid \boldsymbol{x}, s) \\
=& \textstyle\sum_{C \in \bar{\mathcal{Y}}} p(y \in \bar{Y}, \bar{Y} = C \mid y = i, \boldsymbol{x}, s) p(y = i \mid \boldsymbol{x}, s).
\end{aligned}$$

Here, $p(y = i \mid \boldsymbol{x}, s) = p(y = i \mid \boldsymbol{x})$ since the labeling rule is independent of s. In addition, $\sum_{C \in \bar{\mathcal{Y}}} p(y \in \bar{Y}, \bar{Y} = C \mid y = i, \boldsymbol{x}, s) = \sum_{C \in \bar{\mathcal{Y}}_s} p(y \in \bar{Y}, \bar{Y} = C \mid y = i, \boldsymbol{x})$ since given the size s of the label set, the whole set of all the possible label sets becomes $\bar{\mathcal{Y}}_s$. Then, we can obtain

$$\begin{aligned}
p(y \in \bar{Y}, y = i \mid \boldsymbol{x}, s) &= \textstyle\sum_{C \in \bar{\mathcal{Y}}} p(y \in \bar{Y}, \bar{Y} = C \mid y = i, \boldsymbol{x}, s) p(y = i \mid \boldsymbol{x}, s) \\
&= \textstyle\sum_{C \in \bar{\mathcal{Y}}_s} p(y \in \bar{Y}, \bar{Y} = C \mid y = i, \boldsymbol{x}) p(y = i \mid \boldsymbol{x}) \\
&= \textstyle\sum_{C \in \bar{\mathcal{Y}}_s} p(y \in \bar{Y} \mid \bar{Y} = C, y = i, \boldsymbol{x}) p(y = i \mid \boldsymbol{x}) p(\bar{Y} = C \mid \boldsymbol{x}) \\
&= \textstyle\sum_{C \in \bar{\mathcal{Y}}_s} p(y \in \bar{Y} | \bar{Y} = C, y = i, \boldsymbol{x}) p(y = i | \boldsymbol{x}) p(\bar{Y} = C),
\end{aligned}$$

where the last equality holds due to the fact that for each instance $\boldsymbol{x}$, $\bar{Y}$ is uniformly and randomly chosen. Since $p(\bar{Y}=C)=\frac{1}{|\bar{\mathcal{Y}}_s|}$ if $C\in\bar{\mathcal{Y}}_s$ where $|\bar{\mathcal{Y}}_s|=\binom{k}{s}$, we have

$$\begin{aligned}
p(y\in\bar{Y},y=i\mid\boldsymbol{x},s)=&\textstyle\sum_{C\in\bar{\mathcal{Y}}_s}p(y\in\bar{Y}|\bar{Y}=C,y=i,\boldsymbol{x})p(y=i|\boldsymbol{x})p(\bar{Y}=C)\\
=&\frac{1}{\binom{k}{s}}\textstyle\sum_{C\in\bar{\mathcal{Y}}_s}p(y\in\bar{Y}|\bar{Y}=C,y=i,\boldsymbol{x})p(y=i|\boldsymbol{x})\\
=&\frac{1}{\binom{k}{s}}|\bar{\mathcal{Y}}_s^i|p(y=i|\boldsymbol{x})\qquad\left(\because\bar{\mathcal{Y}}_s^i:=\{\bar{Y}\in\bar{\mathcal{Y}}_s\mid i\in\bar{Y}\}\right)\\
=&\frac{\binom{k-1}{s-1}}{\binom{k}{s}}p(y=i|\boldsymbol{x})\qquad\left(\because|\bar{\mathcal{Y}}_s^i|=\tbinom{k-1}{s-1}\right)\\
=&\frac{s}{k}p(y=i|\boldsymbol{x}).
\end{aligned}$$

By further summing up the both side over all the possible i, we can obtain

$$p(y\in\bar{Y}|\boldsymbol{x},s)=\frac{s}{k},$$

which concludes the proof of Lemma 1. □

A.3. Proof of Theorem 2

Let us express $p(\bar{Y}|y\notin\bar{Y},\boldsymbol{x},s)$ as

$$\begin{aligned}
p(\bar{Y}|y\notin\bar{Y},\boldsymbol{x},s)=&\frac{p(y\notin\bar{Y},\bar{Y}|\boldsymbol{x},s)}{p(y\notin\bar{Y}|\boldsymbol{x},s)}\\
=&\frac{p(y\notin\bar{Y}|\bar{Y},\boldsymbol{x},s)p(\bar{Y}|\boldsymbol{x},s)}{p(y\notin\bar{Y}|\boldsymbol{x},s)}\\
=&\frac{p(y\notin\bar{Y}|\bar{Y},\boldsymbol{x},s)p(\bar{Y}|s)}{p(y\notin\bar{Y}|\boldsymbol{x},s)},
\end{aligned}$$

where the last equality holds because $\bar{Y}$ is influenced by the size s, and for each instance $\boldsymbol{x}$, $\bar{Y}$ is uniformly and randomly chosen. Note that given s, there are $|\bar{\mathcal{Y}}_s|$ possible label sets, thus $p(\bar{Y}|s)=\frac{1}{|\bar{\mathcal{Y}}_s|}$ where $|\bar{\mathcal{Y}}_s|=\binom{k}{s}$. In this way, we have

$$\begin{aligned}
p(\bar{Y}|y\notin\bar{Y},\boldsymbol{x},s)=&\frac{p(y\notin\bar{Y}|\bar{Y},\boldsymbol{x},s)p(\bar{Y}|s)}{p(y\notin\bar{Y}|\boldsymbol{x},s)}\\
=&\frac{1}{\binom{k}{s}}\frac{p(y\notin\bar{Y}|\bar{Y},\boldsymbol{x},s)}{1-p(y\in\bar{Y}|\boldsymbol{x},s)}\\
=&\frac{1}{\binom{k}{s}}\frac{1}{1-\frac{s}{k}}p(y\notin\bar{Y}|\bar{Y},\boldsymbol{x},s)\qquad\left(\because\text{by Lemma 1, }p(y\in\bar{Y}|\boldsymbol{x},s)=\frac{s}{k}\right)\\
=&\frac{1}{\binom{k}{s}}\frac{k}{k-s}\textstyle\sum_{y\notin\bar{Y}}p(y|\boldsymbol{x},s)\\
=&\frac{1}{\binom{k-1}{s}}\textstyle\sum_{y\notin\bar{Y}}p(y|\boldsymbol{x}).
\end{aligned}$$

By multiplying $p(\boldsymbol{x})$ on both side, we have

$$p(\boldsymbol{x},\bar{Y}|y\notin\bar{Y},s)=\frac{1}{\binom{k-1}{s}}\textstyle\sum_{y\notin\bar{Y}}p(\boldsymbol{x},y).$$

Then taking into account the variable s, we have

$$\begin{aligned} p(\boldsymbol{x}, \bar{Y} | y \notin \bar{Y}) =& \sum_{j=1}^{k-1} p(s=j) p(\boldsymbol{x}, \bar{Y} | y \notin \bar{Y}, s=j) \\ =& \sum_{j=1}^{k-1} p(s=j) \frac{1}{\binom{k-1}{j}} \sum_{y \notin \bar{Y}} p(\boldsymbol{x}, y) \\ =& \bar{p}(\boldsymbol{x}, \bar{Y}), \end{aligned}$$

which concludes the proof. □

B. Proofs of the Unbiased Risk Estimator

B.1. Proof of Lemma 2

According to our defined distribution, we have already obtained

$$\bar{p}(\boldsymbol{x}, \bar{Y} \mid s=j) = \frac{1}{\binom{k-1}{j}} \sum_{y' \notin \bar{Y}} p(\boldsymbol{x}, y').$$

Then, we can obtain the following equality by operating $\sum_{\bar{Y} \in \bar{\mathcal{Y}}_j^y}$ on both the left and the right hand side:

$$\sum_{\bar{Y} \in \bar{\mathcal{Y}}_j^y} \bar{p}(\boldsymbol{x}, \bar{Y} \mid s=j) = \frac{1}{\binom{k-1}{j}} \sum_{\bar{Y} \in \bar{\mathcal{Y}}_j^y} \sum_{y' \notin \bar{Y}} p(\boldsymbol{x}, y'), \tag{13}$$

where $\bar{\mathcal{Y}}_j^y := \{\bar{Y} \in \bar{\mathcal{Y}} \mid y \in \bar{Y}, |\bar{Y}| = j\}$. In this way, the right hand side of the above equality can be transformed by the following derivations:

$$\begin{aligned} \frac{1}{\binom{k-1}{j}} \sum_{\bar{Y} \in \bar{\mathcal{Y}}_j^y} \sum_{y' \notin \bar{Y}} p(\boldsymbol{x}, y') =& \frac{1}{\binom{k-1}{j}} \sum_{\bar{Y} \in \bar{\mathcal{Y}}_j^y} \left(1 - \sum_{y' \in \bar{Y}} p(\boldsymbol{x}, y')\right) \\ =& \frac{|\bar{\mathcal{Y}}_j^y|}{\binom{k-1}{j}} - \frac{1}{\binom{k-1}{j}} \sum_{\bar{Y} \in \bar{\mathcal{Y}}_j^y} \sum_{y' \in \bar{Y}} p(\boldsymbol{x}, y') \\ =& \frac{\binom{k-1}{j-1}}{\binom{k-1}{j}} - \frac{1}{\binom{k-1}{j}} \sum_{y'} \sum_{\bar{Y}' \in \{\bar{Y}' \in \bar{\mathcal{Y}}_j^y | y' \in \bar{Y}'\}} p(\boldsymbol{x}, y') \qquad \left(\because |\bar{\mathcal{Y}}_j^y| = \tbinom{k-1}{j-1}\right) \\ =& \frac{j}{k-j} - \frac{1}{\binom{k-1}{j}} \left\{ \binom{k-1}{j-1} p(\boldsymbol{x}, y) + \binom{k-2}{j-2} \sum_{y' \neq y} p(\boldsymbol{x}, y') \right\} \\ =& \frac{j}{k-j} - \frac{1}{\binom{k-1}{j}} \left\{ \binom{k-1}{j-1} p(\boldsymbol{x}, y) + \binom{k-2}{j-2} \big(1 - p(\boldsymbol{x}, y)\big) \right\} \\ =& \frac{j}{k-j} - \frac{1}{\binom{k-1}{j}} \left\{ \binom{k-2}{j-2} + \binom{k-2}{j-1} p(\boldsymbol{x}, y) \right\} \qquad \left(\because \tbinom{k-2}{j-1} = \tbinom{k-1}{j-1} - \tbinom{k-2}{j-2}\right) \\ =& \frac{j}{k-j} - \frac{j(j-1)}{(k-j)(k-1)} - \frac{j}{k-1} p(\boldsymbol{x}, y) \\ =& \frac{j}{k-1} - \frac{j}{k-1} p(\boldsymbol{x}, y). \end{aligned} \tag{14}$$

Combing Eq. (13) and Eq. (14), we obtain

$$p(\boldsymbol{x}, y \mid s=j) = p(\boldsymbol{x}, y) = 1 - \frac{k-1}{j} \sum_{\bar{Y} \in \bar{\mathcal{Y}}_j^y} \bar{p}(\boldsymbol{x}, \bar{Y} \mid s=j). \tag{15}$$

In the end, by taking into account the variable s, we have

$$\begin{aligned} p(\boldsymbol{x}, y) &= \textstyle\sum_{j=1}^{k-1} p(s=j) p(\boldsymbol{x}, y \mid s=j) \\ &= \textstyle\sum_{j=1}^{k-1} p(s=j)\Big(1 - \frac{k-1}{j} \sum_{\bar{Y} \in \bar{\mathcal{Y}}_j^y} \bar{p}(\boldsymbol{x}, \bar{Y} \mid s=j)\Big) \\ &= 1 - \textstyle\sum_{j=1}^{k-1} \Big(\frac{k-1}{j} \sum_{\bar{Y} \in \bar{\mathcal{Y}}_j^y} \bar{p}(\boldsymbol{x}, \bar{Y}, s=j)\Big), \end{aligned}$$

which concludes the proof of Lemma 2.

B.2. Proof of Theorem 3

It is intuitive to obtain

$$R(f) = \mathbb{E}_{p(\boldsymbol{x},y)}\big[\mathcal{L}\big(f(\boldsymbol{x}), y\big)\big] = \textstyle\sum_{j=1}^{k-1} p(s=j) \mathbb{E}_{p(\boldsymbol{x},y|s=j)}\big[\mathcal{L}\big(f(\boldsymbol{x}), y\big)\big].$$

Then, we express the right hand side for each $j \in \{1, \ldots, k-1\}$ as

$$\begin{aligned} \mathbb{E}_{p(\boldsymbol{x},y|s=j)}\big[\mathcal{L}\big(f(\boldsymbol{x}), y\big)\big] &= \mathbb{E}_{p(\boldsymbol{x}|s=j)} \mathbb{E}_{p(y|\boldsymbol{x},s=j)}\big[\mathcal{L}\big(f(\boldsymbol{x}), y\big)\big] \\ &= \mathbb{E}_{p(\boldsymbol{x}|s=j)}\Big[\textstyle\sum_{y=1}^{k} p(y|\boldsymbol{x}, s=j) \mathcal{L}\big(f(\boldsymbol{x}), y\big)\Big] \\ &= \mathbb{E}_{p(\boldsymbol{x}|s=j)}\Big[\textstyle\sum_{y=1}^{k} \Big(1 - \frac{k-1}{j} \sum_{\bar{Y} \in \bar{\mathcal{Y}}_j^y} \bar{p}(\bar{Y}|\boldsymbol{x}, s=j)\Big) \mathcal{L}\big(f(\boldsymbol{x}), y\big)\Big] \qquad (\because \text{Eq. (15)}) \\ &= \mathbb{E}_{p(\boldsymbol{x}|s=j)}\Big[\textstyle\sum_{y=1}^{k} \mathcal{L}\big(f(\boldsymbol{x}), y\big) - \frac{k-1}{j} \sum_{y=1}^{k} \sum_{\bar{Y} \in \bar{\mathcal{Y}}_j^y} \bar{p}(\bar{Y}|\boldsymbol{x}, s=j) \mathcal{L}\big(f(\boldsymbol{x}), y\big)\Big] \\ &= \mathbb{E}_{p(\boldsymbol{x}|s=j)}\Big[\textstyle\sum_{y=1}^{k} \mathcal{L}\big(f(\boldsymbol{x}), y\big) - \frac{k-1}{j} \sum_{\bar{Y} \in \bar{\mathcal{Y}}_j} \sum_{y' \in \bar{Y}} \bar{p}(\bar{Y}|\boldsymbol{x}, s=j) \mathcal{L}\big(f(\boldsymbol{x}), y\big)\Big] \\ &= \mathbb{E}_{p(\boldsymbol{x}|s=j)}\Big[\textstyle\sum_{y=1}^{k} \mathcal{L}\big(f(\boldsymbol{x}), y\big) - \frac{k-1}{j} \sum_{\bar{Y} \in \bar{\mathcal{Y}}_j} \bar{p}(\bar{Y}|\boldsymbol{x}, s=j)\Big(\sum_{y' \in \bar{Y}} \mathcal{L}\big(f(\boldsymbol{x}), y'\big)\Big)\Big] \\ &= \mathbb{E}_{p(\boldsymbol{x}|s=j)} \mathbb{E}_{\bar{p}(\bar{Y}|\boldsymbol{x},s=j)}\Big[\textstyle\sum_{y=1}^{k} \mathcal{L}\big(f(\boldsymbol{x}), y\big) - \frac{k-1}{j} \sum_{y' \in \bar{Y}} \mathcal{L}\big(f(\boldsymbol{x}), y'\big)\Big] \\ &= \mathbb{E}_{\bar{p}(\boldsymbol{x},\bar{Y}|s=j)}\Big[\textstyle\sum_{y \notin \bar{Y}} \mathcal{L}\big(f(\boldsymbol{x}), y'\big) - \frac{k-1-j}{j} \sum_{y' \in \bar{Y}} \mathcal{L}\big(f(\boldsymbol{x}), y'\big)\Big] \\ &= \mathbb{E}_{\bar{p}(\boldsymbol{x},\bar{Y}|s=j)}[\bar{\mathcal{L}}_j\big(f(\boldsymbol{x}), \bar{Y}\big)] \\ &= \bar{R}_j(f). \end{aligned}$$

In this way, we can obtain $R(f) = \sum_{j=1}^{k-1} p(s=j) \bar{R}_j(f)$, which concludes the proof of Theorem 3. □

C. Proof of Theorem 4

Recall that the expected risk and empirical risk are represented as

$$\begin{aligned} R(f) &= \textstyle\sum_{j=1}^{k-1} p(s=j) \bar{R}_j(f) = \sum_{j=1}^{k-1} p(s=j) \mathbb{E}_{p(\boldsymbol{x},\bar{Y}|s=j)}[\bar{\mathcal{L}}_j\big(f(\boldsymbol{x}), \bar{Y}\big)], \\ \widehat{R}(f) &= \textstyle\sum_{j=1}^{k-1} \frac{p(s=j)}{n_j} \sum_{i=1}^{n_j} \bar{\mathcal{L}}_j\big(f(\boldsymbol{x}_i), \bar{Y}_i\big). \end{aligned}$$

Here, with a slight abuse of notation, we simply write $\bar{R}_j(f)$ as $R_j(f)$, and define $\widehat{R}_j(f) = 1/n_j \sum_{i=1}^{n_j} \bar{\mathcal{L}}_j\big(f(\boldsymbol{x}_i), \bar{Y}_i\big)$. Thus we have $R(f) = \sum_{j=1}^{k-1} p(s=j) R_j(f)$ and $\widehat{R}(f) = \sum_{j=1}^{k-1} p(s=j) \widehat{R}_j(f)$. Since $f^\star = \arg\min_{f \in \mathcal{F}} R(f)$ and $\widehat{f} = \arg\min_{f \in \mathcal{F}} \widehat{R}(f)$, we can obtain the following lemma.

Lemma 3. *The following inequality holds:*

$$R(\widehat{f}) - R(f^\star) \leqslant 2\sum_{j=1}^{k-1} p(s=j) \sup_{f\in\mathcal{F}} \left|\widehat{R}_j(f) - R_j(f)\right|.$$

Proof. It would be intuitive to obtain

$$\begin{aligned}
R(\widehat{f}) - R(f^\star) =& R(\widehat{f}) - \widehat{R}(\widehat{f}) + \widehat{R}(\widehat{f}) - R(f^\star) \\
\leqslant& R(\widehat{f}) - \widehat{R}(\widehat{f}) + R(\widehat{f}) - R(f^\star) \\
\leqslant& 2\sup_{f\in\mathcal{F}} \left|\widehat{R}(f) - R(f)\right| \\
=& 2\sup_{f\in\mathcal{F}} \left|\sum_{j=1}^{k-1} p(s=j)\widehat{R}_j(f) - \sum_{j=1}^{k-1} p(s=j)R_j(f)\right| \\
\leqslant& 2\sum_{j=1}^{k-1} p(s=j) \sup_{f\in\mathcal{F}} \left|\widehat{R}_j(f) - R_j(f)\right|,
\end{aligned}$$

which concludes the proof of Lemma 3. □

In this way, we will bound $\sup_{f\in\mathcal{F}} \left|\widehat{R}_j(f) - R_j(f)\right|$ for $j = \{1, \ldots, k-1\}$. Before that, we define a function space as

$$\mathcal{H}_j := \{(\boldsymbol{x}, \bar{Y}) \in \mathcal{X} \times \bar{\mathcal{Y}}_j \mapsto \bar{\mathcal{L}}_j\big(f(\boldsymbol{x}), \bar{Y}\big) \mid f \in \mathcal{F}\},$$

where

$$\bar{\mathcal{L}}_j\big(f(\boldsymbol{x}), \bar{Y}\big) := \sum\nolimits_{y\notin\bar{Y}} \mathcal{L}\big(f(\boldsymbol{x}), y\big) - \frac{k-1-j}{j} \sum\nolimits_{y'\in\bar{Y}} \mathcal{L}\big(f(\boldsymbol{x}), y'\big).$$

Besides, we introduce the definition of *Rademacher complexity* (Bartlett & Mendelson, 2002).

Definition 1 (Rademacher complexity (Bartlett & Mendelson, 2002))**.** *Let $Z_1, \ldots, Z_n$ be n i.i.d. random variables drawn from a probability distribution $\mathcal{D}$, $\mathcal{H} = \{h : \mathcal{Z} \to \mathbb{R}\}$ be a class of measurable functions. Then the expected Rademacher complexity of $\mathcal{H}$ is defined as*

$$\mathfrak{R}_n(\mathcal{H}) = \mathbb{E}_{Z_1,\ldots,Z_n\sim\mathcal{D}}\mathbb{E}_{\boldsymbol{\sigma}}\left[\sup_{h\in\mathcal{H}} \frac{1}{n}\sum\nolimits_{i=1}^{n} \sigma_i h(Z_i)\right],$$

where $\boldsymbol{\sigma} = (\sigma_1, \ldots, \sigma_n)$ are Rademacher variables taking the value from $\{-1, +1\}$ with even probabilities.

Then, we have the following lemma.

Lemma 4. *Let $C_{\mathcal{L}} = \sup_{\boldsymbol{x}\in\mathcal{X}, f\in\mathcal{F}, y\in\mathcal{Y}} \mathcal{L}\big(f(\boldsymbol{x}), y\big)$. Then, for all $j = \{1, \ldots, k-1\}$, for any $\delta > 0$, with probability at least $1-\delta$,*

$$\sup_{f\in\mathcal{F}} \left|\widehat{R}_j(f) - R_j(f)\right| \leqslant 2\bar{\mathfrak{R}}_{n_j}(\mathcal{H}_j) + (2k-2j-1)C_{\mathcal{L}}\sqrt{\frac{\log\frac{2}{\delta}}{2n_j}}, \tag{16}$$

where

$$\bar{\mathfrak{R}}_{n_j}(\mathcal{H}_j) = \mathbb{E}_{(\boldsymbol{x}_i,\bar{Y}_i)\sim\bar{p}(\boldsymbol{x},\bar{Y}|s=j)}\mathbb{E}_{\boldsymbol{\sigma}}\left[\sup_{h\in\mathcal{H}_j} \frac{1}{n_j}\sum\nolimits_{i=1}^{n_j} \sigma_i h(\boldsymbol{x}_i, \bar{Y}_i)\right]. \tag{17}$$

Proof. To prove this lemma, we first show that the single direction $\sup_{f\in\mathcal{F}}\big(\widehat{R}_j(f) - R_j(f)\big)$ is bounded with probability at least $1 - \frac{\delta}{2}$, and the other direction can be similarly proved. By the definition of $\bar{\mathcal{L}}_j$, we can easily know the possible maximum of $\bar{\mathcal{L}}_j$ is $(k-j)C_{\mathcal{L}}$, and the possible minimum is $-(k-1-j)C_{\mathcal{L}}$. Suppose an example $(\boldsymbol{x}_i, \bar{Y}_i)$ is replaced by

another arbitrary example $(\boldsymbol{x}_i', \bar{Y}_i')$, then the change of $\sup_{f\in\mathcal{F}}\big(\widehat{R}_j(f) - R_j(f)\big)$ is no greater than $((2k-2j-1)C_{\mathcal{L}})/n_j$. Then, by applying *McDiarmid's inequality* (McDiarmid, 1989), for any $\delta > 0$, with probability at least $1-\frac{\delta}{2}$,

$$\sup_{f\in\mathcal{F}}\big(\widehat{R}_j(f) - R_j(f)\big) \leqslant \mathbb{E}\Big[\sup_{f\in\mathcal{F}}\big(\widehat{R}_j(f) - R_j(f)\big)\Big] + (2k-2j-1)C_{\mathcal{L}}\sqrt{\frac{\log\frac{2}{\delta}}{2n_j}}. \tag{18}$$

In addition, it is routine (Mohri et al., 2012) to show

$$\mathbb{E}\Big[\sup_{f\in\mathcal{F}}\big(\widehat{R}_j(f) - R_j(f)\big)\Big] \leqslant 2\bar{\mathfrak{R}}_{n_j}(\mathcal{H}_j), \tag{19}$$

Combing Eq. (18) and Eq. (19), we have for any $\delta > 0$, with probability at least $1-\frac{\delta}{2}$,

$$\sup_{f\in\mathcal{F}}\big(\widehat{R}_j(f) - R_j(f)\big) \leqslant 2\bar{\mathfrak{R}}_{n_j}(\mathcal{H}_j) + (2k-2j-1)C_{\mathcal{L}}\sqrt{\frac{\log\frac{2}{\delta}}{2n_j}}. \tag{20}$$

By further taking into account the other side $\sup_{f\in\mathcal{F}}\big(R_j(f) - \widehat{R}_j(f)\big)$, we have for any $\delta > 0$, with probability at least $1-\delta$,

$$\sup_{f\in\mathcal{F}}\Big|\widehat{R}_j(f) - R_j(f)\Big| \leqslant 2\bar{\mathfrak{R}}_{n_j}(\mathcal{H}_j) + (2k-2j-1)C_{\mathcal{L}}\sqrt{\frac{\log\frac{2}{\delta}}{2n_j}}.$$

which concludes the proof of Lemma 4. □

Next, we will bound the expected Rademacher complexity of the function space $\mathcal{H}_j$, i.e., $\bar{\mathfrak{R}}_{n_j}(\mathcal{H}_j)$.

Lemma 5. *Assume the loss function $\mathcal{L}\big(f(\boldsymbol{x}), y\big)$ is ρ-Lipschitz with respect to $f(\boldsymbol{x})$ $(0 < \rho < \infty)$ for all $y \in \mathcal{Y}$. Then, for all $j = \{1, \ldots, k-1\}$, the following inequality holds:*

$$\bar{\mathfrak{R}}_{n_j}(\mathcal{H}_j) \leqslant \frac{\rho(k-1)}{j}\sum\nolimits_{y=1}^{k}\mathfrak{R}_{n_j}(\mathcal{G}_y),$$

where

$$\begin{aligned}\mathcal{G}_y =& \{g : \boldsymbol{x} \mapsto f_y(\boldsymbol{x}) \mid f \in \mathcal{F}\},\\ \mathfrak{R}_{n_j}(\mathcal{G}_y) =& \mathbb{E}_{\boldsymbol{x}_i\sim p(\boldsymbol{x})}\mathbb{E}_{\boldsymbol{\sigma}}\Big[\sup_{g\in\mathcal{G}_y}\frac{1}{n_j}\sum\nolimits_{i=1}^{n_j}g(\boldsymbol{x}_i)\Big].\end{aligned}$$

Proof. The expected Rademacher complexity of $\mathcal{H}_j$ can be expressed as

$$\begin{aligned}\bar{\mathfrak{R}}_{n_j}(\mathcal{H}_j) =& \mathbb{E}_{(\boldsymbol{x}_i,\bar{Y}_i)\sim\bar{p}(\boldsymbol{x},\bar{Y}|s=j)}\mathbb{E}_{\boldsymbol{\sigma}}\Big[\sup_{h\in\mathcal{H}_j}\frac{1}{n_j}\sum\nolimits_{i=1}^{n_j}\sigma_i h(\boldsymbol{x}_i,\bar{Y}_i)\Big]\\ =& \mathbb{E}_{(\boldsymbol{x}_i,\bar{Y}_i)\sim\bar{p}(\boldsymbol{x},\bar{Y}|s=j)}\mathbb{E}_{\boldsymbol{\sigma}}\Bigg[\sup_{f\in\mathcal{F}}\frac{1}{n_j}\sum\nolimits_{i=1}^{n_j}\sigma_i\Bigg(\sum\nolimits_{y\notin\bar{Y}_i}\mathcal{L}\big(f(\boldsymbol{x}),y\big) - \frac{k-j-1}{j}\sum\nolimits_{y'\in\bar{Y}_i}\mathcal{L}\big(f(\boldsymbol{x}),y'\big)\Bigg)\Bigg]\\ \leqslant& \mathbb{E}_{(\boldsymbol{x}_i,\bar{Y}_i)\sim\bar{p}(\boldsymbol{x},\bar{Y}|s=j)}\mathbb{E}_{\boldsymbol{\sigma}}\Bigg[\sup_{f\in\mathcal{F}}\frac{1}{n_j}\sum\nolimits_{i=1}^{n_j}\sigma_i\Bigg(\sum\nolimits_{y\notin\bar{Y}_i}\mathcal{L}\big(f(\boldsymbol{x}),y\big)\Bigg)\Bigg]\\ &+ \mathbb{E}_{(\boldsymbol{x}_i,\bar{Y}_i)\sim\bar{p}(\boldsymbol{x},\bar{Y}|s=j)}\mathbb{E}_{\boldsymbol{\sigma}}\Bigg[\sup_{f\in\mathcal{F}}\frac{1}{n_j}\sum\nolimits_{i=1}^{n_j}\sigma_i\Bigg(\frac{k-j-1}{j}\sum\nolimits_{y'\in\bar{Y}_i}\mathcal{L}\big(f(\boldsymbol{x}),y'\big)\Bigg)\Bigg].\end{aligned}$$

Here, we introduce random variables $\alpha_{i,y} = \mathbb{I}[y \in \bar{Y}_i],\ \forall i \in \{1, \cdots, n\}, y \in \mathcal{Y}$, where $\mathbb{I}[\cdot]$ denotes the indicator function. In other words, given a complementary label set $\bar{Y}_i$, if a specific label y satisfies the condition $y \in \bar{Y}_i$, then $\mathbb{I}[y \in \bar{Y}_i] = 1$,

otherwise $\mathbb{I}[y \in \bar{Y}_i] = 0$. Then, we can obtain

$$\begin{aligned}
\bar{\mathfrak{R}}_{n_j}(\mathcal{H}_j) \leqslant & \mathbb{E}_{(\boldsymbol{x}_i,\bar{Y}_i)\sim\bar{p}(\boldsymbol{x},\bar{Y}|s=j)}\mathbb{E}_{\boldsymbol{\sigma}}\left[\sup_{f\in\mathcal{F}}\frac{1}{n_j}\sum_{i=1}^{n_j}\sigma_i\left(\sum\nolimits_{y\notin\bar{Y}_i}\mathcal{L}\big(f(\boldsymbol{x}),y\big)\right)\right] \\
& + \mathbb{E}_{(\boldsymbol{x}_i,\bar{Y}_i)\sim\bar{p}(\boldsymbol{x},\bar{Y}|s=j)}\mathbb{E}_{\boldsymbol{\sigma}}\left[\sup_{f\in\mathcal{F}}\frac{1}{n_j}\sum_{i=1}^{n_j}\sigma_i\left(\frac{k-j-1}{j}\sum\nolimits_{y'\in\bar{Y}_i}\mathcal{L}\big(f(\boldsymbol{x}),y'\big)\right)\right] \\
= & \mathbb{E}_{(\boldsymbol{x}_i,\bar{Y}_i)\sim\bar{p}(\boldsymbol{x},\bar{Y}|s=j)}\mathbb{E}_{\boldsymbol{\sigma}}\left[\sup_{f\in\mathcal{F}}\frac{1}{n_j}\sum_{i=1}^{n_j}\sigma_i\left(\sum_{y=1}^{k}(1-\alpha_{i,y})\mathcal{L}\big(f(\boldsymbol{x}),y\big)\right)\right] \\
& + \mathbb{E}_{(\boldsymbol{x}_i,\bar{Y}_i)\sim\bar{p}(\boldsymbol{x},\bar{Y}|s=j)}\mathbb{E}_{\boldsymbol{\sigma}}\left[\sup_{f\in\mathcal{F}}\frac{1}{n_j}\sum_{i=1}^{n_j}\sigma_i\left(\frac{k-j-1}{j}\sum_{y=1}^{k}\alpha_{i,y}\mathcal{L}\big(f(\boldsymbol{x}),y\big)\right)\right] \\
= & \mathbb{E}_{(\boldsymbol{x}_i,\bar{Y}_i)\sim\bar{p}(\boldsymbol{x},\bar{Y}|s=j)}\mathbb{E}_{\boldsymbol{\sigma}}\left[\sup_{f\in\mathcal{F}}\frac{1}{n_j}\sum_{i=1}^{n_j}\sigma_i\left(\sum_{y=1}^{k}\frac{1}{2}(1-2\alpha_{i,y}+1)\mathcal{L}\big(f(\boldsymbol{x}),y\big)\right)\right] \\
& + \mathbb{E}_{(\boldsymbol{x}_i,\bar{Y}_i)\sim\bar{p}(\boldsymbol{x},\bar{Y}|s=j)}\mathbb{E}_{\boldsymbol{\sigma}}\left[\sup_{f\in\mathcal{F}}\frac{1}{n_j}\sum_{i=1}^{n_j}\sigma_i\left(\frac{k-j-1}{j}\sum_{y=1}^{k}\frac{1}{2}(2\alpha_{i,y}-1+1)\mathcal{L}\big(f(\boldsymbol{x}),y\big)\right)\right] \\
= & \mathbb{E}_{(\boldsymbol{x}_i,\bar{Y}_i)\sim\bar{p}(\boldsymbol{x},\bar{Y}|s=j)}\mathbb{E}_{\boldsymbol{\sigma}}\left[\sup_{f\in\mathcal{F}}\frac{1}{2n_j}\sum_{i=1}^{n_j}\left(\sum_{y=1}^{k}(1-2\alpha_{i,y})\sigma_i\mathcal{L}\big(f(\boldsymbol{x}),y\big)+\sigma_i\mathcal{L}\big(f(\boldsymbol{x}),y\big)\right)\right] \\
& + \mathbb{E}_{(\boldsymbol{x}_i,\bar{Y}_i)\sim\bar{p}(\boldsymbol{x},\bar{Y}|s=j)}\mathbb{E}_{\boldsymbol{\sigma}}\left[\sup_{f\in\mathcal{F}}\frac{1}{2n_j}\sum_{i=1}^{n_j}\left(\frac{k-j-1}{j}\sum_{y=1}^{k}(2\alpha_{i,y}-1)\sigma_i\mathcal{L}\big(f(\boldsymbol{x}),y'\big)+\sigma_i\mathcal{L}\big(f(\boldsymbol{x}),y'\big)\right)\right].
\end{aligned}$$

Here, because $(1-2\alpha_{i,y})\sigma_i$ and $(2\alpha_{i,y}-1)\sigma_i$, and σ_i follow the same distribution, we have

$$\begin{aligned}
\bar{\mathfrak{R}}_{n_j}(\mathcal{H}_j) \leqslant & \mathbb{E}_{(\boldsymbol{x}_i,\bar{Y}_i)\sim\bar{p}(\boldsymbol{x},\bar{Y}|s=j)}\mathbb{E}_{\boldsymbol{\sigma}}\left[\sup_{f\in\mathcal{F}}\frac{1}{2n_j}\sum_{i=1}^{n_j}\left(\sum_{y=1}^{k}(1-2\alpha_{i,y})\sigma_i\mathcal{L}\big(f(\boldsymbol{x}),y\big)+\sigma_i\mathcal{L}\big(f(\boldsymbol{x}),y\big)\right)\right] \\
& + \mathbb{E}_{(\boldsymbol{x}_i,\bar{Y}_i)\sim\bar{p}(\boldsymbol{x},\bar{Y}|s=j)}\mathbb{E}_{\boldsymbol{\sigma}}\left[\sup_{f\in\mathcal{F}}\frac{1}{2n_j}\sum_{i=1}^{n_j}\left(\frac{k-j-1}{j}\sum_{y=1}^{k}(2\alpha_{i,y}-1)\sigma_i\mathcal{L}\big(f(\boldsymbol{x}),y'\big)+\sigma_i\mathcal{L}\big(f(\boldsymbol{x}),y'\big)\right)\right] \\
\leqslant & \mathbb{E}_{(\boldsymbol{x}_i,\bar{Y}_i)\sim\bar{p}(\boldsymbol{x},\bar{Y}|s=j)}\mathbb{E}_{\boldsymbol{\sigma}}\left[\sup_{f\in\mathcal{F}}\frac{1}{n_j}\sum_{i=1}^{n_j}\sum_{y=1}^{k}\sigma_i\mathcal{L}\big(f(\boldsymbol{x}_i),y\big)\right] \\
& + \mathbb{E}_{(\boldsymbol{x}_i,\bar{Y}_i)\sim\bar{p}(\boldsymbol{x},\bar{Y}|s=j)}\mathbb{E}_{\boldsymbol{\sigma}}\left[\sup_{f\in\mathcal{F}}\frac{1}{n_j}\sum_{i=1}^{n_j}\frac{k-j-1}{j}\sum_{y=1}^{k}\sigma_i\mathcal{L}\big(f(\boldsymbol{x}_i),y\big)\right] \\
= & \frac{k-1}{j}\mathbb{E}_{(\boldsymbol{x}_i,\bar{Y}_i)\sim\bar{p}(\boldsymbol{x},\bar{Y}|s=j)}\mathbb{E}_{\boldsymbol{\sigma}}\left[\sup_{f\in\mathcal{F}}\frac{1}{n_j}\sum_{i=1}^{n_j}\sum_{y=1}^{k}\sigma_i\mathcal{L}\big(f(\boldsymbol{x}_i),y\big)\right] \\
\leqslant & \frac{k-1}{j}\sum_{y=1}^{k}\mathbb{E}_{\boldsymbol{x}_i\sim p(\boldsymbol{x})}\mathbb{E}_{\boldsymbol{\sigma}}\left[\sup_{f\in\mathcal{F}}\frac{1}{n_j}\sum_{i=1}^{n_j}\sigma_i\mathcal{L}\big(f(\boldsymbol{x}_i),y\big)\right] \qquad (\because p(\boldsymbol{x})=\bar{p}(\boldsymbol{x}\mid s=j)),
\end{aligned}$$

Then, we have

$$\begin{aligned}
\bar{\mathfrak{R}}_{n_j}(\mathcal{H}_j) \leqslant & \frac{k-1}{j}\sum_{y=1}^{k}\mathbb{E}_{\boldsymbol{x}_i\sim p(\boldsymbol{x})}\mathbb{E}_{\boldsymbol{\sigma}}\left[\sup_{f\in\mathcal{F}}\frac{1}{n_j}\sum_{i=1}^{n_j}\sigma_i\mathcal{L}\big(f(\boldsymbol{x}_i),y\big)\right] \\
\leqslant & \frac{k-1}{j}\sum_{y=1}^{k}\mathfrak{R}_{n_j}(\mathcal{L}\circ\mathcal{F}) \\
\leqslant & \frac{\sqrt{2}\rho k(k-1)}{j}\sum_{y=1}^{k}\mathfrak{R}_{n_j}(\mathcal{G}_y),
\end{aligned}$$

where we applied the Rademacher vector contraction inequality (Maurer, 2016) in the last inequality. □

Table 5. Statistics of the used benchmark datasets.

Dataset	#Train	#Test	#Features	#Classes	Model
MNIST	60,000	10,000	784	10	Linear Model, MLP (d-500-10)
Fashion-MNIST	60,000	10,000	784	10	Linear Model, MLP (d-500-10)
Kuzushiji-MNIST	60,000	10,000	784	10	Linear Model, MLP (d-500-10)
20Newsgroups	16,961	1,885	1,000	20	Linear Model, MLP (d-500-20)
CIFAR-10	50,000	10,000	3,072	10	ResNet, DenseNet
Yeast	1,335	149	8	10	Linear Model
Texture	4,950	550	40	11	Linear Model
Dermatology	329	37	34	6	Linear Model
Synthetic Control	540	60	60	6	Linear Model

Under the assumptions described in the above three lemmas (Lemma 3, Lemma 4, and Lemma 5), for any $\delta > 0$, with probability at least $1 - \delta$,

$$R(\widehat{f}) - R(f^\star) \leqslant \sum_{j=1}^{k-1} p(s=j)\left(\frac{4\sqrt{2}\rho k(k-1)}{j}\sum_{y=1}^{k}\mathfrak{R}_{n_j}(\mathcal{G}_y) + (4k-4j-2)C_{\mathcal{L}}\sqrt{\frac{\log\frac{2(k-1)}{\delta}}{2n_j}}\right).$$

It is clear that by combining the above three lemmas, Theorem 4 is proved. □

D. Derivations and Boundness of the Used Loss Functions

D.1. Derivations of the Used Loss Functions

Conventionally, the label for each instance $\boldsymbol{x}$ is in one-hot encoding. Concretely, if the label of $\boldsymbol{x}$ is y, then we represent the label vector as $\boldsymbol{e}_y$ where $e_{yj} = 1$ if $j = y$, otherwise 0. In this way, we provide the detailed derivations of CCE, MAE, and MSE as follows.

- Categorical Cross Entropy (CCE):

$$\mathcal{L}_{\text{CCE}}(f(\boldsymbol{x}), y) = -\sum_{j=1}^{k} e_{yj}\log p_{\boldsymbol{\theta}}(j|\boldsymbol{x}) = -\log p_{\boldsymbol{\theta}}(y|\boldsymbol{x}).$$

- Mean Absolute Error (MAE):

$$\mathcal{L}_{\text{MAE}}(f(\boldsymbol{x}), y) = \sum_{j=1}^{k} |p_{\boldsymbol{\theta}}(j|\boldsymbol{x}) - e_{yj}| = 2 - 2p_{\boldsymbol{\theta}}(y|\boldsymbol{x}).$$

- Mean Square Error (MSE):

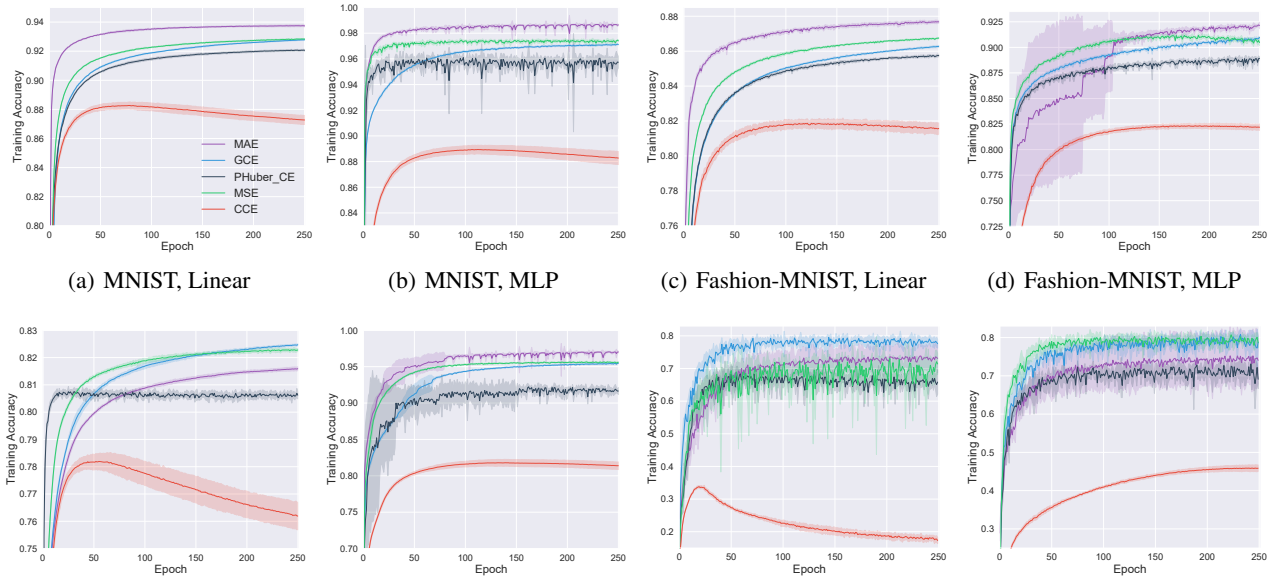
$$\mathcal{L}_{\text{MSE}}(f(\boldsymbol{x}), y) = \sum_{j=1}^{k}\left(p_{\boldsymbol{\theta}}(j|\boldsymbol{x}) - e_{yj}\right)^2 = 1 - 2p_{\boldsymbol{\theta}}(y|\boldsymbol{x}) + \sum_{j=1}^{k} p_{\boldsymbol{\theta}}(j|\boldsymbol{x})^2.$$

D.2. Boundness of the Used Loss Functions

Firstly, it is clear that each loss function is non-negative. Besides, for each loss function, the loss becomes larger if $p_{\boldsymbol{\theta}}(y|\boldsymbol{x})$ gets smaller given the correct label y. Note that $0 < p_{\boldsymbol{\theta}}(y|\boldsymbol{x}) < 1$, hence the upper bound of each loss function is stated as follows.

- MAE: $\mathcal{L}_{\text{MAE}}(f(\boldsymbol{x}), y) < 2$.
- MSE: $\mathcal{L}_{\text{MSE}}(f(\boldsymbol{x}), y) < 1 - 0 + \sum_{j=1}^{k} p_{\boldsymbol{\theta}}(j|\boldsymbol{x})^2 < 2$.
- GCE: $\mathcal{L}_{\text{GCE}}(f(\boldsymbol{x}), y) < 1/q$ where $q = 0.7$.
- PHuber-CE: $\mathcal{L}_{\text{PHuber-CE}}(f(\boldsymbol{x}), y) < \log\tau + 1$ where $\tau = 10$.

Note that for CCE, $\mathcal{L}_{\text{CCE}}(f(\boldsymbol{x}), y) < -\log 0 = \infty$. Therefore, we can know that MAE, MSE, GCE, and PHuber-CE are upper-bounded, while CCE is not upper-bounded.



(a) MNIST, Linear (b) MNIST, MLP (c) Fashion-MNIST, Linear (d) Fashion-MNIST, MLP

(e) Kuzushiji-MNIST, Linear (f) Kuzushiji-MNIST, MLP (g) CIFAR-10, Resnet (h) CIFAR-10, Densenet

Figure 2. Experimental results of different loss functions for different datasets and models. Dark colors show the mean accuracy of 5 trials and light colors show the standard deviation.

E. Additional Information of Experiments

E.1. Datasets and Models

In the experiments of Section 5, we use 5 widely-used large-scale benchmark datasets and 4 regular-scale datasets from the UCI Machine Learning Repository. The statistics of these datasets with the corresponding base models are reported in Table 5. Hyper-parameters for all the approaches are selected so as to maximize the accuracy on a validation set, which is constructed by randomly sampling 10% of the training set. We report the characteristics, the parameter settings (to reproduce the experimental results), and the sources of these datasets as follows.

- MNIST (LeCun et al., 1998): It is a 10-class dataset of handwritten digits (0 to 9). Each instance is a 28×28 grayscale image. Source: `http://yann.lecun.com/exdb/mnist/`
- Kuzushiji-MNIST (Clanuwat et al., 2018): It is a 10-class dataset of cursive Japanese ("Kuzushiji") characters. Each instance is a 28×28 grayscale image.
- Fashion-MNIST (Xiao et al., 2017): It is a 10-class dataset of fashion items (T-shirt/top, trouser, pullover, dress, sandal, coat, shirt, sneaker, bag, and ankle boot). Each instance is a 28×28 grayscale image. Source: `https://github.com/rois-codh/kmnist`
- CIFAR-10 (Krizhevsky et al., 2009): It is a 10-class dataset of 10 different objects (airplane, bird, automobile, cat, deer, dog, frog, horse, ship, and truck). Each instance is a $32 \times 32 \times 3$ colored image in RGB format. This dataset is normalized with mean $(0.4914, 0.4822, 0.4465)$ and standard deviation $(0.247, 0.243, 0.261)$. Source: `https://www.cs.toronto.edu/~kriz/cifar.html`
- 20Newsgroups: It is a 20-class dataset of 20 different newsgroups (comp.graphics, comp.os.ms-windows.misc, comp.sys.ibm.pc.hardware, comp.sys.mac.hardware, comp.windows.x, rec.autos, rec.motorcycles, rec.sport.baseball, rec.sport.hockey, sci.crypt, sci.electronics, sci.med, sci.space, misc.forsale, talk.politics.misc, talk.politics.guns, talk.politics.mideast, talk.religion.misc, alt.atheism, soc.religion.christian). We obtained the tf-idf features, and applied TruncatedSVD (Halko et al., 2011) to reduce the dimension to 1000. We randomly sample 90% of the examples from the whole dataset to construct the training set, and the rest 10% forms the test set. Source: `http://qwone.com/~jason/20Newsgroups/`

Table 6. Classification accuracy (%) of each approach on Kuzushiji-MNIST using linear model. The best performance is highlighted in boldface.

Approach		$s=1$	$s=2$	$s=3$	$s=4$	$s=5$	$s=6$	$s=7$	$s=8$
Upper-bound Losses	EXP	**60.87** (±0.38)	**62.73** (±0.58)	**63.53** (±0.30)	**64.03** (±0.38)	**64.55** (±0.41)	65.06 (±0.15)	65.23 (±0.10)	65.65 (±0.08)
	LOG	60.11 (±0.49)	61.57 (±0.15)	62.71 (±0.32)	63.36 (±0.09)	64.01 (±0.13)	65.68 (±0.27)	**69.35** (±0.22)	**70.10** (±0.18)
Bounded Losses	MAE	60.43 (±0.43)	62.71 (±0.45)	63.51 (±0.10)	63.75 (±0.31)	63.94 (±0.38)	64.61 (±0.19)	64.82 (±0.16)	65.10 (±0.16)
	MSE	58.97 (±0.47)	62.07 (±0.54)	63.05 (±0.38)	63.85 (±0.57)	64.47 (±0.43)	64.80 (±0.34)	65.17 (±0.25)	65.43 (±0.10)
	GCE	60.48 (±0.55)	62.71 (±0.65)	63.13 (±0.30)	63.87 (±0.33)	63.91 (±0.30)	64.28 (±0.07)	64.38 (±0.12)	64.33% (±0.06)
	Phuber-CE	52.69 (±4.22)	56.58 (±3.94)	61.10 (±2.58)	62.32 (±1.50)	64.51 (±0.68)	64.93 (±0.52)	65.96 (±0.37)	65.81 (±0.62)
Unbounded Loss	CCE	51.59 (±0.64)	55.98 (±1.26)	59.15 (±1.18)	61.08 (±0.78)	63.19 (±0.54)	65.05 (±0.51)	66.82 (±0.41)	68.23 (±0.21)
Decomposition before Shuffle	GA	51.72 (±1.04)	53.78 (±1.07)	54.58 (±0.87)	54.78 (±0.58)	55.33 (±0.29)	55.67 (±0.31)	55.91 (±0.42)	56.15 (±0.23)
	NN	55.03 (±1.35)	57.68 (±1.29)	58.87 (±1.19)	59.52 (±0.87)	60.41 (±0.59)	60.89 (±0.53)	61.41 (±0.36)	61.62 (±0.09)
	FREE	57.26 (±0.83)	60.69 (±0.96)	62.77 (±0.79)	63.91 (±0.65)	64.54 (±0.55)	**66.21** (±0.56)	67.00 (±0.28)	67.71 (±0.20)
	PC	54.31 (±1.04)	58.11 (±0.87)	60.15 (±0.79)	61.32 (±0.68)	62.56 (±0.59)	63.55 (±0.43)	64.27 (±0.20)	65.16 (±0.18)
	Forward	60.05 (±0.43)	61.53 (±0.31)	62.43 (±0.26)	62.98 (±0.40)	63.48 (±0.34)	63.95 (±0.29)	64.14 (±0.09)	64.27 (±0.16)
Decomposition after Shuffle	GA	51.72 (±1.05)	53.79 (±1.07)	54.59 (±0.85)	54.83 (±0.58)	55.33 (±0.35)	55.67 (±0.31)	55.90 (±0.41)	56.18 (±0.22)
	NN	55.03 (±1.35)	58.58 (±1.11)	60.43 (±1.00)	61.58 (±0.72)	62.99 (±0.49)	64.00 (±0.48)	65.07 (±0.36)	66.08 (±0.10)
	FREE	57.26 (±0.84)	60.32 (±0.94)	62.11 (±0.64)	62.98 (±0.67)	64.30 (±0.47)	65.18 (±0.45)	66.02 (±0.28)	67.02 (±0.18)
	PC	54.31 (±1.04)	57.32 (±0.76)	58.95 (±0.77)	60.17 (±0.83)	61.47 (±0.45)	62.54 (±0.40)	63.53 (±0.22)	64.74 (±0.22)
	Forward	60.02 (±0.44)	61.75 (±0.25)	62.68 (±0.23)	63.19 (±0.28)	63.59 (±0.19)	63.94 (±0.09)	64.18 (±0.14)	64.32 (±0.15)

- Yeast, Texture, Dermatology, Synthetic Control: They are all the datasets from the UCI Machine Learning Repository. Since they are all regular-scale datasets, we only apply linear model on them. For each dataset, we randomly sample 90% of the examples from the whole dataset to construct the training set, and the rest 10% forms the test set. The detailed parameter settings can be found in our provided code package. Source: `https://archive.ics.uci.edu/ml/datasets.php`

For the used models, the detailed information of the used 34-layer ResNet (He et al., 2016) and 22-layer DenseNet (Huang et al., 2017) can be found in the corresponding papers.

E.2. Experimental Results on Training Accuracy

Here, we report the mean and standard deviation of training accuracy (the training set is evaluated with ordinary labels) of 5 trials in Figure 2, to compare the bounded loss functions MAE, MSE, GCE, PHuber-CE, and the unbounded loss function CCE. The training accuracy can reflect the ability of the loss function in identifying the correct label from the non-complementary labels.

From Figure 2, we can find that CCE always achieves the worst performance among all the loss functions, which implies that unbounded loss function is worse than bounded loss function, using our provided empirical risk estimator. This observation clearly supports our conjecture that the negative term in our empirical risk estimator could cause the over-fitting issue. In addition, we can also find that compared with other bounded loss functions, MAE achieves comparable performance in most

Table 7. Classification accuracy (%) of each approach on Kuzushiji-MNIST using MLP. The best performance is highlighted in boldface.

Approach		$s=1$	$s=2$	$s=3$	$s=4$	$s=5$	$s=6$	$s=7$	$s=8$
Upper-bound Losses	EXP	71.66 (±3.48)	**82.51** (±3.08)	84.45 (±0.24)	87.10 (±0.37)	88.35 (±0.18)	**89.61** (±0.33)	90.18 (±0.37)	90.92 (±0.15)
	LOG	**77.07** (±3.00)	82.39 (±0.73)	**85.54** (±0.35)	**87.60** (±0.40)	**88.87** (±0.34)	89.25 (±0.37)	**90.22** (±0.31)	**91.19** (±0.11)
Bounded Losses	MAE	69.87 (±1.04)	73.60 (±5.77)	79.97 (±3.71)	85.34 (±2.78)	86.91 (±3.06)	89.10 (±0.46)	90.32 (±0.31)	91.06 (±0.34)
	MSE	57.56 (±0.92)	71.37 (±0.89)	78.26 (±0.49)	82.97 (±0.41)	85.37 (±0.45)	86.82 (±0.13)	88.03 (±0.11)	88.69 (±0.05)
	GCE	63.85 (±1.27)	74.11 (±2.38)	79.18 (±2.31)	83.65 (±0.15)	85.23 (±0.25)	86.32 (±0.27)	87.12 (±0.20)	87.64 (±0.09)
	Phuber-CE	10.24 (±4.09)	14.76 (±2.11)	26.60 (±1.58)	73.43 (±1.50)	81.41 (±0.58)	83.00 (±0.42)	84.69 (±0.47)	85.59 (±0.52)
Unbounded Loss	CCE	56.17 (±0.64)	60.89 (±0.61)	64.18 (±0.77)	66.57 (±0.41)	69.14 (±0.49)	71.63 (±0.31)	74.55 (±0.31)	78.22 (±0.22)
Decomposition before Shuffle	GA	70.25 (±0.24)	76.50 (±0.47)	79.77 (±0.32)	82.03 (±0.22)	84.05 (±0.64)	85.58 (±0.32)	86.40 (±0.24)	87.49 (±0.15)
	NN	65.33 (±0.51)	71.34 (±0.53)	75.46 (±0.31)	78.67 (±0.58)	81.40 (±0.28)	84.08 (±0.16)	86.56 (±0.39)	88.61 (±0.12)
	FREE	53.90 (±1.05)	60.32 (±1.14)	63.98 (±0.85)	66.79 (±0.64)	69.31 (±0.73)	71.65 (±0.73)	74.43 (±0.28)	76.61 (±0.33)
	PC	56.36 (±0.56)	62.37 (±0.50)	66.09 (±0.44)	69.51 (±0.47)	72.46 (±0.35)	75.18 (±0.33)	78.50 (±0.52)	82.40 (±0.38)
	Forward	75.40 (±2.02)	83.19 (±0.61)	85.18 (±0.48)	86.63 (±0.38)	87.51 (±0.29)	88.29 (±0.29)	88.96 (±0.26)	89.41 (±0.25)
Decomposition after Shuffle	GA	70.25 (±0.24)	75.91 (±1.37)	78.46 (±2.84)	80.60 (±3.35)	82.14 (±4.51)	83.48 (±4.92)	84.01 (±5.35)	84.65 (±6.28)
	NN	63.73 (±0.97)	67.26 (±0.82)	69.46 (±0.74)	71.25 (±0.62)	73.15 (±0.45)	74.82 (±0.35)	77.09 (±0.17)	79.39 (±0.21)
	FREE	55.33 (±0.89)	60.81 (±0.97)	64.65 (±0.89)	67.01 (±0.70)	69.60 (±0.78)	71.63 (±0.46)	74.22 (±0.40)	77.16 (±0.50)
	PC	56.68 (±1.28)	61.07 (±0.99)	63.86 (±0.67)	65.61 (±0.44)	68.03 (±0.64)	69.74 (±0.65)	72.49 (±0.37)	75.17 (±0.46)
	Forward	66.09 (±0.49)	73.20 (±3.05)	75.76 (±2.61)	82.53 (±2.60)	86.27 (±0.65)	88.05 (±0.27)	89.24 (±0.22)	90.22 (±0.20)

cases, while it is sometimes inferior to other bounded losses due to its optimization issue (Zhang & Sabuncu, 2018). All the above observations on the training accuracy (Figure 2) are very similar to those observations on the test accuracy (Figure 1 in our paper).

E.3. Experimental Results on Fixed Complementary Label Set

We also conduct additional experiments to investigate the influence of the variable s on Kuzushiji-MNIST using both linear model and MLP. Specifically, we study the case where the size of each complementary label set s is fixed at j (i.e., $p(s=j)=1$) while increasing j from 1 to $k-2$. The detailed experimental results are shown in Table 6 and Table 7. From the two tables, we can find that the (test) classification accuracy of our approaches increases as j increases. This observation is clearly in accordance with our derived estimation error bound (Theorem 4), as the estimation error would decrease if j increases. In addition, as shown in the two tables, our proposed upper-bound losses outperform other approaches in most cases. This observation also demonstrates the effectiveness of our proposed upper-bound losses.