Complementary-Label Learning for Arbitrary Losses and Models

Takashi Ishida [1 2] **Gang Niu** [2] **Aditya Krishna Menon** [3] **Masashi Sugiyama** [2 1]

Abstract

In contrast to the standard classification paradigm where the true class is given to each training pattern, *complementary-label learning* only uses training patterns each equipped with a complementary label, which only specifies one of the classes that the pattern does *not* belong to. The goal of this paper is to derive a novel framework of complementary-label learning with an unbiased estimator of the classification risk, for arbitrary losses and models—all existing methods have failed to achieve this goal. Not only is this beneficial for the learning stage, it also makes model/hyper-parameter selection (through cross-validation) possible without the need of any ordinarily labeled validation data, while using any linear/non-linear models or convex/non-convex loss functions. We further improve the risk estimator by a non-negative correction and gradient ascent trick, and demonstrate its superiority through experiments.

1. Introduction

Modern classification methods usually require massive data with high-quality labels, but preparing such datasets is unrealistic in many domains. To mitigate the problem, previous works have investigated ways to learn from weak supervision: *semi-supervised learning* (Chapelle et al., 2006; Miyato et al., 2016; Kipf & Welling, 2017; Sakai et al., 2017; Tarvainen & Valpola, 2017; Oliver et al., 2018), *noisy-label learning* (Natarajan et al., 2013; Menon et al., 2015; Patrini et al., 2017; Ma et al., 2018; Han et al., 2018; Charoenphakdee et al., 2019), *positive-unlabeled learning* (Elkan & Noto, 2008; du Plessis et al., 2014; Kiryo et al., 2017), *positive-confidence learning* (Ishida et al., 2018), *similar-unlabeled learning* (Bao et al., 2018), *unlabeled-unlabeled learning* (du Plessis et al., 2013; Lu et al., 2019), and others.

In this paper, we consider learning from another natural type of weak supervision called *complementary-label learning* (Ishida et al., 2017; Yu et al., 2018), where the label only specifies one of the classes that the pattern does *not* belong to. For example, a crowdsourced worker can tell us a pattern does not belong to a certain class, instead of identifying the correct class. In contrast to the ordinary case where the true class is given to each pattern (which often needs to be chosen out of many candidate classes precisely), collecting these complementary labels is obviously much easier and less costly.

Another potential application is collecting survey data that requires extremely private questions (Ishida et al., 2017). It would be less mentally demanding, if we explain to the respondent that we will transform their provided true label to a complementary label, before the data is saved into the database. This might become common in the future where privacy concerns are increasing.

A natural question is, however, is it possible to learn from such complementary labels (without *any* true labels)?

The problem has previously been tackled by Ishida et al. (2017), showing that the classification risk can be recovered only from complementarily labeled data. They also gave theoretical analysis with a statistical consistency guarantee. However, they required strong restrictions on the loss functions, allowing only one-versus-all and pairwise comparison multi-class loss functions (Zhang, 2004), with certain non-convex binary losses. This is a severe limitation since the softmax cross-entropy loss, which cannot be expressed by the two losses above, is the most popular loss in deep learning nowadays.

Later, Yu et al. (2018) proposed a different formulation for complementary labels by employing the forward loss correction technique (Patrini et al., 2017) to adjust the learning objective, but limiting the loss function to softmax cross-entropy loss. Their proposed risk estimator is not necessarily *unbiased* but the minimizer is theoretically guaranteed to be *consistent* with the minimizer of the risk for ordinary labels (under an implicit assumption on the model for convergence analysis). They also extended the problem setting to where complementary labels are chosen in an uneven (biased) way.

In this paper, we first derive an unbiased risk estimator

[1]The University of Tokyo [2]RIKEN [3]Google Research. Correspondence to: Takashi Ishida <ishida@ms.k.u-tokyo.ac.jp>.

Proceedings of the 36^{th} International Conference on Machine Learning, Long Beach, California, PMLR 97, 2019. Copyright 2019 by the author(s).

with a general loss function, making *any* loss functions available for use: not only the softmax cross-entropy loss function but other convex/non-convex loss functions can also be applied. We also do not have implicit assumptions on the classifier, allowing both linear and non-linear models. We also prove that our new framework is a generalization of previous complementary-label learning (Ishida et al., 2017).

Yu et al. (2018) does not have an unbiased risk estimator, which means users will need clean data with true labels to calculate the error rate during the validation process. On the other hand, our proposed unbiased risk estimator can handle *complementarily* labeled validation data not only for our learning objective, but also for that of Yu et al. (2018). This is helpful since collecting clean data is usually much more expensive. Note that in the example of survey with extremely private questions explained earlier, it may be impossible to even collect a small number of validation data with true labels.

Finally, our proposed unbiased risk estimator has an issue that the classification risk can attain negative values after learning, leading to overfitting. We further propose a non-negative correction to the original unbiased risk estimator to improve our estimator. The modified objective is no longer guaranteed to be an unbiased risk estimator, but the unbiased risk estimator can still be used for validation procedures for this modified learning objective. We experimentally show that our proposed method is comparable to or better than previous methods (Ishida et al., 2017; Yu et al., 2018) in terms of classification accuracy.

A summary of our contributions is as follows:

- We propose a new unbiased risk estimator, allowing usage of any loss (convex, non-convex) and any model (parametric, non-parametric) for complementary-label learning.
- This risk can be used not only as a learning objective, but as a validation criterion even for other methods, such as Ishida et al. (2017) and Yu et al. (2018).
- We further investigate correction schemes to make complementary-label learning practical and demonstrated the performance in experiments.

2. Review of previous works

In this section, we introduce some notations and review the formulations of learning from ordinary labels, learning from complementary labels, learning from ordinary & complementary labels, and learning from partial labels.

2.1. Learning from ordinary labels

Let $\mathcal{X}$ be an instance space and $\mathcal{D}$ be the joint distribution over $\mathcal{X} \times [K]$ for class label set $[K] := \{1, 2, \ldots, K\}$, with random variables $(X, Y) \sim \mathcal{D}$. The data at hand is sampled independently and identically from the joint distribution: $\{(x_i, y_i)\}_{i=1}^n \overset{\text{i.i.d.}}{\sim} \mathcal{D}$. The joint distribution $\mathcal{D}$ can be either decomposed into class-conditionals $\{P_k\}_{k=1}^K$ and base rate $\{\pi_k\}_{k=1}^K$, where $P_k := \mathbb{P}(X|Y = k)$ and $\pi_k := \mathbb{P}(Y = k)$, or the marginal M and class-probability function $\boldsymbol{\eta} : \mathcal{X} \to \Delta_k$, where $M := \mathbb{P}(X)$, $\boldsymbol{\eta}_k(x) := \mathbb{P}(Y = k|X = x)$ and Δ_K is the conditional probability simplex for K classes. A loss is any $\ell : [K] \times \mathbb{R}^K \to \mathbb{R}_+$. The decision function is any $\boldsymbol{g} : \mathcal{X} \to \mathbb{R}^K$ and $\boldsymbol{g}_k(X)$ is the k-th element of $\boldsymbol{g}(X)$. The risk for the decision function $\boldsymbol{g}$ with respect to loss ℓ and implicit distribution $\mathcal{D}$ is:

$$R(\boldsymbol{g}; \ell) : = \mathbb{E}_{(X,Y)\sim\mathcal{D}}[\ell(Y, \boldsymbol{g}(X))], \tag{1}$$

where $\mathbb{E}$ denotes the expectation. Two useful equivalent expressions of classification risk (1) used in later sections are

$$R(\boldsymbol{g}; \ell) = \mathbb{E}_X[\boldsymbol{\eta}(x)^\top \boldsymbol{\ell}(\boldsymbol{g}(X))] = \sum_{k=1}^K \pi_k \mathbb{E}_{\mathbb{P}_k}\Big[\ell(k, \boldsymbol{g}(X))\Big], \tag{2}$$

where,

$$\boldsymbol{\ell}(\boldsymbol{g}(X)) := [\ell(1, \boldsymbol{g}(X)), \ell(2, \boldsymbol{g}(X)), \ldots, \ell(K, \boldsymbol{g}(X))]^\top.$$

The goal of classification is to learn the decision function $\boldsymbol{g}$ that minimizes the risk. In the usual classification case with ordinarily labeled data at hand, approximating the risk empirically is straightforward:

$$\widehat{R}(\boldsymbol{g}; \ell) := \frac{1}{n}\sum_{i=1}^n \ell(y_i, \boldsymbol{g}(x_i)).$$

Some well known multi-class loss functions are one-versus-all and pairwise comparison losses:

$$\ell_{\text{OVA}}\big(k, \boldsymbol{g}(x)\big) = s\big(\boldsymbol{g}_k(x)\big) + \frac{1}{K-1}\sum_{k' \neq k} s\big(-\boldsymbol{g}_{k'}(x)\big), \tag{3}$$

$$\ell_{\text{PC}}\big(k, \boldsymbol{g}(x)\big) = \sum_{k' \neq k} s\big(\boldsymbol{g}_k(x) - \boldsymbol{g}_{k'}(x)\big), \tag{4}$$

where $s(z) : \mathbb{R} \to \mathbb{R}_+$ is a binary loss function.

2.2. Learning from complementary labels

Next we consider the problem of learning from complementary labels (Ishida et al., 2017). We observe patterns each equipped with a complementary label $\{(x_{i'}, \overline{y}_{i'})\}_{i'=1}^{n'}$

Table 1: Comparison of two proposed complementary-label methods with previous works. We first propose a general unbiased risk estimator for complementary labels that has no restrictions on loss functions and models. We next propose a modified non-negative formulation which solves overfitting issues and leads to better experimental results. Even though the non-negative formulation is no longer an unbiased estimator as a learning objective, the unbiased estimator can be used in the validation procedure.

Methods	loss assump. free	model assump. free	unbiased estimator	explicit risk correction
Ishida et al. (2017)	×	✓	✓	×
Yu et al. (2018)	×	×	×	×
Proposed (General formulation)	✓	✓	✓	×
Proposed (Non-negative formulation)	✓	✓	×	✓

sampled independently and identically from a different joint distribution $\overline{\mathcal{D}} \neq \mathcal{D}$. We denote random variables as $(X, \overline{Y}) \sim \overline{\mathcal{D}}$. As before, we assume this distribution can be decomposed into either class-conditionals $\{\overline{P}_k\}_{k=1}^K$ and base rate $\{\overline{\pi}\}_{k=1}^K$, or marginal M and class-probability function $\overline{\boldsymbol{\eta}} : \mathcal{X} \to \Delta_K$, where $\overline{P}_k := \mathbb{P}(X|\overline{Y} = k)$, $\overline{\pi}_k := \mathbb{P}(\overline{Y} = k)$, $M := \mathbb{P}(X)$, $\overline{\boldsymbol{\eta}}_k(x) := \mathbb{P}(\overline{Y} = k|X = x)$, and $\overline{Y}$ is the complementary label.

Without any assumptions on $\overline{\mathcal{D}}$, it is impossible to design a suitable learning procedure. The assumption for unbiased complementary learning used in Ishida et al. (2017) was

$$\overline{\boldsymbol{\eta}}(x) = \boldsymbol{T}\boldsymbol{\eta}(x), \tag{5}$$

where $\boldsymbol{T} \in \mathbb{R}^{K \times K}$ is a matrix that takes 0 on diagonals and $\frac{1}{K-1}$ on non-diagonals.

This assumption implies all other labels are chosen with uniform probability. This can be forced by designing the data collecting system to first pick up a label randomly and then ask the worker if the data belong to the label with a yes or no. When the answer is no, we will attach that label as the complementary label, and the data will follow the uniform assumption. Under this assumption, Ishida et al. (2017) proved that they can recover the classification risk (1) from an alternative formulation using only complementarily labeled data when the loss function satisfies certain conditions. More specifically, usable loss functions are one-versus-all or pairwise comparison multi-class loss functions (Zhang, 2004):

$$\overline{\ell}_{\text{OVA}}\big(\overline{k}, \boldsymbol{g}(x)\big) = \frac{1}{K-1}\sum_{k \neq \overline{k}} s\big(\boldsymbol{g}_k(x)\big) + s\big(-\boldsymbol{g}_{\overline{k}}(x)\big) \tag{6}$$

$$\overline{\ell}_{\text{PC}}\big(\overline{k}, \boldsymbol{g}(x)\big) = \sum_{k' \neq \overline{k}} s\big(\boldsymbol{g}_k(x) - \boldsymbol{g}_{\overline{k}}(x)\big) \tag{7}$$

each with binary loss function $s(z)$ that satisfies $s(z) + s(-z) = 1$, such as ramp loss $s_{\text{R}}(z) = \frac{1}{2}\max\big(0, \min(2, 1-z)\big)$ or sigmoid loss $s_{\text{S}}(z) = \frac{1}{1+e^z}$.

Having an unbiased risk estimator is also helpful for the validation process. Since we do not have ordinary labels in our validation set in the complementary-label learning setting, we cannot follow the usual validation procedure that uses zero-one error or accuracy. If we have an unbiased estimator of the original classification risk (which can be interpreted as zero-one error), we can use the empirical risk for (cross)-validated complementary data to select the best hyper-parameter or deploy early stopping.

An extension of the above method was considered in Yu et al. (2018) by using a different assumption than (5): there is some bias amongst the possible complementary labels that can be chosen, thus the non-diagonals of $\boldsymbol{T}$ is not restricted to $\frac{1}{K-1}$. However, one will need to estimate $\boldsymbol{T}$ beforehand, which is fairly difficult without strong assumptions. Furthermore, in this setup, it is necessary to encourage the worker to provide more difficult complementary labels, for example, by giving higher rewards to certain classes. Otherwise, the complementary label given by the worker may be too obvious and uninformative. Even though the two assumptions are mathematically similar, the data generation process may be different. In this paper we focus on the former assumption.

Unlike Ishida et al. (2017), Yu et al. (2018) did not directly provide a risk estimator, but they showed that the *minimizer* of their learning objective agrees with the minimizer of the original classification risk (1). Note that, in their formulation, the loss function is restricted to the softmax cross-entropy loss. Furthermore, the use of a highly non-linear model is supposed for consistency guarantee in their theoretical analysis. Since the learning objective of Yu et al. (2018) does not correspond to the classification risk, one will need clean data with true labels to calculate the error rate during the validation process. On the other hand, our proposed risk estimator in this paper can cope with *complementarily* labeled validation data not only for our own learning objective, but can be used to select hyper-parameters for others such as Yu et al. (2018).

2.3. Learning from ordinary & complementary labels

In many practical situations, we may also have ordinarily labeled data in addition to complementarily labeled data. Ishida et al. (2017) touched on the idea of crowdsourcing for an application with both types of data. For example, we may choose one of the classes randomly by following the uniform distribution, with probability $\frac{1}{K-1}$ for each class, and ask crowdworkers whether a pattern belongs to the chosen class or not. Then the pattern is treated as ordinarily labeled if the answer is yes; otherwise, the pattern is regarded as complementarily labeled. If the true label was y for a pattern, we can naturally assume that the crowdworker will answer yes by $\mathbb{P}(Y=y|X=x)$ and no by $1-\mathbb{P}(Y=y|X=x)$. This way, ordinarily labeled data can be regarded as patterns from $\mathcal{D}$, and complementarily labeled data from $\overline{\mathcal{D}}$, justifying the assumption of unbiased complementary learning (5).

In Ishida et al. (2017), they considered a convex combination of the classification risks derived from ordinarily labeled data and complementarily labeled data:

$$\alpha\overline{R}(\boldsymbol{g};\overline{\ell})+(1-\alpha)R(\boldsymbol{g};\ell),$$

where $\overline{R}(\boldsymbol{g};\overline{\ell})=\mathbb{E}_{(X,\overline{Y})\sim\overline{\mathcal{D}}}[\overline{\ell}(\overline{Y},\boldsymbol{g}(X))]$ and $\alpha\in[0,1]$ is a hyper-parameter that interpolates between the two risks. The combined (also unbiased) risk estimator can utilize both kinds of data in order to obtain better classifiers, which was demonstrated to perform well in experiments.

2.4. Learning from partial labels

In *learning from partial labels* (Cour et al., 2011), a candidate set of labels (which includes the correct class) is given to each pattern. A different way to view complementary label is a candidate set that includes every class except the complementary label. Even though the proposed method of Cour et al. (2011) shows statistical consistency, it does not give an unbiased estimator of the classification risk. Further, it has different assumptions, e.g., dominance relation, while Ishida et al. (2017) and this paper focus on assumption (5) with different data generation process and applications.

3. Proposed method

As discussed in the previous section, the method by Ishida et al. (2017) works well in practice, but it has restriction on the loss functions—the popular softmax cross-entropy loss is not allowed. On the other hand, the method by Yu et al. (2018) allows us to use the softmax cross-entropy loss, but it does not directly provide an estimator of the classification risk and thus model selection is problematic in practice.

We first describe our general unbiased risk formulation in Section 3.1. Then we discuss how the estimator can be further improved in Section 3.2. Thirdly, we propose a way for our risk estimator to avoid overfitting by a *non-negative risk estimator* in Section 3.3. Finally, we show practical implementation of our risk estimator with stochastic optimization methods in Section 3.4.

3.1. General risk formulation

First, we describe our general unbiased risk formulation. We give the following theorem, which allows unbiased estimation of the classification risk from complementarily labeled patterns:

Theorem 1. *For any ordinary distribution $\mathcal{D}$ and complementary distribution $\overline{\mathcal{D}}$ related by (5) with decision function $\boldsymbol{g}$, and loss ℓ, we have*

$$R(\boldsymbol{g};\ell)=\overline{R}(\boldsymbol{g};\overline{\ell})=\mathbb{E}_{(X,\overline{Y})\sim\overline{\mathcal{D}}}[\overline{\ell}(\overline{Y},\boldsymbol{g}(X))], \quad (8)$$

for the complementary loss

$$\overline{\boldsymbol{\ell}}\big(\boldsymbol{g}(x)\big):=\Big(-(K-1)\boldsymbol{I}_K+\boldsymbol{1}\boldsymbol{1}^\top\Big)\cdot\boldsymbol{\ell}\big(\boldsymbol{g}(x)\big), \quad (9)$$

or equivalently,

$$\overline{\ell}\big(k,\boldsymbol{g}(x)\big)=-(K-1)\cdot\ell\big(k,\boldsymbol{g}(x)\big)+\sum_{j=1}^{K}\ell\big(j,\boldsymbol{g}(x)\big), \quad (10)$$

where $\boldsymbol{I}_K$ is a $K\times K$ identity matrix and $\boldsymbol{1}$ is a K-dimensional column vector with 1 in each element.

Proof can be found in Appendix A. It is worth noting that, in the above derivation, there are no constraints on the loss function and classifier. Thus, we can use any loss (convex/non-convex) and any model (linear/non-linear, parametric/non-parametric) for complementary learning.

Next, we show the relationship between our proposed framework and previous complementary-label learning (Ishida et al., 2017).

Corollary 2. *If one-versus-all loss (6) or pairwise comparison loss (7) is used with binary loss function that satisfy $s(z)+s(-z)=1$, the classification risk can be written as,*

$$\overline{R}(\boldsymbol{g};\overline{\ell})=(K-1)\mathbb{E}_{\overline{\mathcal{D}}}\big[\overline{\ell}\big(\overline{Y},\boldsymbol{g}(X)\big)\big]-M_1+M_2, \quad (11)$$

where M_1 and M_2 are non-negative constants that satisfy $\sum_{\overline{y}=1}^{K}\overline{\ell}\big(\overline{y},\boldsymbol{g}(x)\big)=M_1$ for all x and $\overline{\ell}\big(\overline{y},\boldsymbol{g}(x)\big)+\ell\big(\overline{y},\boldsymbol{g}(x)\big)=M_2$ for all x and $\overline{y}$.

Proof can be found in Appendix B. Since this is equivalent to the first two Theorems in Ishida et al. (2017), our proposed version is a generalization of the previous unbiased complementary-label learning framework.

The key idea of the proof in Theorem 1 is to not rely on the condition that $\sum_{k=1}^{K}\overline{\ell}\big(k,\boldsymbol{g}(x)\big)$ is a constant for all

x, used in Ishida et al. (2017), which is inspired by the property of binary 0-1 loss s_{0-1}, where $s_{0-1}(z)$ is 1 if $z < 0$ and 0 otherwise. Such a technique was also used when designing unbiased risk estimators for learning from positive and unlabeled data in a binary classification setup (du Plessis et al., 2014), but was later shown to be unnecessary (du Plessis et al., 2015). Note that Theorem 1 can be regarded as a special case of a framework proposed for learning from weak labels (Cid-Sueiro et al., 2014).

By using (10), the classification risk can be written as

$$R(\boldsymbol{g};\ell) = \sum_{k=1}^{K} \overline{\pi}_k \mathbb{E}_{\overline{P}_k}\Big[-(K-1)\cdot\ell\big(k,\boldsymbol{g}(X)\big) + \sum_{j=1}^{K}\ell\big(j,\boldsymbol{g}(X)\big)\Big]. \tag{12}$$

Here, we rearrange our complementarily labeled dataset as $\{\mathcal{X}_k\}_{k=1}^{K}$, where $\mathcal{X}_k$ denotes the samples complementarily labeled as class k. Then, this expression of the classification risk can be approximated by,

$$\widehat{R}(\boldsymbol{g};\ell) = \sum_{k=1}^{K} \frac{\widehat{\overline{\pi}}_k}{|\mathcal{X}_k|} \sum_{x_i\in\mathcal{X}_k}\Big[-(K-1)\cdot\ell\big(k,\boldsymbol{g}(x_i)\big) + \sum_{j=1}^{K}\ell\big(j,\boldsymbol{g}(x_i)\big)\Big], \tag{13}$$

where n_k is the number of patterns complementarily labeled as the kth class.

3.2. Necessity of risk correction

The original expression of the classification risk (1) includes an expectation over non-negative loss $\ell : [K] \times \mathbb{R}^K \to \mathbb{R}_+$, so the risk and its empirical approximator are both lower-bounded by zero. On the other hand, the expression (12) derived above contains a negative element. Although (12) is still non-negative by definition, due to the negative term, its empirical estimator can go negative, leading to over-fitting.

We elaborate on this issue with an illustrative numerical example. In the left graph of Figure 1, we show an example of training a linear model trained on the handwritten digits dataset MNIST[1], with complementary labels generated to satisfy (5). We used *Adam* (Kingma & Ba, 2015) for optimization with learning rate $5\mathrm{e}-5$, mini-batch size of 100, and weight decay of $1\mathrm{e}-4$ with 300 epochs. The empirical classification risk (13) is shown in black. We can see that the empirical classification risk continues decreasing and can go below zero at around 100 epochs. The test accuracy on the right graph hits the peak also at around epoch 100 and then the accuracy gradually deteriorates.

This issue stands out even more significantly when we use a flexible model. The middle graph shows the empirical classification risk for a multilayer perceptron (MLP) with one hidden layer (500 units), where *ReLU* (Nair & Hinton, 2010) was used as the activation function. The optimization setup was the same as the case of the linear model above. We can see the empirical risk decreasing much more quickly and going negative. Correspondingly, as the right graph shows, the test accuracy drops significantly after the empirical risk goes negative.

In fact, a similar issue is already implicit in the original paper by Ishida et al. (2017): According to Corollary 2 (or Theorem 1 in Ishida et al. (2017)), the unbiased risk estimator includes subtraction of a positive constant term which increases with respect to the number of classes. This means that the learning objective of Ishida et al. (2017) has a (negative) lower bound.

3.3. Non-negative risk estimator

As we saw in Section 3.2, our risk estimator can suffer from overfitting due to the non-negative issue. Here, we propose a correction to the risk estimator to overcome this problem.

Each term in the risk with ordinary labels (right-hand side of (2)), which corresponds to each class, is non-negative. We can reformulate (12) in order to show the counterpart for each non-negative term in the right-hand side of (2) for complementarily labeled data as

$$R(\boldsymbol{g};\ell) = \sum_{k=1}^{K}\Big[-(K-1)\overline{\pi}_k\cdot\mathbb{E}_{\overline{P}_k}\big[\ell\big(k,\boldsymbol{g}(X)\big)\big] + \sum_{j=1}^{K}\overline{\pi}_j\cdot\mathbb{E}_{\overline{P}_j}\big[\ell\big(k,\boldsymbol{g}(X)\big)\big]\Big]. \tag{14}$$

These counterparts (14) were originally non-negative when ordinary labels were used. In the left and middle graphs of Figure 1, we plot the decomposed risks with respect to each *ordinary* class (14) (shown in different colors). We can see that the decomposed risks for all classes become negative eventually. Based on this observation, our basic idea for correction is to enforce non-negativity for each ordinary class, with the expression based on complementary labels. More specifically, we propose a non-negative version by

$$\sum_{k=1}^{K}\max\Big\{0, \Big[-(K-1)\overline{\pi}_k\cdot\mathbb{E}_{\overline{P}_k}\big[\ell\big(k,\boldsymbol{g}(X)\big)\big] + \sum_{j=1}^{K}\overline{\pi}_j\cdot\mathbb{E}_{\overline{P}_j}\big[\ell\big(k,\boldsymbol{g}(X)\big)\big]\Big]\Big\}. \tag{15}$$

(15) is equivalent to (14), since $\max\{0,a\} = a$ if a is non-negative. By using the datasets used for (13), this non-negative risk can be naïvely approximated by the sample

[1] See http://yann.lecun.com/exdb/mnist/.

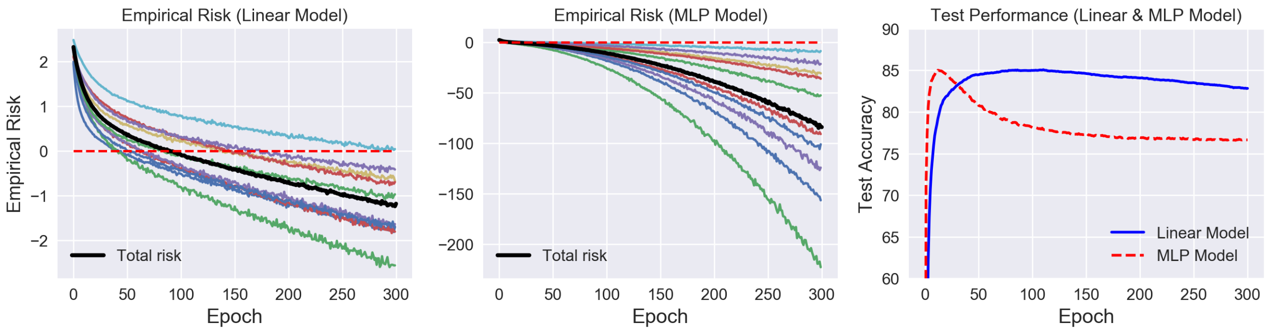


Figure 1: The left and middle graphs shows the total risk (13) (in black color) and the risk decomposed into each *ordinary* class term (14) (in other colors) for training data with linear and MLP models, respectively. The right graph shows the corresponding test accuracy for both models.

average as

$$\sum_{k=1}^{K} \max\Big\{0, \Big[-(K-1)\cdot\frac{\overline{\pi}_k}{|\mathcal{X}_k|}\sum_{x_i\in\mathcal{X}_k}\ell(k,\boldsymbol{g}(x_i)) + \sum_{j=1}^{K}\frac{\overline{\pi}_j}{|\mathcal{X}_j|}\sum_{x_{i'}\in\mathcal{X}_j}\ell(k,\boldsymbol{g}(x_{i'}))\Big]\Big\}. \tag{16}$$

The empirical version of (14) may suffer from a negative objective, but (16) is non-negative (even though their population versions are equivalent.)

Enforcing the reformulated risk to become non-negative was previously explored in Kiryo et al. (2017), in the context of binary classification from positive and unlabeled data. The positive class risk is already bounded below by zero in their case (because they have true positive labels), so there was a max operator only on the negative class risk. We follow their footsteps, but since our setting is a multi-class scenario and also differs by not having *any* true labels, we put a max operator on each of the K classes.

3.4. Approximate non-negative risk estimator

Implementation with max operator We now illustrate how to design a practical implementation under stochastic optimization for our non-negative risk estimator. An unfortunate issue is that the minimization of (16) is not point-wise due to the max-operator, thus cannot be used directly for stochastic optimization methods with mini-batch. However, an upper bound of the risk can be minimized in parallel by using mini-batch as the following,

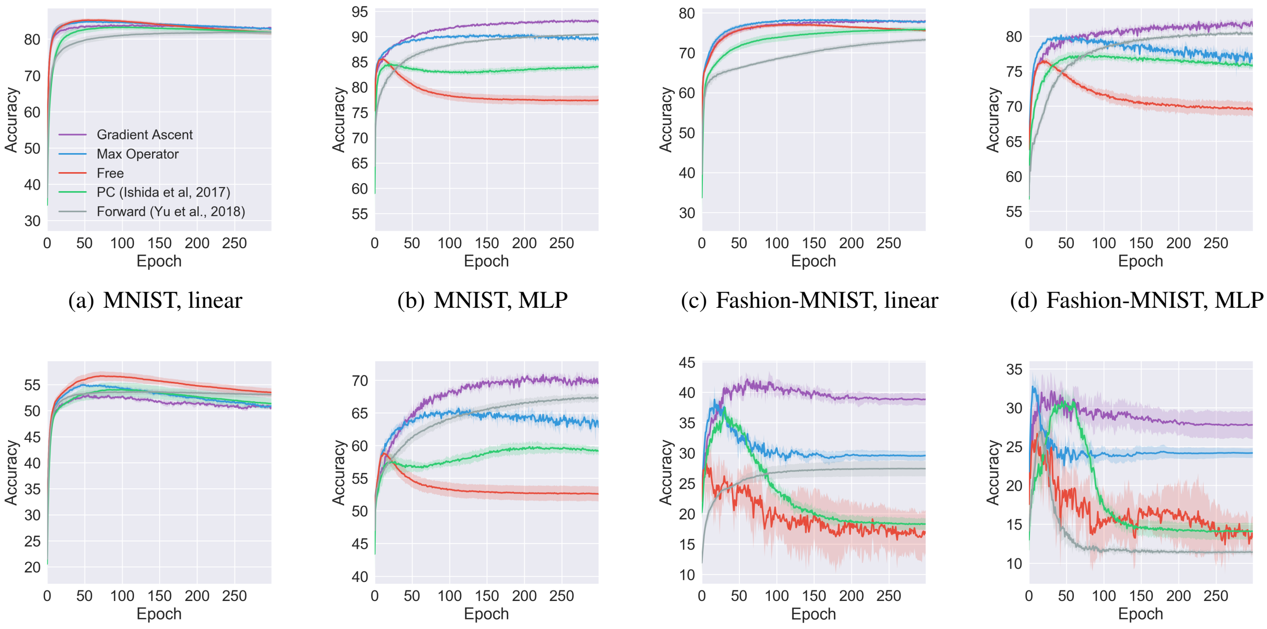
$$\frac{1}{B}\sum_{b=1}^{N}\sum_{k=1}^{K}\max\Big\{0, -(K-1)\overline{\pi}_k\cdot\widehat{\mathbb{E}}_{\overline{P}_k}\big[\ell\big(k,\boldsymbol{g}(X)\big);\mathcal{X}_k^b\big] + \sum_{j=1}^{K}\overline{\pi}_j\cdot\widehat{\mathbb{E}}_{\overline{P}_j}\big[\ell\big(k,\boldsymbol{g}(X)\big);\mathcal{X}_j^b\big]\Big\}, \tag{17}$$

Algorithm 1 Complementary-label learning with gradient ascent

```
Input: complementarily labeled training data $\{\mathcal{X}_k\}_{k=1}^K$, where $\mathcal{X}_k$ denotes the samples complementarily labeled as class $k$;
Output: model parameter $\theta$ for $g(x;\theta)$
Let $\mathcal{A}$ be an external SGD-like stochastic optimization algorithm such as Kingma & Ba (2015)
while no stopping criterion has been met:
    Shuffle $\{\mathcal{X}_j\}_j^K$ into $B$ mini-batches;
    for $b = 1$ to $B$:
        Denote $\{\mathcal{X}_j^b\}$ as the $b$-th mini-batch for complementary class $j$
        Denote $r_k^b(\theta) = -(K-1)\overline{\pi}_k\cdot\widehat{\mathbb{E}}_{\overline{P}_k}[\ell(k,\boldsymbol{g});\mathcal{X}_k^b] + \sum_{j=1}^K\overline{\pi}_j\cdot\widehat{\mathbb{E}}_{\overline{P}_j}[\ell(k,\boldsymbol{g});\mathcal{X}_j^b]$
        if $\min_k[r_1^b(\theta),\ldots,r_k^b(\theta),\ldots,r_K^b(\theta)] > -\beta$:
            Denote $L^b(\theta) = \sum_{k=1}^K r_k^b(\theta)$
            Set gradient $\nabla_\theta L^b(\theta)$;
            Update $\theta$ by $\mathcal{A}$ with its current step size $\eta$;
        else:
            Denote $\widetilde{L}^b(\theta) = \sum_{k=1}^K\min\{-\beta, r_k^b(\theta)\}$
            Set gradient $-\nabla_\theta\widetilde{L}^b(\theta)$;
            Update $\theta$ by $\mathcal{A}$ with a discounted step size $\gamma\eta$;
```

where $\widehat{\mathbb{E}}$ is the empirical version of the expectation and B is the number of mini-batches.

Implementation with gradient ascent If the objective is negative for a certain mini-batch, the previous implementation based on the max operator will prevent the objective to further *decrease*. However, if the objective is already negative, that mini-batch has already started to overfit. The max operator cannot contribute to decrease the degree of overfitting. From this perspective, there is still room to improve the overfitting issue, and it would be preferable to *increase* itself to make this mini-batch less overfitted.



(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, DenseNet

(h) CIFAR-10, ResNet

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

Our idea is the following. We denote the risk that corresponds to the kth ordinary class for the ith mini-batch as

$$r_k^b(\theta) = -(K-1)\overline{\pi}_k \cdot \widehat{\mathbb{E}}_{\overline{P}_k}[\ell(k, \boldsymbol{g}(X)); \mathcal{X}_k^b] + \sum_{j=1}^{K} \overline{\pi}_j \cdot \widehat{\mathbb{E}}_{\overline{P}_j}\big[\ell(k, \boldsymbol{g}(X)); \mathcal{X}_j^b\big],$$

and the total risk as $L^b(\theta) = \sum_{k=1}^{K} r_k^b(\theta)$. When $\min_k\{r_k^b(\theta)\}_{k=1}^K \geq -\beta$, we conduct gradient descent as usual with gradient $\nabla_\theta L^b(\theta)$. On the other hand, if $\min_k\{r_k^b(\theta)\}_{k=1}^K < -\beta$, we first squash the class-decomposed risks over $-\beta$ to $-\beta$ with a min operator, and then sum the results:

$$\widetilde{L}^b(\theta) = \sum_{k=1}^{K} \min\{-\beta, r_k^b(\theta)\}.$$

Next we set the gradient in the opposite direction with $-\nabla_\theta \widetilde{L}^b(\theta)$. Conceptually, we are going *up* the gradient $\nabla_\theta \widetilde{L}^b(\theta)$ for *only* the class-decomposed risks below $-\beta$, to avoid the class-decomposed risks that are already large to further increase. Note that β is a hyper-parameter that controls the tolerance of negativity. $\beta = 0$ would mean there is zero tolerance, but in practice we can also have $-\beta \neq 0$ for a threshold that allows some negative ($-\beta < 0$) or positive ($-\beta > 0$) amount. The procedure is shown in detail in Algorithm 1.

4. Experiments

In this section, we compare the 3 methods that we have proposed in Section 3, which are *Free* (Unbiased risk estimator that is loss assumption free, based on Eq. (13)), *Max Operator* (based on Eq. (17)), and *Gradient Ascent* (based on Alg.1). For *Gradient Ascent*, we used $\beta = 0$ and $\gamma = 1$ for simplicity. Mini-batch size was set to 256. We also compare with two baseline methods: Pairwise comparison (*PC*) with ramp loss from Ishida et al. (2017) and *Forward* correction from Yu et al. (2018). For training, we used only complementarily labeled data, which was generated so that the assumption of (5) is satisfied. This is straightforward when the dataset has a uniform (ordinarily-labeled) class prior, because it reduces to just choosing a class randomly other than the true class.

In Appendix C, we explain the details of the datasets used in the experiments: MNIST, Fashion-MNIST, Kuzushiji-MNIST, and CIFAR-10. The implementation is based on Pytorch[2] and our demo code is available online[3].

4.1. Comparison of all epochs during training

Setup For MNIST, Fashion-MNIST, and Kuzushiji-MNIST, a linear-in-input model with a bias term and a MLP model ($d - 500 - 1$) was trained with softmax cross-entropy

[2] https://pytorch.org

[3] https://github.com/takashiishida/comp

Table 2: Test mean and standard deviation of the classification accuracy for 4 trials. Method name outside (inside) parenthesis shows the criterion of training (validation) objective. Best is shown in **bold** or underline for column 2∼4 or column 2∼6, respectively.

Dataset	*GA (Free)*	*PC (PC)*	*Fwd (Fwd)*	*PC (Free)*	*Fwd (Free)*
MNIST	$88.1 \pm 2.5\%$	$79.3 \pm 3.3\%$	$\mathbf{88.7 \pm 0.3\%}$	$80.2 \pm 2.9\%$	$\underline{89.4 \pm 0.4\%}$
Fashion	$\underline{\mathbf{78.7 \pm 1.4\%}}$	$74.7 \pm 1.6\%$	$77.5 \pm 1.2\%$	$75.7 \pm 1.2\%$	$73.5 \pm 5.5\%$
Kuzushiji	$\mathbf{63.8 \pm 1.1\%}$	$56.7 \pm 4.9\%$	$62.0 \pm 1.1\%$	$56.1 \pm 4.2\%$	$\underline{65.4 \pm 1.7\%}$
CIFAR-10	$\underline{\mathbf{36.8 \pm 0.6\%}}$	$33.4 \pm 2.0\%$	$30.8 \pm 1.6\%$	$25.9 \pm 7.6\%$	$30.8 \pm 1.7\%$

loss function (except *PC*) for 300 epochs. Weight decay of $1\mathrm{e}-4$ for weight parameters and learning rate of $5\mathrm{e}-5$ for Adam (Kingma & Ba, 2015) was used.

For CIFAR-10, DenseNet (Huang et al., 2017) and ResNet-34 (He et al., 2016) were used with weight decay of $5\mathrm{e}-4$ and initial learning rate of $1\mathrm{e}-2$. For optimization, stochastic gradient descent was used with the momentum set to 0.9. Learning rate was halved every 30 epochs.

Results We show the accuracy for all 300 epochs on test data to demonstrate how the issues discussed in Section 3.2 appear and how different implementations in Section 3.4 are effective. In Figure 2, we show the mean and standard deviation of test accuracy for 4 trials on test data evaluated with ordinary labels.

First we compare our 3 proposed methods with each other. For linear models in MNIST, Fashion-MNIST, and Kuzushiji-MNIST, all proposed methods work similarly. However in the case of using a more flexible MLP model or using DenseNet/ResNet in CIFAR-10, we can see that *Free* is the worst, *Max Operator* is better and *Gradient Ascent* is the best out of the proposed three methods for most of the epochs (*Free* < *Max Operator* < *Gradient Ascent*). These results are consistent with the discussions of overfitting in Section 3.2 and the motivations for different implementations in Section 3.4.

Next, we compare with baseline methods. For linear models, all methods have similar performance. However for deep models (MLP, DenseNet, and ResNet), the superiority stands out for *Gradient Ascent* for all datasets.

4.2. Experiments with validation process

Setup Next, we perform experiments with a train, validation, and test split. The dataset is constructed by splitting the original training data used in the previous experiments into train/validation with a 9:1 ratio. Note that the validation data only has complementary labels since it is splitted from the set of complementarily labeled training data. We use the same MLP models for MNIST, Fashion-MNIST, and Kuzushiji-MNIST. We use DenseNet for CIFAR-10.

Since *Gradient Ascent (GA)* seemed to work better than *Free* and *Max Operator* previously, we omit *Free* and *Max Operator* and compare *GA* with baseline methods (*PC* and *Forward(Fwd)*). For the validation objective, we used the corresponding criterion for each method, which is shown in the first 3 columns with parenthesis, in Table 2. We also conducted experiments using our proposed general unbiased estimator *Free* as the validation criterion for baseline methods (*PC* and *Fwd*), which is shown in the last 2 columns in Table 2. SGD with momentum of 0.9 was used for 250 epochs. Weight-decay was fixed to $1e-4$ and learning rate candidates are $\{1\mathrm{e}-4, 5\mathrm{e}-4, 1\mathrm{e}-3, 5\mathrm{e}-3, 1\mathrm{e}-2, 5\mathrm{e}-2\}$ for CIFAR-10 and $\{5\mathrm{e}-5, 1\mathrm{e}-4, 5\mathrm{e}-4, 1\mathrm{e}-3, 5\mathrm{e}-3, 1\mathrm{e}-2\}$ for other datasets. For CIFAR-10, we added learning rate decay with the same settings from Section 4.1.

Results In Table 2, we showed the mean and standard deviation of test accuracy for 4 trials, with the model that gave the best validation score out of all epochs for all hyper-parameter candidates. By comparing the first 3 columns, *GA* seems to work well. We can also observe that in most cases, *PC (Free)* and *Fwd (Free)* performs similarly or better than *PC (PC)* and *Fwd (Fwd)*, respectively. This confirms the discussion in earlier sections that our general unbiased risk estimator is useful not only as a learning objective, but also useful as a validation objective for baseline methods.

5. Conclusion

We first proposed a general risk estimator for learning from complementary labels that does not require restrictions on the form of the loss function or the model. However, since the proposed method suffers from overfitting, we proposed a modified version to alleviate this issue in two ways and have better performance. At last, we conducted experiments to show our proposed method outperforms or is comparable to current state-of-the-art methods for various benchmark datasets and for both linear and deep models.

Recently, *complementary-label learning* has been applied to *online learning* (Kaneko et al., 2019), *generative discriminative learning* (Xu et al., 2019), and *medical image segmentation* (Rezaei et al., 2019). This implies applying the idea of complementary labels to other domains may be useful, which can be an interesting future direction.

Acknowledgments

TI was supported by Sumitomo Mitsui DS Asset Management. MS was supported by JST CREST JPMJCR1403. We thank the anonymous reviewers for the helpful suggestions.

References

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

Chapelle, O., Schölkopf, B., and Zien, A. (eds.). *Semi-Supervised Learning*. MIT Press, 2006.

Charoenphakdee, N., Lee, J., and Sugiyama, M. On symmetric losses for learning from corrupted labels. In *ICML*, 2019.

Cid-Sueiro, J., García-García, D., and Santos-Rodríguez, R. Consistency of losses for learning from weak labels. In *ECML-PKDD*, 2014.

Clanuwat, T., Bober-Irizar, M., Kitamoto, A., Lamb, A., Yamamoto, K., and Ha, D. Deep learning for classical Japanese literature. In *NeurIPS Workshop on Machine Learning for Creativity and Design*, 2018.

Cour, T., Sapp, B., and Taskar, B. Learning from partial labels. *Journal of Machine Learning Research*, 12:1501–1536, 2011.

du Plessis, M. C., Niu, G., and Sugiyama, M. Clustering unclustered data: Unsupervised binary labeling of two datasets having different class balances. In *TAAI*, 2013.

du Plessis, M. C., Niu, G., and Sugiyama, M. Analysis of learning from positive and unlabeled data. In *NIPS*, 2014.

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

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

Guo, C., Pleiss, G., Sun, Y., and Weinberger, K. Q. On calibration of modern neural networks. In *ICML*, 2017.

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

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

Huang, G., Liu, Z., van der Maaten, L., and Weinberger, K. Q. Densely connected convolutional networks. In *CVPR*, 2017.

Ishida, T., Niu, G., Hu, W., and Sugiyama, M. Learning from complementary labels. In *NIPS*, 2017.

Ishida, T., Niu, G., and Sugiyama, M. Binary classification from positive-confidence data. In *NeurIPS*, 2018.

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. L. Adam: A method for stochastic optimization. In *ICLR*, 2015.

Kipf, T. N. and Welling, M. Semi-supervised classification with graph convolutional networks. In *ICLR*, 2017.

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

Lecun, Y., Bottou, L., Bengio, Y., and Haffner, P. Gradient-based learning applied to document recognition. In *Proceedings of the IEEE*, pp. 2278–2324, 1998.

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.

Ma, X., Wang, Y., Houle, M. E., Zhou, S., Erfani, S. M., Xia, S.-T., Wijewickrema, S., and Bailey, J. Dimensionality-driven learning with noisy labels. In *ICML*, 2018.

Menon, A. K., van Rooyen, B., Ong, C. S., and Williamson, R. C. Learning from corrupted binary labels via class-probability estimation. In *ICML*, 2015.

Miyato, T., Maeda, S., Koyama, M., Nakae, K., and Ishii, S. Distributional smoothing with virtual adversarial training. In *ICLR*, 2016.

Nair, V. and Hinton, G. E. Rectified linear units improve restricted Boltzmann machines. In *ICML*, 2010.

Natarajan, N., Dhillon, I. S., Ravikumar, P., and Tewari, A. Learning with noisy labels. In *NIPS*, 2013.

Oliver, A., Odena, A., Raffel, C., Cubuk, E. D., and Goodfellow, I. J. Realistic evaluation of deep semi-supervised learning algorithms. In *NeurIPS*, 2018.

Patrini, G., Rozza, A., Menon, A. K., Nock, R., and Qu, L. Making deep neural networks robust to label noise: A loss correction approach. In *CVPR*, 2017.

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*, 2017.

Tarvainen, A. and Valpola, H. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *NIPS*, 2017.

Torralba, A., Fergus, R., and Freeman, W. T. 80 million tiny images: A large data set for nonparametric object and scene recognition. In *IEEE Trans. PAMI*, 2008.

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

Xu, Y., Gong, M., Chen, J., Liu, T., Zhang, K., and Batmanghelich, K. Generative-discriminative complementary learning. *arXiv preprint arXiv:1904.01612*, 2019.

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

Zhang, T. Statistical analysis of some multi-category large margin classification methods. *Journal of Machine Learning Research*, 5:1225–1251, 2004.

A. Proof of Theorem 1

Proof. First of all,

$$
\begin{aligned}
\mathbb{P}(X,\overline{Y}=\overline{y}) &= \frac{1}{K-1}\sum_{y\neq\overline{y}}\mathbb{P}(X,Y=y)\\
&= \frac{1}{K-1}\Big(\sum_{y=1}^{K}\mathbb{P}(X,Y=y)-\mathbb{P}(X,Y=\overline{y})\Big)\\
&= \frac{1}{K-1}\big(\mathbb{P}(X)-\mathbb{P}(X,Y=\overline{y})\big).
\end{aligned}
$$

The first equality holds since the marginal distribution is equivalent for $\mathcal{D}$ and $\overline{\mathcal{D}}$ and we assume (5). Consequently,

$$
\begin{aligned}
\mathbb{P}(\overline{Y}=\overline{y}|X=x) &= \frac{\mathbb{P}(X=x,\overline{Y}=\overline{y})}{\mathbb{P}(X=x)}\\
&= \frac{1}{K-1}\cdot\Big(1-\frac{\mathbb{P}(X,Y=\overline{y})}{\mathbb{P}(X=x)}\Big)\\
&= \frac{1}{K-1}\cdot\big(1-\mathbb{P}(Y=\overline{y}|X=x)\big)\\
&= -\frac{1}{K-1}\mathbb{P}(Y=\overline{y}|X=x)+\frac{1}{K-1}.
\end{aligned}
$$

More simply, we have $\boldsymbol{\eta}(x)=-(K-1)\overline{\boldsymbol{\eta}}(x)+\mathbf{1}$. Finally, we transform the classification risk,

$$
\begin{aligned}
R(g;\ell) &= \mathbb{E}_{(X,Y)\sim\mathcal{D}}[\ell(Y,\boldsymbol{g}(X))] = \mathbb{E}_{X\sim M}[\boldsymbol{\eta}^\top\boldsymbol{\ell}(\boldsymbol{g}(X))]\\
&= \mathbb{E}_{X\sim M}\big[\big(-(K-1)\overline{\boldsymbol{\eta}}^\top+\mathbf{1}^\top\big)\boldsymbol{\ell}\big(\boldsymbol{g}(X)\big)\big]\\
&= \mathbb{E}_{X\sim M}\big[-(K-1)\overline{\boldsymbol{\eta}}^\top\boldsymbol{\ell}\big(\boldsymbol{g}(X)\big)+\mathbf{1}^\top\boldsymbol{\ell}\big(\boldsymbol{g}(X)\big)\big]\\
&= \mathbb{E}_{(X,\overline{Y})\sim\overline{\mathcal{D}}}\big[-(K-1)\cdot\ell\big(\overline{Y},\boldsymbol{g}(X)\big)\big]\\
&\qquad\qquad +\mathbf{1}^\top\mathbb{E}_{X\sim M}\big[\boldsymbol{\ell}\big(\boldsymbol{g}(X)\big)\big]\\
&= \sum_{k=1}^{K}\overline{\pi}_k\cdot\mathbb{E}_{X\sim\overline{P}_k}\Big[-(K-1)\cdot\ell\big(k,\boldsymbol{g}(X)\big)\\
&\qquad\qquad\qquad +\mathbf{1}^\top\boldsymbol{\ell}\big(\boldsymbol{g}(X)\big)\Big]\\
&= \overline{R}(g;\overline{\ell})
\end{aligned}
$$

for the complementary loss, $\overline{\ell}(k,\boldsymbol{g}):=-(K-1)\ell(k,\boldsymbol{g})+\mathbf{1}^\top\boldsymbol{\ell}(\boldsymbol{g})$, which concludes the proof. □

B. Proof of Corollary 2

Proof.

$$
\begin{aligned}
\overline{R}(\boldsymbol{g};\overline{\ell}) &= \mathbb{E}_{\overline{\mathcal{D}}}[\overline{\ell}(\overline{Y},\boldsymbol{g}(X))]\\
&= \mathbb{E}_{\overline{\mathcal{D}}}[-(K-1)\ell(\overline{Y},\boldsymbol{g}(X))+\sum_{j=1}^{K}\ell(j,\boldsymbol{g}(X))]\\
&= \mathbb{E}_{\overline{\mathcal{D}}}\big[-(K-1)[M_2-\overline{\ell}(\overline{Y},\boldsymbol{g}(X))]+M_1\big]\\
&= (K-1)\mathbb{E}_{\overline{\mathcal{D}}}[\overline{\ell}(\overline{Y},\boldsymbol{g}(X))]+M_1-(K-1)M_2\\
&= (K-1)\mathbb{E}_{\overline{\mathcal{D}}}[\overline{\ell}(\overline{Y},\boldsymbol{g}(X))]-M_1+M_2
\end{aligned}
$$

Table 3: Summary statistics of benchmark datasets. In the experiments with validation dataset in Section 4.2, train data is further splitted into train/validation with a ratio of 9:1. Fashion is Fashion-MNIST and Kuzushiji is Kuzushiji-MNIST.

Name	# Train	# Test	# Dim	# Classes	Model
MNIST	60k	10k	784	10	Linear, MLP
Fashion	60k	10k	784	10	Linear, MLP
Kuzushiji	60k	10k	784	10	Linear, MLP
CIFAR-10	50k	10k	2,048	10	DenseNet, Resnet

The second equality holds because we use (10). The third equality holds because we are using losses that satisfy $\sum_j \ell(j,\boldsymbol{g}(x))=M_1$ for all x and $\ell(\overline{y},\boldsymbol{g}(x))+\overline{\ell}(\overline{y},\boldsymbol{g}(x))=M_2$ for all x and $\overline{y}$. The 4th equality rearranges terms. The 5th equality holds because $M_1-(K-1)M_2=-M_1+M_2$ for $\overline{\ell}_{\text{OVA}}$ and $\overline{\ell}_{\text{PC}}$. This can be easily shown by using $M_1=K$ and $M_2=2$ for $\overline{\ell}_{\text{OVA}}$, and $M_1=K(K-1)/2$ and $M_2=K-1$ for $\overline{\ell}_{\text{PC}}$. □

C. Datasets

In the experiments in Section 4, we use 4 benchmark datasets explained below. The summary statistics of the four datasets are given in Table 3.

- MNIST[4] (Lecun et al., 1998) is a 10 class dataset of handwritten digits: $1,2\ldots,9$ and 0. Each sample is a 28×28 grayscale image.
- Fashion-MNIST[5] (Xiao et al., 2017) is a 10 class dataset of fashion items: T-shirt/top, Trouser, Pullover, Dress, Coat, Sandal, Shirt, Sneaker, Bag, and Ankle boot. Each sample is a 28×28 grayscale image.
- Kuzushiji-MNIST[6] (Clanuwat et al., 2018) is a 10 class dataset of cursive Japanese ("Kuzushiji") characters. Each sample is a 28×28 grayscale image.
- CIFAR-10[7] is a 10 class dataset of various objects: airplane, automobile, bird, cat, deer, dog, frog, horse, ship, and truck. Each sample is a colored image in $32\times32\times3$ RGB format. It is a subset of the 80 million tiny images dataset (Torralba et al., 2008).

[4] http://yann.lecun.com/exdb/mnist/

[5] https://github.com/zalandoresearch/fashion-mnist

[6] https://github.com/rois-codh/kmnist

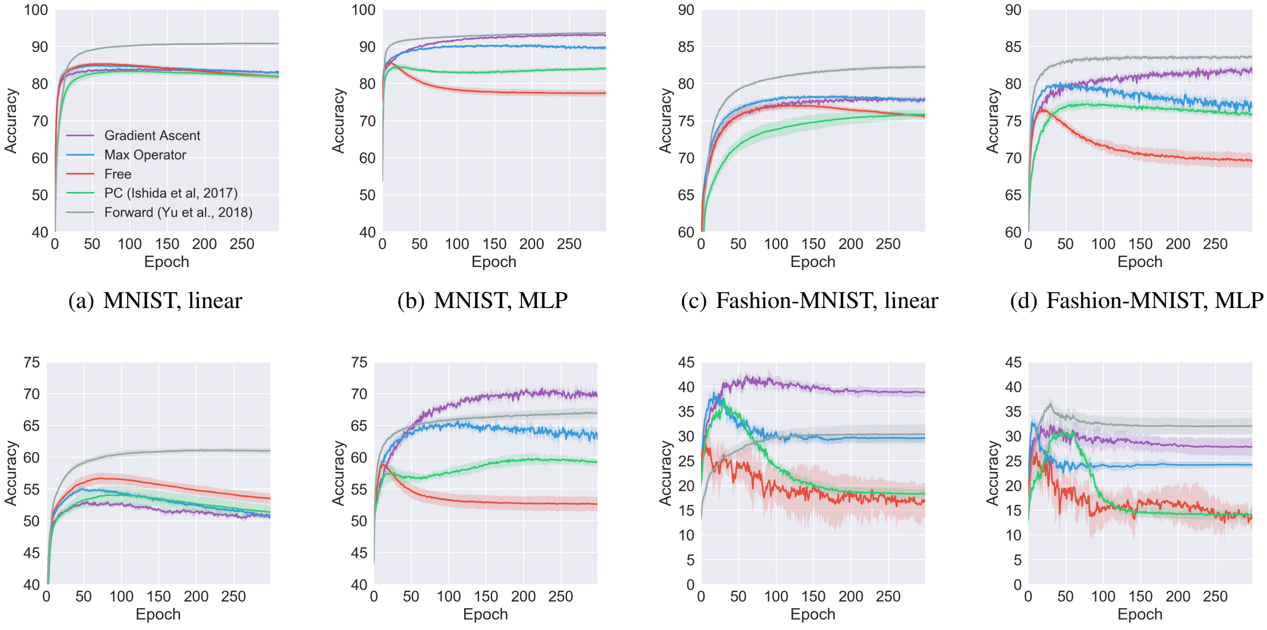
[7] https://www.cs.toronto.edu/~kriz/cifar.html

Errata (Nov. 19, 2019)

We have found errors in our previous implementation for the forward method (Yu et al., 2018), and would like to report the updated results based on the fixed implementation.

In Figure 3, the forward method performs better than previously reported. This is especially true for linear models. For neural network models, the results seem to be dataset-dependent: For MNIST and Fashion-MNIST, the proposed gradient ascent method is similar to the forward method. For Kuzushiji-MNIST, the proposed gradient ascent method is still better than the forward method. For CIFAR-10, the proposed gradient ascent method with DenseNet still performs the best with around 40% accuracy. In Table 4, the proposed gradient ascent method is better for CIFAR-10, but the forward method is better for MNIST and Fashion-MNIST. The two methods perform similarly for Kuzushiji-MNIST.

Addtionally, we investigate the reason behind the good performance of forward methods with a linear model. In Figure 4, we visualize the reliability diagrams (Guo et al., 2017) and histograms of the softmax output of the forward method, for MNIST, Fashion-MNIST, and Kuzushiji-MNIST. We can see that the linear model is much more confidence-calibrated compared to MLP models. The forward method requires the model to be flexible in order to guarantee that the solution gives the true class posterior under the clean joint distribution, given uncountably infinite training data. In the figures, however, we can see that with finite training data, a flexible model can be over-confident (further away from the gray dotted line), while a linear model is more confidence-calibrated (more closer to the gray dotted line).



(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, DenseNet (h) CIFAR-10, ResNet

Figure 3: Updated results of Figure 2. The forward method has been swapped with the fixed implementation.

Table 4: Updated results of Table 2. The results in the 4th and 6th columns have been swapped with the fixed implementation.

Dataset	*GA (Free)*	*PC (PC)*	*Fwd (Fwd)*	*PC (Free)*	*Fwd (Free)*
MNIST	$88.1 \pm 2.5\%$	$79.3 \pm 3.3\%$	$\mathbf{93.3 \pm 0.2\%}$	$80.2 \pm 2.9\%$	$92.2 \pm 0.6\%$
Fashion	$78.7 \pm 1.4\%$	$74.7 \pm 1.6\%$	$\mathbf{82.7 \pm 0.4\%}$	$75.7 \pm 1.2\%$	$82.5 \pm 0.6\%$
Kuzushiji	$63.8 \pm 1.1\%$	$56.7 \pm 4.9\%$	$\mathbf{64.1 \pm 0.4\%}$	$56.1 \pm 4.2\%$	$63.9 \pm 1.9\%$
CIFAR-10	$\mathbf{36.8 \pm 0.6\%}$	$33.4 \pm 2.0\%$	$33.2 \pm 0.9\%$	$25.9 \pm 7.6\%$	$33.7 \pm 1.1\%$

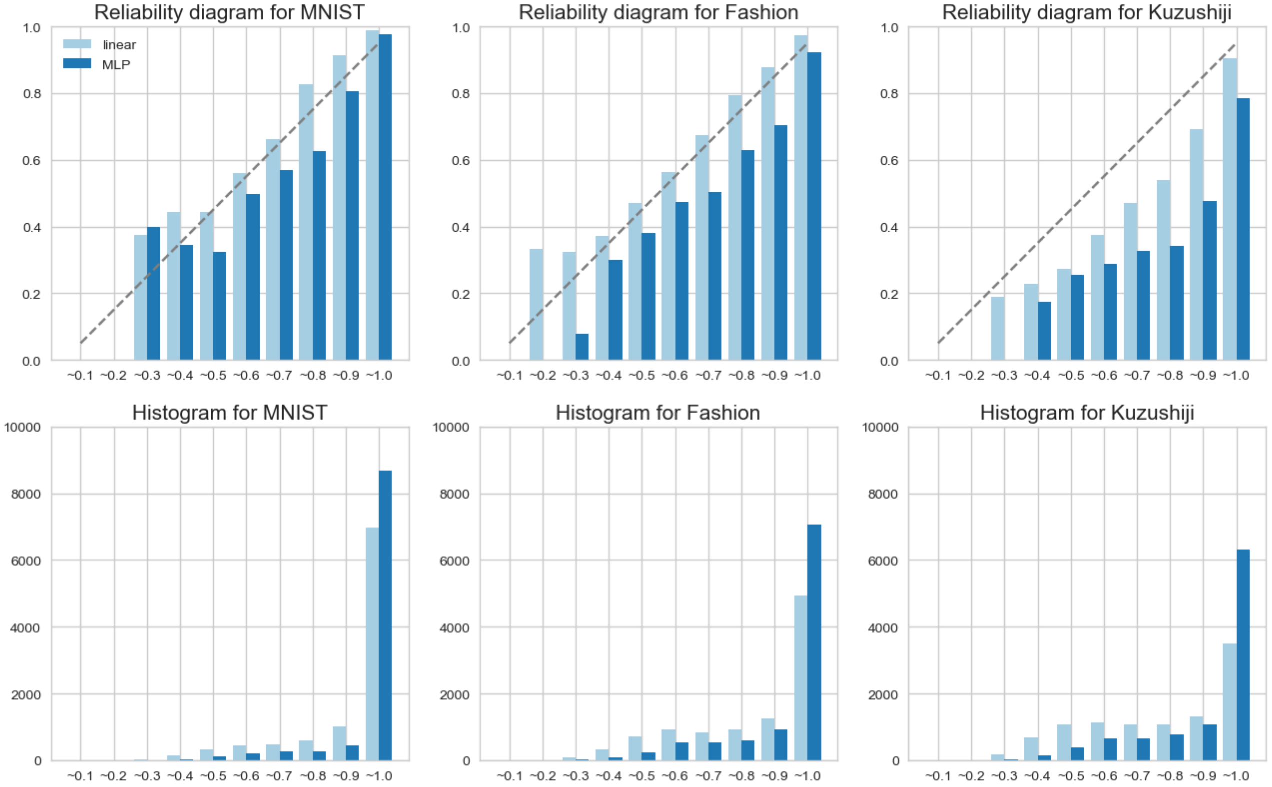


Figure 4: The bottom figures show the histogram of the output of the softmax layer in the forward method, with 10 bins in the horizontal axis, for MNIST, Fashion-MNIST, and Kuzushiji-MNIST. The light blue color shows the linear model and the dark blue color shows the MLP model. The top figures show the reliability diagrams for the same datasets. The vertical axis shows the proportion of correct predictions in each bins. The gray dotted line shows the identity function as an ideal case.