A Second-Order Approach to Learning with Instance-Dependent Label Noise

Zhaowei Zhu[†]   Tongliang Liu[§]   Yang Liu[†]

[†]Computer Science and Engineering, University of California, Santa Cruz

[§]Trustworthy Machine Learning Lab, The University of Sydney

[†]{zwzhu,yangliu}@ucsc.edu   [§]tongliang.liu@sydney.edu.au

Abstract

The presence of label noise often misleads the training of deep neural networks. Departing from the recent literature which largely assumes the label noise rate is only determined by the true label class, the errors in human-annotated labels are more likely to be dependent on the difficulty levels of tasks, resulting in settings with instance-dependent label noise. We first provide evidences that the heterogeneous instance-dependent label noise is effectively down-weighting the examples with higher noise rates in a non-uniform way and thus causes imbalances, rendering the strategy of directly applying methods for class-dependent label noise questionable. Built on a recent work peer loss [24], we then propose and study the potentials of a second-order approach that leverages the estimation of several covariance terms defined between the instance-dependent noise rates and the Bayes optimal label. We show that this set of second-order statistics successfully captures the induced imbalances. We further proceed to show that with the help of the estimated second-order statistics, we identify a new loss function whose expected risk of a classifier under instance-dependent label noise is equivalent to a new problem with only class-dependent label noise. This fact allows us to apply existing solutions to handle this better-studied setting. We provide an efficient procedure to estimate these second-order statistics without accessing either ground truth labels or prior knowledge of the noise rates. Experiments on CIFAR10 and CIFAR100 with synthetic instance-dependent label noise and Clothing1M with real-world human label noise verify our approach. Our implementation is available at https://github.com/UCSC-REAL/CAL.

1. Introduction

Deep neural networks (DNNs) are powerful in revealing and fitting the relationship between feature X and label Y when a sufficiently large dataset is given. However, the label Y usually requires costly human efforts for ac-

curate annotations. With limited budgets/efforts, the resulting dataset would be noisy, and the existence of label noise may mislead DNNs to learn or memorize wrong correlations [10, 11, 35, 38, 47]. To make it worse, the label noise embedded in human annotations is often instance-dependent, e.g., some difficult examples are more prone to be mislabeled [34]. This hidden and imbalanced distribution of noise often has a detrimental effect on the training outcome [15, 23]. It remains an important and challenging task to learn with instance-dependent label noise.

Theory-supported works addressing instance-dependent label noise mostly rely on loss correction, which requires estimating noise rates [40]. Recent work has also considered the possibility of removing the dependency on estimating noise rates [5]. The proposed solution uses a properly specified regularizer to eliminate the effect of instance-dependent label noise. The common theme of the above methods is the focus on learning the underlying clean distribution by using certain forms of <u>first-order statistics</u> of model predictions. In this paper, we propose a second-order approach with the assistance of additional <u>second-order statistics</u> and explore how this information can improve the robustness of learning with instance-dependent label noise. Our main contributions summarize as follows.

1. Departing from recent works [5, 24, 27, 29, 32, 40, 42] which primarily rely on the first-order statistics (i.e. expectation of the models' predictions) to improve the robustness of loss functions, we propose a novel second-order approach and emphasize the importance of using second-order statistics (i.e. several covariance terms) when dealing with instance-dependent label noise.

2. With the perfect knowledge of the covariance terms defined above, we identify a new loss function that transforms the expected risk of a classifier under instance-dependent label noise to a risk with only class-dependent label noise, which is an easier case and can be handled well by existing solutions. Based on peer loss [24], we further show the expected risk of class-dependent noise is equivalent to an affine transformation of the expected

risk under the Bayes optimal distribution. Therefore we establish that our new loss function for *Covariance-Assisted Learning (CAL)* will induce the same minimizer as if we can access the clean Bayes optimal labels.

3. We show how the second-order statistics can be estimated efficiently using existing sample selection techniques. For a more realistic case where the covariance terms cannot be perfectly estimated, we prove the worst-case performance guarantee of our solution.

4. In addition to the theoretical guarantees, the performance of the proposed second-order approach is tested on the CIFAR10 and CIFAR100 datasets with synthetic instance-dependent label noise and the Clothing1M dataset with real-world human label noise.

1.1. Related Works

Below we review the most relevant literature.

Bounded loss functions   Label noise encodes a different relation between features and labels. A line of literature treats the noisy labels as outliers. However, the convex loss functions are shown to be prone to mistakes when outliers exist [25]. To handle this setting, the cross-entropy (CE) loss can be generalized by introducing temperatures to logarithm functions and exponential functions [1, 2, 50]. Noting the CE loss grows explosively when the prediction $f(x)$ approaches zero, some solutions focus on designing bounded loss functions [8, 9, 31, 36]. These methods focus on the numerical property of loss functions, and most of them do not discuss the type of label noise under treatment.

Learning clean distributions   To be noise-tolerant [26], it is necessary to understand the effect of label noise statistically. With the class-dependent assumption, the loss can be corrected/reweighted when the noise transition T is available, which can be estimated by discovering anchor points [22, 29, 39], exploiting clusterability [51], regularizing total variation [49], or minimizing volume of T [20]. The loss correction/reweighting methods rely closely on the quality of the estimated noise transition matrix. To make it more robust, an additive slack variable ΔT [41] or a multiplicative dual T [45] can be used for revision. Directly extending these loss correction methods to instance-dependent label noise is prohibitive since the transition matrix will become a function of feature X and the number of parameters to be estimated is proportional to the number of training instances. Recent follow-up works often introduce extra assumption [40] or measure [4]. Statistically, the loss correction approach is learning the underlying clean distribution if a perfect T is applied. When the class-dependent noise rate is known, surrogate loss [27], an unbiased loss function targeting on binary classifications, also learns the clean distribution. Additionally, the symmetric cross-entropy loss [36], an information-based loss L_{DMI} [43], a correlated agreement (CA) based loss peer loss [24], and ==its adaptation for== ==encouraging confident predictions== [5] are proposed to learn the underlying clean distribution without knowing the noise transition matrix.

Other popular methods   Other methods exist with more sophisticated training framework or pipeline, including sample selection [5, 12, 16, 18, 37, 46, 44], label correction [13, 21, 33], and semi-supervised learning [19, 28], etc.

2. Preliminaries

This paper targets on a classification problem given a set of N training examples with Instance-Dependent label Noise (IDN) denoted by $\widetilde{D} := \{(x_n, \tilde{y}_n)\}_{n \in [N]}$, where $[N] := \{1, 2, \cdots, N\}$ is the set of indices. The corresponding noisy data distribution is denoted by $\widetilde{\mathcal{D}}$. Examples $(x_n, \tilde{y}_n)$ are drawn according to random variables $(X, \widetilde{Y}) \sim \widetilde{\mathcal{D}}$. Our goal is to design a learning mechanism that is guaranteed to be robust when learning with only accessing $\widetilde{D}$. Before proceeding, we summarize important definitions as follows.

Clean distribution $\mathcal{D}$ Each noisy example $(x_n, \tilde{y}_n) \in \widetilde{D}$ corresponds to a clean example $(x_n, y_n) \in D$, which contains one *unobservable* ground-truth label, a.k.a. clean label. Denote by $\mathcal{D}$ the clean distribution. Clean examples (x_n, y_n) are drawn from random variables $(X, Y) \sim \mathcal{D}$.

Bayes optimal distribution $\mathcal{D}^*$ Denote by Y^* the Bayes optimal label given feature X, that is: $Y^*|X := \arg\max_Y \mathbb{P}(Y|X), (X, Y) \sim \mathcal{D}$. The distribution of (X, Y^*) is denoted by $\mathcal{D}^*$. Note the Bayes optimal distribution $\mathcal{D}^*$ is different from the clean distribution $\mathcal{D}$ when $\mathbb{P}(Y|X) \notin \{0, 1\}$. Due to the fact that the information encoded between features and labels is corrupted by label noise, and both clean labels and Bayes optimal labels are unobservable, inferring the Bayes optimal distribution $\mathcal{D}^*$ from the noisy dataset $\widetilde{D}$ is a non-trivial task. Notably there exist two approaches [5, 6] that provide guarantees on constructing the Bayes optimal dataset. We would like to remind the readers that the noisy label $\tilde{y}_n$, clean label y_n, and Bayes optimal label y_n^* for the same feature x_n may disagree with each other.

Most of our developed approaches will focus on dealing with the Bayes optimal distribution $\mathcal{D}^*$. By referring to $\mathcal{D}^*$, as we shall see later, we are allowed to estimate the second-order statistics defined w.r.t. Y^*.

Noise transition matrix $T(X)$ Traditionally, the noise transition matrix is defined based on the relationship between clean distributions and noisy distributions [5, 24, 30, 40]. In recent literature [6], the Bayes optimal label (a.k.a. distilled label in [6]) also plays a significant role. In the image classification tasks where the performance is measured by the clean test accuracy, predicting the Bayes optimal label achieves the best performance. This fact motivates us to define a new noise transition matrix based on the Bayes

optimal label as follows:

$$T_{i,j}(X) = \mathbb{P}(\widetilde{Y} = j | Y^* = i, X),$$

where $T_{i,j}(X)$ denotes the (i,j)-th element of the matrix $T(X)$. Its expectation is defined as $T := \mathbb{E}[T(X)]$, with the (i,j)-th element being $T_{i,j} := \mathbb{E}[T_{i,j}(X)]$.

Other notations Let $\mathcal{X}$ and $\mathcal{Y}$[1] be the space of feature X and label Y, respectively. The classification task aims to identify a classifier $f : \mathcal{X} \rightarrow \mathcal{Y}$ that maps X to Y accurately. One common approach is minimizing the empirical risk using DNNs with respect to the *cross-entropy (CE) loss* defined as: $\ell(f(X), Y) := -\ln(f_X[Y])$, $Y \in [K]$, where $f_X[Y]$ denotes the Y-th component of $f(X)$ and K is the number of classes. Let $\mathbb{1}\{\cdot\}$ be the indicator function taking value 1 when the specified condition is satisfied and 0 otherwise. Define the *0-1 loss* as $\mathbb{1}(f(X), Y) := \mathbb{1}\{f(X) \neq Y\}$. Define the Bayes optimal classifier f^* as $f^* = \arg\min_f \mathbb{E}_{\mathcal{D}^*}[\mathbb{1}(f(X), Y^*)]$. Noting the CE loss is classification-calibrated [3], given enough clean data, the Bayes optimal classifier can be learned using the CE loss: $f^* = \arg\min_f \mathbb{E}_{\mathcal{D}}[\ell(f(X), Y)]$.

Goal Different from the goals in surrogate loss [27], L_{DMI} [43], peer loss [24], and CORES^2 [5], which focus on recovering the performance of learning on clean distributions, we aim to learn a classifier f from the noisy distribution $\widetilde{\mathcal{D}}$ which also minimizes $\mathbb{E}[\mathbb{1}(f(X), Y^*)], (X, Y^*) \sim \mathcal{D}^*$. Note $\mathbb{E}[\mathbb{1}(f^*(X), Y^*)] = 0$ holds for the Bayes optimal classifier f^*. Thus, in the sense of searching for the Bayes optimal classifier, our goals are aligned with the ones focusing on the clean distribution.

3. Insufficiency of First-Order Statistics

Peer loss [24] and its inspired confidence regularizer [5] are two recently introduced robust losses that operate without the knowledge of noise transition matrices, which presents them as preferred solutions for more complex noise settings. In this section, we will first review the usages of first-order statistics in peer loss and the confidence regularizer (Section 3.1), and then analyze the insufficiency of using only the first-order statistics when handling the challenging IDN (Section 3.2). Besides, we will anatomize the down-weighting effect of IDN and provide intuitions for how to make IDN easier to handle (Section 3.3).

We formalize our arguments using peer loss, primarily due to 1) its clean analytical form, and 2) that our later proposed solution will be built on peer loss too. Despite the focus on peer loss, we believe these observations are generally true when other existing training approaches meet IDN.

For ease of presentation, the following analyses focus on binary cases (with classes $\{-1, +1\}$). Note the class -1

---

should be mapped to class 0 following the notations in Section 2. For a clear comparison with previous works, we follow the notation in [24] and use class $\{-1, +1\}$ to represent classes $\{0, 1\}$ when $K = 2$. The error rates in $\widetilde{Y}$ are then denoted as $e_+(X) := \mathbb{P}(\widetilde{Y} = -1 | Y^* = +1, X)$, $e_-(X) := \mathbb{P}(\widetilde{Y} = +1 | Y^* = -1, X)$. Most of the discussions generalize to the multi-class setting.

3.1. Using First-Order Statistics in Peer Loss

It has been proposed and proved in peer loss [24] and CORES^2 [5] that the learning could be robust to label noise by considering some first-order statistics related to the model predictions. For each example $(x_n, \tilde{y}_n)$, peer loss [24] has the following form:

$$\ell_{\text{PL}}(f(x_n), \tilde{y}_n) := \ell(f(x_n), \tilde{y}_n) - \ell(f(x_{n_1}), \tilde{y}_{n_2}),$$

where $(x_{n_1}, \tilde{y}_{n_1})$ and $(x_{n_2}, \tilde{y}_{n_2})$ are two randomly sampled peer samples for n. The first-order statistics related to model predictions characterized by the peer term $\ell(f(x_{n_1}), \tilde{y}_{n_2})$ are further extended to a confidence regularizer in CORES^2 [5]:

$$\ell_{\text{CORES}^2}(f(x_n), \tilde{y}_n) := \ell(f(x_n), \tilde{y}_n) - \beta \mathbb{E}_{\mathcal{D}_{\widetilde{Y}|\widetilde{D}}}[\ell(f(x_n), \widetilde{Y})],$$

where β is a hyperparameter controlling the ability of regularizer, and $\mathcal{D}_{\widetilde{Y}|\widetilde{D}}$ is the marginal distribution of $\widetilde{Y}$ given dataset $\widetilde{D}$. Although it has been shown in [5] that learning with an appropriate β would be robust to instance-dependent label noise theoretically, in real experiments, converging to the guaranteed optimum by solving a highly non-convex problem is difficult.

3.2. Peer Loss with IDN

Now we analyze the possible performance degradation of using the binary peer loss function proposed in [24] to handle IDN. Denote by

$$\tilde{f}_{\text{peer}}^* := \arg\min_f \mathbb{E}_{\widetilde{\mathcal{D}}}\left[\mathbb{1}_{\text{PL}}(f(X), \widetilde{Y})\right]$$

the optimal classifier learned by minimizing 0-1 peer loss, where $\mathbb{1}_{\text{PL}}$ represents ℓ_{PL} with 0-1 loss (could also be generalized for ℓ_{CORES^2} with 0-1 loss). Let $p^* := \mathbb{P}(Y^* = +1)$. With a bounded variance in the error rates, supposing

$$\mathbb{E}|e_+(X) - \mathbb{E}[e_+(X)]| \leq \epsilon_+, \ \mathbb{E}|e_-(X) - \mathbb{E}[e_-(X)]| \leq \epsilon_-,$$

the worst-case performance bound for using pure peer loss is provided in Theorem 1 and proved in Appendix B.1.

Theorem 1 (Performance of peer loss). *With the peer loss function proposed in [24], we have*

$$\mathbb{E}[\mathbb{1}(\tilde{f}_{peer}^*(X), Y^*)] \leq \frac{2(\epsilon_+ + \epsilon_-)}{1 - e_+ - e_-} + 2|p^* - 0.5|.$$

Theorem 1 shows the ratio of wrong predictions given by $\tilde{f}^*_{\text{peer}}$ includes two components. The former term $\frac{2(\epsilon_+ + \epsilon_-)}{1-e_+-e_-}$ is directly caused by IDN, indicating the error is increasing when the instance-dependent noise rates have larger mean (larger $e_+ + e_-$) and larger variation (larger $\epsilon_+ + \epsilon_-$). The latter term $2|p^* - 0.5|$ shows possible errors induced by an unbalanced $\mathcal{D}^*$. Theorem 1 generalizes peer loss where $\epsilon_+ = \epsilon_- = 0$, i.e., the error rates are homogeneous across data instances, and there is no need to consider any second-order statistics that involve the distribution of noise rates.

3.3. Down-weighting Effect of IDN

We further discuss motivations and intuitions by studying how IDN affects the training differently from the class-dependent one. Intuitively, a high noise rate reduces the informativeness of a particular example (x, y), therefore "down-weighting" its contribution to training. We now analytically show this under peer loss.

As a building block, the invariant property (in terms of the *clean distribution* $\mathcal{D}$) originally discovered by peer loss on class-dependent label noise is first adapted for the *Bayes optimal distribution* $\mathcal{D}^*$. Define $e_- := \mathbb{P}(\widetilde{Y} = +1|Y^* = -1)$ and $e_+ := \mathbb{P}(\widetilde{Y} = -1|Y^* = +1)$. Focusing on a particular class-dependent $\widetilde{\mathcal{D}}$, we provide Lemma 1 and its proof in Appendix A.1.

Lemma 1 (Invariant property of peer loss [24]). *Peer loss is invariant to class-dependent label noise:*

$$\mathbb{E}_{\widetilde{\mathcal{D}}}[\mathbb{1}_{PL}(f(X), \widetilde{Y})] = (1-e_+-e_-)\mathbb{E}_{\mathcal{D}^*}[\mathbb{1}_{PL}(f(X), Y^*)]. \quad (1)$$

Then we discuss the effect of IDN. Without loss of generality, consider a case where noisy examples are drawn from two noisy distributions $\widetilde{\mathcal{D}}_{\text{I}}$ and $\widetilde{\mathcal{D}}_{\text{II}}$, and the noise rate of $\widetilde{\mathcal{D}}_{\text{II}}$ is higher than $\widetilde{\mathcal{D}}_{\text{I}}$, i.e. $e_{+,\text{II}} + e_{-,\text{II}} > e_{+,\text{I}} + e_{-,\text{I}}$, where $e_{+,\text{I (II)}} := \mathbb{P}_{\widetilde{\mathcal{D}}_{\text{I (II)}}}(\widetilde{Y} = -1|Y^* = +1)$. Assume a particular setting of IDN that the noise is class-dependent (but not instance-dependent) only within each distribution, and different between two distributions, i.e. *part-dependent* [40]. Let $\mathcal{D}^*_{\text{I}}$ and $\mathcal{D}^*_{\text{II}}$ be the Bayes optimal distribution related to $\widetilde{\mathcal{D}}_{\text{I}}$ and $\widetilde{\mathcal{D}}_{\text{II}}$. For simplicity, we write $\mathbb{P}((X, Y^*) \sim \mathcal{D}^*_{\text{I(II)}}|(X, Y^*) \sim \mathcal{D}^*)$ as $\mathbb{P}(\mathcal{D}^*_{\text{I(II)}})$. Then $\mathbb{P}(\mathcal{D}^*_{\text{I}}) = \mathbb{P}(\widetilde{\mathcal{D}}_{\text{I}})$ and $\mathbb{P}(\mathcal{D}^*_{\text{II}}) = \mathbb{P}(\widetilde{\mathcal{D}}_{\text{II}})$. Note $\mathbb{P}(\widetilde{\mathcal{D}}_{\text{I}})e_{+,\text{I}} + \mathbb{P}(\widetilde{\mathcal{D}}_{\text{II}})e_{+,\text{II}} = e_+$ and $\mathbb{P}(\widetilde{\mathcal{D}}_{\text{I}})e_{-,\text{I}} + \mathbb{P}(\widetilde{\mathcal{D}}_{\text{II}})e_{-,\text{II}} = e_-$. Then we have the following equality:

$$\mathbb{E}_{\widetilde{\mathcal{D}}}[\mathbb{1}_{\text{PL}}(f(X), \widetilde{Y})]$$
$$= \mathbb{P}(\widetilde{\mathcal{D}}_{\text{I}})(1-e_{+,\text{I}}-e_{-,\text{I}})\mathbb{E}_{\mathcal{D}^*_{\text{I}}}[\mathbb{1}_{\text{PL}}(f(X), Y^*)]$$
$$+ \mathbb{P}(\widetilde{\mathcal{D}}_{\text{II}})(1-e_{+,\text{II}}-e_{-,\text{II}})\mathbb{E}_{\mathcal{D}^*_{\text{II}}}[\mathbb{1}_{\text{PL}}(f(X), Y^*)]$$
$$= (1-e_{+,\text{I}}-e_{-,\text{I}})\bigg(\mathbb{P}(\mathcal{D}^*_{\text{I}})\mathbb{E}_{\mathcal{D}^*_{\text{I}}}[\mathbb{1}_{\text{PL}}(f(X), Y^*)]$$
$$+ \frac{1-e_{+,\text{II}}-e_{-,\text{II}}}{1-e_{+,\text{I}}-e_{-,\text{I}}}\mathbb{P}(\mathcal{D}^*_{\text{II}})\mathbb{E}_{\mathcal{D}^*_{\text{II}}}[\mathbb{1}_{\text{PL}}(f(X), Y^*)]\bigg),$$

where

$$\frac{1-e_{+,\text{II}}-e_{-,\text{II}}}{1-e_{+,\text{I}}-e_{-,\text{I}}} < 1$$

indicates *down-weighting* examples drawn from $\widetilde{\mathcal{D}}_{\text{II}}$ (compared to the class-dependent label noise).

What can we learn from this observation? First, we show the peer loss is already down weighting the importance of the more noisy examples. However, simply dropping examples with potentially high-level noise might lead the classifier to learn a biased distribution. Moreover, subjectively confusing examples are more prone to be mislabeled and critical for accurate predictions [34], thus need to be carefully addressed. Our second observation is that if we find a way to compensate for the "imbalances" caused by the down-weighting effects shown above, the challenging instance-dependent label noise could be transformed into a class-dependent one, which existing techniques can then handle. More specifically, the above result shows the down-weighting effect is characterized by $T(X)$, implying only using the first-order statistics of model predictions without considering the distributions of the noise transition matrix $T(X)$ is insufficient to capture the complexity of the learning task. However, accurately estimating $T(X)$ is prohibitive since the number of parameters to be estimated is almost at the order of $O(NK^2)$ – recall N is the number of training examples and K is the number of classes. Even though we can roughly estimate $T(X)$, applying element-wise correction relying on the estimated $T(X)$ may accumulate errors. Therefore, to achieve the transformation from the instance-dependent to the easier class-dependent, we need to resort to other statistical properties of $T(X)$.

4. Covariance-Assisted Learning (CAL)

From the analyses in Section 3.3, we know the instance-dependent label noise will "automatically" assign different weights to examples with different noise rates, thus cause imbalances. When the optimal solution does not change under such down-weighting effects, the first-order statistics based on peer loss [5, 24] work well. However, for a more robust and general solution, using additional information to "balance" the effective weights of different examples is necessary. Although the Bayes optimal distribution is not accessible in real experiments, we first assume its existence for theoretical analyses in the ideal case, then we will discuss the gap to this optimal solution when we can only use a proxy $\hat{\mathcal{D}}$ that can be constructed efficiently.

4.1. Extracting Covariance from IDN

Again consider an instance-dependent noisy distribution $\widetilde{\mathcal{D}}$ with binary classes where $\widetilde{Y} \in \{-1, +1\}$. Define the following two random variables to facilitate analyses:

$$Z_1(X) := 1-e_+(X)-e_-(X), \quad Z_2(X) = e_+(X)-e_-(X).$$

Recall $e_+ := \mathbb{E}[e_+(X)]$ and $e_- := \mathbb{E}[e_-(X)]$. Let $\text{Cov}_{\mathcal{D}}(A, B) := \mathbb{E}[(A - \mathbb{E}[A])(B - \mathbb{E}[B])]$ be the covariance between random variables A and B w.r.t. the distribution $\mathcal{D}$. The exact effects of IDN on peer loss functions are revealed in Theorem 2 and proved in Appendix B.2.

Theorem 2 (Decoupling binary IDN). *In binary classifications, the expected peer loss with IDN writes as:*

$$\mathbb{E}_{\widetilde{\mathcal{D}}}[\mathbb{1}_{PL}(f(X), \widetilde{Y})] = (1 - e_+ - e_-)\mathbb{E}_{\mathcal{D}^*}[\mathbb{1}_{PL}(f(X), Y^*)]$$
$$+ Cov_{\mathcal{D}^*}(Z_1(X), \mathbb{1}(f(X), Y^*))$$
$$+ Cov_{\mathcal{D}^*}(Z_2(X), \mathbb{1}(f(X), -1)). \quad (2)$$

Theorem 2 effectively divides the instance-dependent label noise into two parts. As shown in Eq. (2), the first line is the same as Eq. (1) in Lemma 1, indicating the average effect of instance-dependent label noise can be treated as a class-dependent one with parameters e_+, e_-. The additional two covariance terms in the second and the third lines of Eq. (2) characterize the additional contribution of examples due to their differences in the label noise rates. The covariance terms will become larger for a setting with more diverse noise rates, capturing a more heterogeneous and uncertain learning environment. Interested readers are also referred to the high-level intuitions for using covariance terms at the end of Section 3.3.

We now briefly discuss one extension of Theorem 2 to a K-class classification task. Following the assumption adopted in [24], we consider a particular setting of IDN whose the expected transition matrix satisfies $T_{i,j} = T_{k,j}, \forall i \neq j \neq k$. Denote by $e_j = T_{i,j}, \forall i \neq j$. Corollary 1 decouples the effects of IDN in multi-class cases and is proved in Appendix C.1.

Corollary 1 (Decoupling multi-class IDN). *In multi-class classifications, when the expected transition matrix satisfies $e_j = T_{i,j} = T_{k,j}, \forall i \neq j \neq k$, the expected peer loss with IDN writes as:*

$$\mathbb{E}_{\widetilde{\mathcal{D}}}[\ell_{PL}(f(X), \widetilde{Y})] = (1 - \sum_{i \in [K]} e_i)\mathbb{E}_{\mathcal{D}^*}[\ell_{PL}(f(X), Y^*)]$$
$$+ \sum_{j \in [K]} \mathbb{E}_{\mathcal{D}_{Y^*}} [Cov_{\mathcal{D}^*|Y^*}(T_{Y^*,j}(X), \ell(f(X), j))],$$

where $\mathcal{D}_{Y^*}$ is the marginal distribution of Y^* and $\mathcal{D}^*|Y^*$ is the conditional distribution of $\mathcal{D}^*$ given Y^*.

4.2. Using Second-Order Statistics

Inspired by Theorem 2, if $\mathcal{D}^*$ is available, we can subtract two covariance terms and make peer loss invariant to IDN. Specifically, define

$$\tilde{f}^*_{\text{CAL}} = \arg \min_f \mathbb{E}_{\widetilde{\mathcal{D}}}[\mathbb{1}_{\text{PL}}(f(X), \widetilde{Y})] - \text{Cov}(Z_1(X), \mathbb{1}(f(X), Y^*))$$
$$- \text{Cov}(Z_2(X), \mathbb{1}(f(X), -1)).$$

We have the following optimality guarantee and its proof is deferred to Appendix B.3.

Theorem 3. $\tilde{f}^*_{\text{CAL}} \in \arg \min_f \mathbb{E}_{\mathcal{D}^*}[\mathbb{1}(f(X), Y^*)].$

For a K-class classification problem, a general loss function for our *Covariance-Assisted Learning (CAL)* approach is given by

$$\ell_{\text{CAL}}(f(x_n), \tilde{y}_n) = \ell_{\text{PL}}(f(x_n), \tilde{y}_n)$$
$$- \sum_{j \in [K]} \mathbb{E}_{\mathcal{D}_{Y^*}} [\text{Cov}_{\mathcal{D}^*|Y^*}(T_{Y^*,j}(X), \ell(f(X), j))]. \quad (3)$$

Eq. (3) shows the Bayes optimal distribution $\mathcal{D}^*$ is critical in implementing the proposed covariance terms. However, $\mathcal{D}^*$ cannot be obtained trivially, and only imperfect proxy constructions of the dataset (denoted by $\hat{D}$) could be expected. Detailed constructions of $\hat{D}$ are deferred to Section 4.2.1.

Advantages of using covariance terms There are several advantages of using the proposed covariance terms. Unlike directly correcting labels according to $\mathcal{D}^*$, the proposed covariance term can be viewed as a "soft" correction that maintains the information encoded in both original noisy labels and the estimated Bayes optimal labels. Keeping both information is beneficial as suggested in [13]. Moreover, compared to the direct loss correction approaches [30, 40, 41], we keep the original learning objective and apply "correction" using an additional term. Our method is more robust in practice compared to these direct end-to-end loss correction approaches due to two reasons: 1) The covariance term summarizes the impact of the complex noise using an average term, indicating that our approach is less sensitive to the estimation precision of an individual example; 2) As will be shown in Section 4.3, the proposed method is tolerant with accessing an imperfect $\mathcal{D}^*$.

Estimating the covariance terms relies on samples drawn from distribution $\mathcal{D}^*$. Thus, we need to construct a dataset $\hat{D}$, which is similar or unbiased w.r.t. D^*. We will first show the algorithm for constructing $\hat{D}$, then provide details for DNN implementations.

4.2.1 Constructing $\hat{D}$

To achieve unbiased estimates of the variance terms, the high-level intuition for constructing $\hat{D}$ is determining whether the label of each example in $\widetilde{D}$ is Bayes optimal or not by comparing the likelihood, confidence, or loss of classifying the (noisy) label to some thresholds. There are several methods for constructing $\hat{D}$: distillation [6], searching to exploit [44], and sample sieve [5]. If the model does not overfit the label noise and learns the noisy distribution, both methods in [6] and [44] work well. However, for the challenging instance-dependent label noise, overfitting occurs easily thus techniques to avoid overfitting are necessary. In this paper, we primarily adapt the sample sieve proposed in [5], which uses a confidence regularizer to avoid

Algorithm 1: Constructing $\hat{D}$

Input: Noisy dataset $\widetilde{D}$. Thresholds $L_{\min} \leq L_{\max}$.
Number of epochs T. $\hat{D} = \widetilde{D}$.

1 Train the sample sieve in [5] for T epochs and get the model f;
2 **for** $n \in [N]$ **do**
3 $\quad$ Calculate $\alpha_{n,T}$ following [5];
4 $\quad$ **if** $\ell_{CORES^2}(f(x_n), \tilde{y}_n) - \alpha_{n,T} \leq L_{\min}$ **then**
5 $\quad\quad$ $\hat{y}_n = \tilde{y}_n$;
6 $\quad$ **else if** $\ell_{CORES^2}(f(x_n), \tilde{y}_n) - \alpha_{n,T} > L_{\max}$ **then**
7 $\quad\quad$ $\hat{y}_n = \arg\max_{y \in [K]} f_{x_n}[y]$;
8 $\quad$ **else**
9 $\quad\quad$ $\hat{y}_n = -1$ (drop example n);
10 **end**

Output: $\hat{D} := \{(x_n, \hat{y}_n) : n \in [N], \hat{y}_n \neq -1\}$

overfitting, to construct $\hat{D}$. Specifically, as shown in [5], in each epoch t, the regularized loss for each example is adjusted by the parameter $\alpha_{n,t}$, which can be calculated based on model predictions in linear time. In the ideal cases assumed in [5], any example with a positive adjusted loss is corrupted (with a wrong label).

We summarized the corresponding procedures in Algorithm 1, where the critical thresholds for comparing losses are denoted by $L_{\min}$ and $L_{\max}$. At Line 5, if the loss adjusted by $\alpha_{n,t}$ is small enough (smaller than the threshold $L_{\min}$), we assume $\tilde{y}_n$ is the Bayes optimal label. Accordingly, at Line 7, if the adjusted loss is too large (larger than the threshold $L_{\max}$), we treat $\tilde{y}_n$ as a corrupted one and assume the class with maximum predicted probability to be Bayes optimal one. For the examples with moderate adjusted loss, we drop it as indicated in Line 9. In ideal cases with infinite model capacity and sufficiently many examples (as assumed in [5]), we can set thresholds $L_{\min} = L_{\max} = 0$ to guarantee a separation of clean and corrupted examples, thus $\hat{D}$ will be an unbiased proxy to $\mathcal{D}^*$ [2]. However, in real experiments, when both the model capacity and the number of examples are limited, we may need to tune $L_{\min}$ and $L_{\max}$ to obtain a high-quality construction of $\hat{D}$. In this paper, we set $L_{\min} = L_{\max}$ to ensure $|\hat{D}| = |D^*|$ and reduce the effort to tuning both thresholds simultaneously.

Note that using $\hat{D}$ to estimate the covariance terms could be made theoretically more rigorous by applying appropriate re-weighting techniques [6, 7, 14]. See Appendix D.1 for more discussions and corresponding guarantees. We omit the details here due to the space limit. Nonetheless, our approach is tolerant of an imperfect $\hat{D}$, which will be shown theoretically in Section 4.3.

4.2.2 Implementations

For implementations with deep neural network solutions, we need to estimate the transition matrix $T(X)$ relying on $\hat{D}$ and estimate the covariance terms along with stochastic gradient descent (SGD) updates.

Covariance Estimation in SGD As required in (3), with a particular $\hat{D}$, each computation for $\hat{T}_{i,j}(x_n)$ requires only one time check of the associated noisy label as follows:

$$\hat{T}_{i,j}(x_n) = \mathbb{1}\{\hat{y}_n = i, \tilde{y}_n = j\}. \tag{4}$$

When $\hat{D}$ is unbiased w.r.t. D^*, the estimation in (4) is also unbiased because

$$\mathbb{E}_{\widetilde{D}|X,\hat{Y}=i}[\hat{T}_{i,j}(X)] = \mathbb{E}_{\widetilde{D}|X,\hat{Y}=i}[\mathbb{1}\{\hat{Y}=i,\widetilde{Y}=j|X\}]$$
$$= \mathbb{P}(\widetilde{Y}=j|X,\hat{Y}=i) = \mathbb{P}(\widetilde{Y}=j|X,Y^*=i).$$

Noting $\mathrm{Cov}_{\mathcal{D}}(A, B) := \mathbb{E}[(A - \mathbb{E}[A])(B - \mathbb{E}[B])] = \mathbb{E}[(A - \mathbb{E}[A]) \cdot B]$, the covariance can be estimated empirically as

$$\frac{1}{N} \sum_{n \in [N]} \sum_{i,j \in [K]} \mathbb{1}\{y_n^* = i\} \left[(\hat{T}_{i,j}(x_n) - \hat{T}_{i,j}) \cdot \ell(f(x_n), j) \right].$$

For each batch of data, the above estimation has $O(N)$ complexities in computation and space. To reduce both complexities, with the cost of the estimation quality, we use $|E_b|$ examples to estimate the covariance in each batch, where E_b is the set of sample indices of batch-b. Per sample wise, Eq. (3) can be transformed to

$$\ell_{\mathrm{CAL}}(f(x_n), \tilde{y}_n) = \ell_{\mathrm{PL}}(f(x_n), \tilde{y}_n)$$
$$- \sum_{i,j \in [K]} \mathbb{1}\{y_n^* = i\} \left[(\hat{T}_{i,j}(x_n) - \hat{T}_{i,j}) \cdot \ell(f(x_n), j) \right].$$

With the above implementation, the estimation is done locally for each point in $O(1)$ complexity.

4.3. CAL with Imperfect Covariance Estimates

As mentioned earlier, $\mathcal{D}^*$ cannot be perfectly obtained in practice. Thus, there is a performance gap between the ideal case (with perfect knowledge of $\mathcal{D}^*$) and the actually achieved one. We now analyze the effect of imperfect covariance terms (Theorem 4).

Denote the imperfect covariance estimates by $\hat{D}^\tau$, where $\tau \in [0, 1]$ is the expected ratio (a.k.a. probability) of correct examples in $\hat{D}^\tau$: $\tau = \mathbb{E}[\mathbb{1}\{(X, \hat{Y}) \in \hat{D}^\tau | (X, Y^*) \in D^*\}] = \mathbb{P}((X, \hat{Y}) \sim \hat{D}^\tau | (X, Y^*) \sim \mathcal{D}^*)$. With $\hat{D}^\tau$, the minimizer of the 0-1 CAL loss is given by:

$$\tilde{f}^*_{\mathrm{CAL}\text{-}\tau} = \arg\min_f \mathbb{E}_{\widetilde{D}} \Big[\mathbb{1}_{\mathrm{PL}}(f(X), \widetilde{Y}) - \mathrm{Cov}_{\hat{D}^\tau}(Z_1(X), \mathbb{1}(f(X), \hat{Y}))$$
$$- \mathrm{Cov}_{\hat{D}^\tau}(Z_2(X), \mathbb{1}(f(X), -1)) \Big].$$

Theorem 4 reports the error bound produced by $\tilde{f}^*_{\text{CAL-}\tau}$. See Appendix B.4 for the proof.

Theorem 4 (Imperfect Covariance). *With $\hat{D}^\tau$, when $p^* = 0.5$, we have*

$$\mathbb{E}[\mathbb{1}(\tilde{f}^*_{\text{CAL-}\tau}(X), Y^*)] \leq \frac{4(1-\tau)(\epsilon_+ + \epsilon_-)}{1 - e_+ - e_-}.$$

Theorem 4 shows the quality of $\hat{D}^\tau$ controls the scale of the worst-case error upper-bound. Compared with Theorem 1 where no covariance term is used, we know the covariance terms will always be helpful when $\tau \in [0.5, 1]$. That is, the training with the assistance of covariance terms will achieve better (worst-case) accuracy on the Bayes optimal distribution when the construction $\hat{D}^\tau$ is better than a dataset that includes each instance in D^* randomly with 50% chance.

5. Experiments

We now present our experiment setups and results.

5.1. General Experiment Settings

Datasets and models   The advantage of introducing our second-order approach is evaluated on three benchmark datasets: CIFAR10, CIFAR100 [17] and Clothing1M [42]. Following the convention from [5, 43], we use ResNet34 for CIFAR10 and CIFAR100 and ResNet50 for Clothing1M. Noting the expected peer term $\mathbb{E}_{\mathcal{D}_{\widetilde{Y}|\widetilde{D}}}[\ell(f(x_n), \widetilde{Y})]$ (a.k.a. *confidence regularizer (CR)* as implemented in [5]) is more stable and converges faster than the one with peer samples, we train with ℓ_{CORES^2}. It also enables a fair ablation study since $\hat{D}$ is constructed relying on [5]. For numerical stability, we use a cut-off version of the cross-entropy loss $\ell(f(x), y) = -\ln(f_x[y] + \varepsilon)$. Specifically, we use $\varepsilon = 10^{-8}$ for the traditional cross-entropy term, use $\varepsilon = 10^{-5}$ for the CR term, and the covariance term. All the experiments use a momentum of 0.9. The weight decay is set as 0.0005 for CIFAR experiments and 0.001 for Clothing1M.

Noise type   For CIFAR datasets, the instance-dependent label noise is generated following the method from [5, 40]. The basic idea is randomly generating one vector for each class (K vectors in total) and project each incoming feature onto these K vectors. The label noise is added by jointly considering the clean label and the projection results. See Appendix D.2 for details. In expectation, the noise rate η is the overall ratio of examples with a wrong label in the entire dataset. For the Clothing1M dataset, we train on 1 million noisy training examples that encode the real-world human noise.

5.2. Baselines

We compare our method with several related works, where the cross-entropy loss is tested as a common baseline. Additionally, the generalized cross-entropy [50] is compared as a generalization of mean absolute error and cross-entropy designed for label noise. Popular loss correction based methods [29, 40, 41], sample selection based methods [5, 12, 37, 46], and noise-robust loss functions [24, 43] are also chosen for comparisons. All the compared methods adopt similar data augmentations, including standard random crop, random flip, and normalization. Note the recent work on part-dependent label noise [40] did not apply random crop and flip on the CIFAR dataset. For a fair comparison with [40], we remove the corresponding data augmentations from our approach and defer the comparison to Appendix D.3. The semi-supervised learning based methods with extra feature-extraction and data augmentations are not included. All the CIFAR experiments are repeated 5 times with independently synthesized IDN. The highest accuracies on the clean testing dataset are averaged over 5 trials to show the best generalization ability of each method.

5.3. Performance Comparisons

5.3.1   CIFAR

In experiments on CIFAR datasets, we use a batch size of 128, an initial learning rate of 0.1, and reduce it by a factor of 10 at epoch 60.

Construct $\hat{D}$ To construct $\hat{D}$, we update the DNN for 65 epochs by minimizing ℓ_{CORES^2} (without dynamic sample sieve) and apply Algorithm 1 with $L_{\min} = L_{\max} = -8$.[3] For a numerically stable solution, we use the square root of the noise prior for the CR term in ℓ_{CORES^2} as $-\beta \sum_{i \in [K]} \frac{\sqrt{\mathbb{P}(\widetilde{Y}=i|\widetilde{D})}}{\sum_{j=1}^{K} \sqrt{\mathbb{P}(\widetilde{Y}=j|\widetilde{D})}} \ell(f(x_n), i)$. The hyperparameter β is set to 2 for CIFAR10 and 10 for CIFAR100.

Train with CAL   With an estimate of D^*, we re-train the model 100 epochs. The hyper-parameter β is set to 1 for CIFAR10 and 10 for CIFAR100. Note the hyperparameters $(L_{\min}, L_{\max}, \beta)$ can be better set if a clean validation set is available.

Performance   Table 1 compares the means and standard deviations of test accuracies on the clean test dataset when the model is trained with synthesized instance-dependent label noise in different levels. All the compared methods use ResNet34 as the backbone. On CIFAR10, with a low-level label noise ($\eta = 0.2$), all the compared methods perform well and achieve higher average test accuracies than the standard CE loss. When the overall noise

---

[3]Theoretically, we have $L_{\min} = L_{\max} = 0$ if both the CE term and the CR term use a log loss without cut-off ($\varepsilon = 0$). Current setting works well (not the best) for CIFAR experiments empirically.

Table 1. Comparison of test accuracies (%) using different methods.

Method	Inst. CIFAR10			Inst. CIFAR100		
	$\eta = 0.2$	$\eta = 0.4$	$\eta = 0.6$	$\eta = 0.2$	$\eta = 0.4$	$\eta = 0.6$
CE (Standard)	85.45±0.57	76.23±1.54	59.75±1.30	57.79±1.25	41.15±0.83	25.68±1.55
Forward T [29]	87.22±1.60	79.37±2.72	66.56±4.90	58.19±1.37	42.80±1.01	27.91±3.35
L_{DMI} [43]	88.57±0.60	82.82±1.49	69.94±1.31	57.90±1.21	42.70±0.92	26.96±2.08
L_q [50]	85.81±0.83	74.66±1.12	60.76±3.08	57.03±0.27	39.81±1.18	24.87±2.46
Co-teaching [12]	88.87±0.24	73.00±1.24	62.51±1.98	43.30±0.39	23.21±0.57	12.58±0.51
Co-teaching+ [46]	89.80±0.28	73.78±1.39	59.22±6.34	41.71±0.78	24.45±0.71	12.58±0.51
JoCoR [37]	88.78±0.15	71.64±3.09	63.46±1.58	43.66±1.32	23.95±0.44	13.16±0.91
Reweight-R [41]	90.04±0.46	84.11±2.47	72.18±2.47	58.00±0.36	43.83±8.42	36.07±9.73
Peer Loss [24]	89.12±0.76	83.26±0.42	74.53±1.22	61.16±0.64	47.23±1.23	31.71±2.06
CORES^2 [5]	91.14±0.46	83.67±1.29	77.68±2.24	66.47±0.45	58.99±1.49	38.55±3.25
CAL	**92.01±0.75**	**84.96±1.25**	**79.82±2.56**	**69.11±0.46**	**63.17±1.40**	**43.58±3.30**

rates increase to high, most of the methods suffer from severe performance degradation while CAL still achieves the best performance. There are similar observations on CIFAR100. By comparing CAL with CORES^2, we conclude that the adopted second-order statistics do work well and bring non-trivial performance improvement. Besides, on the CIFAR100 dataset with $\eta = 0.4$ and 0.6, we observe Reweight-R [41] has a large standard deviation and a relatively high mean, indicating it may perform as well as or even better than CAL in some trials. It also shows the potential of using a revised transition matrix T [41] in severe and challenging instance-dependent label noise settings.

5.3.2 Clothing1M

For Clothing1M, we first train the model following the settings in [5] and construct $\hat{D}$ with the best model. Noting the overall accuracy of noisy labels in Clothing1M is about 61.54% [42], we set an appropriate $L_{\min} = L_{\max}$ such that 61.54% of training examples satisfying $\ell_{\text{CORES}^2} - \alpha_{n,t} \leq L_{\min}$. With $\hat{D}$, we sample a class-balanced dataset by randomly choosing $18,976$ noisy examples for each class and continue training the model with $\beta = 1$ and an initial learning rate of 10^{-5} for 120 epochs. Other parameters are set following [5]. See Appendix D.4 for more detailed experimental settings. Table 2 shows CAL performs well in the real-world human noise.

5.4. Ablation Study

Table 3 shows either the covariance term or the peer term can work well individually and significantly improve the performance when they work jointly. Comparing the first row with the second row, we find the second-order statistics can work well (except for $\eta = 0.4$) even without the peer (CR) term. In row 4, we show the performance at epoch 65 since the second-order statistics are estimated relying on the model prediction at this epoch. By comparing row 4 with row 5, we know the second-order statistics indeed lead to non-trivial improvement in the performance. Even though the covariance term individually can only achieve an accu-

Table 2. The best epoch (clean) test accuracies on Clothing1M.

Method	Accuracy
CE (standard)	68.94
Forward T [29]	70.83
Co-teaching [12]	69.21
JoCoR [37]	70.30
L_{DMI} [43]	72.46
PTD-R-V[40]	71.67
CORES^2 [5]	73.24
CAL	**74.17**

Table 3. Analysis of each component of CAL on CIFAR10. The result of a particular trial is presented. *Cov.*: the covariance term. *Peer*: the CR term [5] (a.k.a. expected peer term [24]).

row #	Cov.	Peer	Epoch	$\eta = 0.2$	$\eta = 0.4$	$\eta = 0.6$
1	✗	✗	Best	90.47	82.56	64.65
2	✓	✗	Best	92.10	78.49	73.55
3	✗	✓	Best	91.85	84.41	78.74
4	✗	✓	Fixed@65	90.73	82.76	77.70
5	✓	✓	Best	92.69	85.55	81.54

racy of 78.49 when $\eta = 0.4$, it can still contribute more than 1% of the performance improvement (from 84.41% to 85.55%) when it is implemented with the peer term. This observation shows the robustness of CAL.

6. Conclusions

This paper has proposed a second-order approach to transforming the challenging instance-dependent label noise into a class-dependent one such that existing methods targeting the class-dependent label noise could be implemented. Currently, the necessary information for the covariance term is estimated based on a sample selection method. Future directions of this work include extensions to other methods for estimating the covariance terms accurately. We are also interested in exploring the combination of second-order information with other robust learning techniques.

References

[1] Ehsan Amid, Manfred KK Warmuth, Rohan Anil, and Tomer Koren. Robust bi-tempered logistic loss based on bregman divergences. In *Advances in Neural Information Processing Systems*, pages 14987–14996, 2019. 2

[2] Ehsan Amid, Manfred K Warmuth, and Sriram Srinivasan. Two-temperature logistic regression based on the tsallis divergence. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 2388–2396. PMLR, 2019. 2

[3] Peter L Bartlett, Michael I Jordan, and Jon D McAuliffe. Convexity, classification, and risk bounds. *Journal of the American Statistical Association*, 101(473):138–156, 2006. 3

[4] Antonin Berthon, Bo Han, Gang Niu, Tongliang Liu, and Masashi Sugiyama. Confidence scores make instance-dependent label-noise learning possible. *arXiv preprint arXiv:2001.03772*, 2020. 2

[5] Hao Cheng, Zhaowei Zhu, Xingyu Li, Yifei Gong, Xing Sun, and Yang Liu. Learning with instance-dependent label noise: A sample sieve approach. In *International Conference on Learning Representations*, 2021. 1, 2, 3, 4, 5, 6, 7, 8, 18

[6] Jiacheng Cheng, Tongliang Liu, Kotagiri Ramamohanarao, and Dacheng Tao. Learning with bounded instance-and label-dependent label noise. In *Proceedings of the 37th International Conference on Machine Learning*, ICML '20, 2020. 2, 5, 6, 17

[7] Tongtong Fang, Nan Lu, Gang Niu, and Masashi Sugiyama. Rethinking importance weighting for deep learning under distribution shift. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 11996–12007. Curran Associates, Inc., 2020. 6, 17

[8] Aritra Ghosh, Himanshu Kumar, and PS Sastry. Robust loss functions under label noise for deep neural networks. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017. 2

[9] Maoguo Gong, Hao Li, Deyu Meng, Qiguang Miao, and Jia Liu. Decomposition-based evolutionary multiobjective optimization to self-paced learning. *IEEE Transactions on Evolutionary Computation*, 23(2):288–302, 2018. 2

[10] Bo Han, Gang Niu, Xingrui Yu, Quanming Yao, Miao Xu, Ivor Tsang, and Masashi Sugiyama. Sigua: Forgetting may make learning with noisy labels more robust. In *International Conference on Machine Learning*, pages 4006–4016. PMLR, 2020. 1

[11] Bo Han, Quanming Yao, Tongliang Liu, Gang Niu, Ivor W Tsang, James T Kwok, and Masashi Sugiyama. A survey of label-noise representation learning: Past, present and future. *arXiv preprint arXiv:2011.04406*, 2020. 1

[12] Bo Han, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor Tsang, and Masashi Sugiyama. Co-teaching: Robust training of deep neural networks with extremely noisy labels. In *Advances in neural information processing systems*, pages 8527–8537, 2018. 2, 7, 8

[13] Jiangfan Han, Ping Luo, and Xiaogang Wang. Deep self-learning from noisy labels. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5138–5147, 2019. 2, 5

[14] Jiayuan Huang, Arthur Gretton, Karsten Borgwardt, Bernhard Schölkopf, and Alex J Smola. Correcting sample selection bias by unlabeled data. In *Advances in neural information processing systems*, pages 601–608, 2007. 6, 17

[15] Lu Jiang, Di Huang, Mason Liu, and Weilong Yang. Beyond synthetic noise: Deep learning on controlled noisy labels. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119, pages 4804–4815. PMLR, 13–18 Jul 2020. 1

[16] Lu Jiang, Zhengyuan Zhou, Thomas Leung, Li-Jia Li, and Li Fei-Fei. Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels. In *International Conference on Machine Learning*, pages 2304–2313. PMLR, 2018. 2

[17] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009. 7

[18] Kuang-Huei Lee, Xiaodong He, Lei Zhang, and Linjun Yang. Cleannet: Transfer learning for scalable image classifier training with label noise. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5447–5456, 2018. 2

[19] Junnan Li, Richard Socher, and Steven C.H. Hoi. Dividemix: Learning with noisy labels as semi-supervised learning. In *International Conference on Learning Representations*, 2020. 2

[20] Xuefeng Li, Tongliang Liu, Bo Han, Gang Niu, and Masashi Sugiyama. Provably end-to-end label-noise learning without anchor points. *arXiv preprint arXiv:2102.02400*, 2021. 2

[21] Yuncheng Li, Jianchao Yang, Yale Song, Liangliang Cao, Jiebo Luo, and Li-Jia Li. Learning from noisy labels with distillation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1910–1918, 2017. 2

[22] Tongliang Liu and Dacheng Tao. Classification with noisy labels by importance reweighting. *IEEE Transactions on pattern analysis and machine intelligence*, 38(3):447–461, 2015. 2

[23] Yang Liu. The importance of understanding instance-level noisy labels, 2021. 1

[24] Yang Liu and Hongyi Guo. Peer loss functions: Learning from noisy labels without knowing noise rates. In *Proceedings of the 37th International Conference on Machine Learning*, ICML '20, 2020. 1, 2, 3, 4, 5, 7, 8, 12, 16

[25] Philip M Long and Rocco A Servedio. Random classification noise defeats all convex potential boosters. *Machine learning*, 78(3):287–304, 2010. 2

[26] Naresh Manwani and PS Sastry. Noise tolerance under risk minimization. *IEEE transactions on cybernetics*, 43(3):1146–1151, 2013. 2

[27] Nagarajan Natarajan, Inderjit S Dhillon, Pradeep K Ravikumar, and Ambuj Tewari. Learning with noisy labels. In *Advances in neural information processing systems*, pages 1196–1204, 2013. 1, 2, 3

[28] Duc Tam Nguyen, Chaithanya Kumar Mummadi, Thi Phuong Nhung Ngo, Thi Hoai Phuong Nguyen, Laura Beggel, and Thomas Brox. Self: Learning to filter noisy labels with self-ensembling. In *International Conference on Learning Representations*, 2020. 2

[29] Giorgio Patrini, Alessandro Rozza, Aditya Krishna Menon, Richard Nock, and Lizhen Qu. Making deep neural networks robust to label noise: A loss correction approach. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1944–1952, 2017. 1, 2, 7, 8

[30] Giorgio Patrini, Alessandro Rozza, Aditya Krishna Menon, Richard Nock, and Lizhen Qu. Making deep neural networks robust to label noise: A loss correction approach. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 2, 5

[31] Jun Shu, Qian Zhao, Keyu Chen, Zongben Xu, and Deyu Meng. Learning adaptive loss for robust learning with noisy labels. *arXiv preprint arXiv:2002.06482*, 2020. 2

[32] Arash Vahdat. Toward robustness against label noise in training deep discriminative neural networks. In *Advances in Neural Information Processing Systems*, pages 5596–5605, 2017. 1

[33] Andreas Veit, Neil Alldrin, Gal Chechik, Ivan Krasin, Abhinav Gupta, and Serge Belongie. Learning from noisy large-scale datasets with minimal supervision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 839–847, 2017. 2

[34] Qizhou Wang, Bo Han, Tongliang Liu, Gang Niu, Jian Yang, and Chen Gong. Tackling instance-dependent label noise via a universal probabilistic model. *arXiv preprint arXiv:2101.05467*, 2021. 1, 4

[35] Qizhou Wang, Jiangchao Yao, Chen Gong, Tongliang Liu, Mingming Gong, Hongxia Yang, and Bo Han. Learning with group noise. *arXiv preprint arXiv:2103.09468*, 2021. 1

[36] Yisen Wang, Xingjun Ma, Zaiyi Chen, Yuan Luo, Jinfeng Yi, and James Bailey. Symmetric cross entropy for robust learning with noisy labels. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 322–330, 2019. 2

[37] Hongxin Wei, Lei Feng, Xiangyu Chen, and Bo An. Combating noisy labels by agreement: A joint training method with co-regularization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13726–13735, 2020. 2, 7, 8

[38] Xiaobo Xia, Tongliang Liu, Bo Han, Chen Gong, Nannan Wang, Zongyuan Ge, and Yi Chang. Robust early-learning: Hindering the memorization of noisy labels. In *International Conference on Learning Representations*, 2021. 1

[39] Xiaobo Xia, Tongliang Liu, Bo Han, Nannan Wang, Jiankang Deng, Jiatong Li, and Yinian Mao. Extended T: Learning with mixed closed-set and open-set noisy labels. *arXiv preprint arXiv:2012.00932*, 2020. 2

[40] Xiaobo Xia, Tongliang Liu, Bo Han, Nannan Wang, Mingming Gong, Haifeng Liu, Gang Niu, Dacheng Tao, and Masashi Sugiyama. Part-dependent label noise: Towards instance-dependent label noise. In *Advances in Neural Information Processing Systems*, volume 33, pages 7597–7610, 2020. 1, 2, 4, 5, 7, 8, 17, 18

[41] Xiaobo Xia, Tongliang Liu, Nannan Wang, Bo Han, Chen Gong, Gang Niu, and Masashi Sugiyama. Are anchor points really indispensable in label-noise learning? In *Advances in Neural Information Processing Systems*, pages 6838–6849, 2019. 2, 5, 7, 8

[42] Tong Xiao, Tian Xia, Yi Yang, Chang Huang, and Xiaogang Wang. Learning from massive noisy labeled data for image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2691–2699, 2015. 1, 7, 8

[43] Yilun Xu, Peng Cao, Yuqing Kong, and Yizhou Wang. L_dmi: A novel information-theoretic loss function for training deep nets robust to label noise. In *Advances in Neural Information Processing Systems*, volume 32, 2019. 2, 3, 7, 8

[44] Quanming Yao, Hansi Yang, Bo Han, Gang Niu, and James T Kwok. Searching to exploit memorization effect in learning with noisy labels. In *Proceedings of the 37th International Conference on Machine Learning*, ICML '20, 2020. 2, 5

[45] Yu Yao, Tongliang Liu, Bo Han, Mingming Gong, Jiankang Deng, Gang Niu, and Masashi Sugiyama. Dual T: Reducing estimation error for transition matrix in label-noise learning. In *Advances in Neural Information Processing Systems*, volume 33, pages 7260–7271, 2020. 2

[46] Xingrui Yu, Bo Han, Jiangchao Yao, Gang Niu, Ivor W Tsang, and Masashi Sugiyama. How does disagreement help generalization against label corruption? *arXiv preprint arXiv:1901.04215*, 2019. 2, 7, 8

[47] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530*, 2016. 1

[48] Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*, 2018. 18

[49] Yivan Zhang, Gang Niu, and Masashi Sugiyama. Learning noise transition matrix from only noisy labels via total variation regularization. *arXiv preprint arXiv:2102.02414*, 2021. 2

[50] Zhilu Zhang and Mert Sabuncu. Generalized cross entropy loss for training deep neural networks with noisy labels. In *Advances in neural information processing systems*, pages 8778–8788, 2018. 2, 7, 8

[51] Zhaowei Zhu, Yiwen Song, and Yang Liu. Clusterability as an alternative to anchor points when learning with noisy labels. *arXiv preprint arXiv:2102.05291*, 2021. 2

Appendix

A. Proof for Lemmas

A.1. Proof for Lemma 1

Proof. We try to build the connection between noisy distribution $\widetilde{\mathcal{D}}$ and the underlying Bayes optimal distribution $\mathcal{D}^*$ by the noise rates e_+ and e_-. The primary difference from the proof of Lemma 2 in [24] is the usage of $\mathcal{D}^*$. Note:

$$
\begin{aligned}
&\mathbb{E}_{\widetilde{\mathcal{D}}}[\ell(f(X), \widetilde{Y})] \\
=&\mathbb{E}_{\mathcal{D}^*}\left[\sum_{j\in\{-1,+1\}} \mathbb{P}(\widetilde{Y}=j|X,Y^*)\ell(f(X),j)\right] \\
=&\mathbb{E}_{\mathcal{D}^*}\left[\sum_{j\in\{-1,+1\}} \mathbb{P}(\widetilde{Y}=j|Y^*)\ell(f(X),j)\right] \\
=&\sum_{i\in\{-1,+1\}} \mathbb{P}(Y^*=i)\mathbb{E}_{\mathcal{D}^*|Y^*=i}[\mathbb{P}(\widetilde{Y}=+1|Y^*=i)\ell(f(X),+1)+\mathbb{P}(\widetilde{Y}=-1|Y^*=i)\ell(f(X),-1)] \\
=&\mathbb{P}(Y^*=+1)\mathbb{E}_{\mathcal{D}^*|Y^*=+1}[(1-e_+)\ell(f(X),+1)+e_+\ell(f(X),-1)] \\
&+\mathbb{P}(Y^*=-1)\mathbb{E}_{\mathcal{D}^*|Y^*=-1}[(1-e_-)\ell(f(X),-1)+e_-\ell(f(X),+1)].
\end{aligned}
$$

Similarly, following the proof of Lemma 2 in [24], we can prove this lemma. $\square$

A.2. Proof for Lemma 2

Peer Loss on the Bayes Optimal Distribution Recall our goal is to learn a classifier f from the noisy distribution $\widetilde{\mathcal{D}}$ which also minimizes the loss on the corresponding Bayes optimal distribution D^*, i.e. $\mathbb{E}[\mathbb{1}(f(X), Y^*)], (X, Y^*) \sim \mathcal{D}^*$. Before considering the case with label noise, we need to prove peer loss functions induce the Bayes optimal classifier when minimizing the 0-1 loss on $\mathcal{D}^*$ as in Lemma 2.

Lemma 2. *Given the Bayes optimal distribution $\mathcal{D}^*$, the optimal peer classifier defined below:*

$$
f^*_{peer} = \arg\min_f \ \mathbb{E}_{\mathcal{D}^*}[\mathbb{1}_{PL}(f(X), Y^*)]
$$

also minimizes $\mathbb{E}_{\mathcal{D}^*}[\mathbb{1}(f(X), Y^*)]$.

See the proof below. It has been shown in [24] that Lemma 2 holds for the clean distribution $\mathcal{D}$ when the clean dataset is class-balanced, i.e. $\mathbb{P}(Y=-1) = \mathbb{P}(Y=+1) = 0.5$. For the Bayes optimal distribution $\mathcal{D}^*$, as shown in Lemma 2, there is *no requirement* for the prior $p^* := \mathbb{P}(Y^*=+1)$.

Proof. Recall Y^* is the Bayes optimal label defined as

$$
Y^*|X := \arg\max_Y \ \mathbb{P}(Y|X), (X,Y) \sim \mathcal{D}.
$$

We need to prove that the "optimal peer classifier" defined below:

$$
f^*_{\text{peer}} = \arg\min_f \ \mathbb{E}_{\mathcal{D}^*}[\mathbb{1}_{\text{PL}}(f(X), Y^*)]
$$

is the same as the Bayes optimal classifier f^*. To see this, suppose the claim is wrong. Denote by (notations ϵ_+ and ϵ_- are defined only for this proof):

$$
\epsilon_+ := \mathbb{P}(f^*_{\text{peer}}(X) = -1|f^*(X) = +1), \quad \epsilon_- := \mathbb{P}(f^*_{\text{peer}}(X) = +1|f^*(X) = -1)
$$

and denote by $p^* := \mathbb{P}(f^*(X) = +1)$. Then

$$
\begin{aligned}
&\mathbb{E}_{\mathcal{D}^*}[\mathbb{1}_{\mathrm{PL}}(f^*_{\mathrm{peer}}(X), Y^*)] \\
&= \mathbb{P}(f^*_{\mathrm{peer}}(X) \neq Y^*) - p^* \cdot \mathbb{P}(f^*_{\mathrm{peer}}(X) \neq +1) - (1-p^*) \cdot \mathbb{P}(f^*_{\mathrm{peer}}(X) \neq -1) \\
&= p^* \cdot \epsilon_+ + (1-p^*) \cdot \epsilon_- - p^* \cdot \mathbb{P}(f^*_{\mathrm{peer}}(X) \neq +1) - (1-p^*) \cdot \mathbb{P}(f^*_{\mathrm{peer}}(X) \neq -1) \\
&= p^* \cdot \epsilon_+ + (1-p^*) \cdot \epsilon_- \\
&\quad - p^* \cdot \left(\mathbb{P}(f^*_{\mathrm{peer}}(X) \neq +1 | f^*(X) \neq +1)\mathbb{P}(f^*(X) \neq +1) + \mathbb{P}(f^*_{\mathrm{peer}}(X) \neq +1 | f^*(X) \neq -1)\mathbb{P}(f^*(X) \neq -1)\right) \\
&\quad - (1-p^*) \cdot \left(\mathbb{P}(f^*_{\mathrm{peer}}(X) \neq -1 | f^*(X) \neq +1)\mathbb{P}(f^*(X) \neq +1) + \mathbb{P}(f^*_{\mathrm{peer}}(X) \neq -1 | f^*(X) \neq -1)\mathbb{P}(f^*(X) \neq -1)\right) \\
&= p^* \cdot \epsilon_+ + (1-p^*) \cdot \epsilon_- \\
&\quad - p^* \cdot \mathbb{P}(f^*(X) \neq +1)(1-\epsilon_-) - p^* \cdot \mathbb{P}(f^*(X) \neq -1) \cdot \epsilon_+ \\
&\quad - (1-p^*) \cdot \mathbb{P}(f^*(X) \neq -1)(1-\epsilon_+) - (1-p^*) \cdot \mathbb{P}(f^*(X) \neq +1) \cdot \epsilon_- \\
&= 0 - p^* \cdot \mathbb{P}(f^*(X) \neq +1) - (1-p^*) \cdot \mathbb{P}(f^*(X) \neq -1) \\
&\quad + p^*(\epsilon_+ + \mathbb{P}(f^*(X) \neq +1)\epsilon_- - \mathbb{P}(f^*(X) \neq -1)\epsilon_+) \\
&\quad + (1-p^*)(\epsilon_- + \mathbb{P}(f^*(X) \neq -1)\epsilon_+ - \mathbb{P}(f^*(X) \neq +1)\epsilon_-) \\
&> 0 - p^* \cdot \mathbb{P}(f^*(X) \neq +1) - (1-p^*) \cdot \mathbb{P}(f^*(X) \neq -1) \\
&= \mathbb{E}_{\mathcal{D}^*}[\mathbb{1}_{\mathrm{PL}}(f^*(X), Y^*)]
\end{aligned}
$$

contradicting the optimality of f^*_{peer}. Thus our claim is proved. $\qquad\square$

B. Proof for Theorems

B.1. Proof for Theorem 1

Proof. The covariance $\mathrm{Cov}(\cdot, \cdot)$ in this proof is taken over the Bayes optimal distribution $\mathcal{D}^*$. The following proof is built on the result of Theorem 2, i.e. Eq. (2). First note

$$
\begin{aligned}
\mathrm{Cov}(Z_1(X), \mathbb{1}(f_1(X), Y^*) - \mathbb{1}(f_2(X), Y^*)) &= \mathbb{E}[(Z_1(X) - \mathbb{E}[Z_1(X)]) \cdot (\mathbb{1}(f_1(X), Y^*) - \mathbb{1}(f_2(X), Y^*))] \\
&\leq \mathbb{E}[|(Z_1(X) - \mathbb{E}[Z_1(X)]|] \\
&\leq \mathbb{E}|e_+(X) - \mathbb{E}[e_+(X)]| + \mathbb{E}|e_-(X) - \mathbb{E}[e_-(X)]|
\end{aligned}
$$

Similarly, one can show that

$$
\mathrm{Cov}(Z_2(X), \mathbb{1}(f_1(X), -1) - \mathbb{1}(f_2(X), -1)) \leq \mathbb{E}|e_+(X) - \mathbb{E}[e_+(X)]| + \mathbb{E}|e_-(X) - \mathbb{E}[e_-(X)]|
$$

Now with bounded variance in the error rates, suppose:

$$
\mathbb{E}|e_+(X) - \mathbb{E}[e_+(X)]| \leq \epsilon_+, \quad \mathbb{E}|e_-(X) - \mathbb{E}[e_-(X)]| \leq \epsilon_-
$$

Note

$$
\begin{aligned}
\tilde{f}^*_{\mathrm{peer}} &:= \arg\min_f \mathbb{E}_{\widetilde{\mathcal{D}}}\left[\mathbb{1}_{\mathrm{PL}}(f(X), \tilde{Y})\right] \\
&= \arg\min_f \left[(1 - e_+ - e_-)\mathbb{E}_{\mathcal{D}^*}[\mathbb{1}_{\mathrm{PL}}(f(X), Y^*) + \mathrm{Cov}(Z_1(X), \mathbb{1}(f(X), Y^*)) + \mathrm{Cov}(Z_2(X), \mathbb{1}(f(X), -1))\right] \\
&= \arg\min_f \Big[(1 - e_+ - e_-)\left(\mathbb{E}_{\mathcal{D}^*}[\mathbb{1}(f(X), Y^*) - p^* \cdot \mathbb{E}_{\mathcal{D}^*}[\mathbb{1}(f(X), +1)] - (1-p^*) \cdot \mathbb{E}_{\mathcal{D}^*}[\mathbb{1}(f(X), -1)]\right) \\
&\quad + \mathrm{Cov}(Z_1(X), \mathbb{1}(f(X), Y^*)) + \mathrm{Cov}(Z_2(X), \mathbb{1}(f(X), -1))\Big].
\end{aligned}
$$

Then

$$
\mathbb{E}_{\mathcal{D}^*}\left[\mathbb{1}(\tilde{f}^*_{\text{peer}}(X), Y^*)\right] + \frac{\text{Cov}(Z_1(X), \mathbb{1}(\tilde{f}^*_{\text{peer}}(X), Y^*)) + \text{Cov}(Z_2(X), \mathbb{1}(\tilde{f}^*_{\text{peer}}(X), -1))}{1 - e_+ - e_-}
$$

$$
=\mathbb{E}_{\mathcal{D}^*}\left[\mathbb{1}(\tilde{f}^*_{\text{peer}}(X), Y^*)\right] - 0.5 \cdot \mathbb{E}_X[\mathbb{1}(\tilde{f}^*_{\text{peer}}(X), +1)] - 0.5 \cdot \mathbb{E}_X[\mathbb{1}(\tilde{f}^*_{\text{peer}}(X), -1)] + 0.5
$$
$$
+ \frac{\text{Cov}(Z_1(X), \mathbb{1}(\tilde{f}^*_{\text{peer}}(X), Y^*)) + \text{Cov}(Z_2(X), \mathbb{1}(\tilde{f}^*_{\text{peer}}(X), -1))}{1 - e_+ - e_-}
$$

$$
\leq\mathbb{E}_{\mathcal{D}^*}\left[\mathbb{1}(\tilde{f}^*_{\text{peer}}(X), Y^*)\right] - p^* \cdot \mathbb{E}_X[\mathbb{1}(\tilde{f}^*_{\text{peer}}(X), +1)] - (1 - p^*) \cdot \mathbb{E}_X[\mathbb{1}(\tilde{f}^*_{\text{peer}}(X), -1)] + |p^* - 0.5| + 0.5
$$
$$
+ \frac{\text{Cov}(Z_1(X), \mathbb{1}(\tilde{f}^*_{\text{peer}}(X), Y^*)) + \text{Cov}(Z_2(X), \mathbb{1}(\tilde{f}^*_{\text{peer}}(X), -1))}{1 - e_+ - e_-}
$$

$$
\leq\mathbb{E}_{\mathcal{D}^*}\left[\mathbb{1}(f^*(X), Y^*)\right] - p^* \cdot \mathbb{E}_X[\mathbb{1}(f^*(X), +1)] - (1 - p^*) \cdot \mathbb{E}_X[\mathbb{1}(f^*(X), -1)] + |p^* - 0.5| + 0.5
$$
$$
+ \frac{\text{Cov}(Z_1(X), \mathbb{1}(f^*(X), Y^*)) + \text{Cov}(Z_2(X), \mathbb{1}(f^*(X), -1))}{1 - e_+ - e_-}
$$

$$
\leq\mathbb{E}_{\mathcal{D}^*}\left[\mathbb{1}(f^*(X), Y^*)\right] + \frac{\text{Cov}(Z_1(X), \mathbb{1}(f^*(X), Y^*)) + \text{Cov}(Z_2(X), \mathbb{1}(f^*(X), -1))}{1 - e_+ - e_-} + 2|p^* - 0.5|.
$$

Thus

$$
\mathbb{E}_{\mathcal{D}^*}\left[\mathbb{1}(\tilde{f}^*_{\text{peer}}(X), Y^*) - \mathbb{1}(f^*(X), Y^*)\right]
$$
$$
\leq\frac{\text{Cov}(Z_1(X), \mathbb{1}(f^*(X), Y^*) - \mathbb{1}(\tilde{f}^*_{\text{peer}}(X), Y^*)) + \text{Cov}(Z_2(X), \mathbb{1}(f^*(X), -1) - \mathbb{1}(\tilde{f}^*_{\text{peer}}(X), -1))}{1 - e_+ - e_-} + 2|p^* - 0.5|
$$
$$
\leq 2\frac{\mathbb{E}|e_+(X) - \mathbb{E}[e_+(X)]| + \mathbb{E}|e_-(X) - \mathbb{E}[e_-(X)]|}{1 - e_+ - e_-} + 2|p^* - 0.5|
$$
$$
\leq\frac{2(\epsilon_+ + \epsilon_-)}{1 - e_+ - e_-} + 2|p^* - 0.5|.
$$

Noting $\mathbb{1}(f^*(X), Y^*) = 0$, we finish the proof.

$\square$

B.2. Proof for Theorem 2

Proof. The covariance $\text{Cov}(\cdot, \cdot)$ in this proof is taken over the Bayes optimal distribution $\mathcal{D}^*$. Recall

$$
e_+(X) := \mathbb{P}(\widetilde{Y} = -1|Y^* = +1, X), e_-(X) := \mathbb{P}(\widetilde{Y} = +1|Y^* = -1, X)
$$

and

$$
e_+ := \mathbb{E}_X[e_+(X)], \quad e_- := \mathbb{E}_X[e_-(X)]
$$

We first have the following equality:

$$
\begin{aligned}
\mathbb{E}_{\widetilde{\mathcal{D}}}[\mathbb{1}_{\text{PL}}(f(X), \tilde{Y})] = {}& \mathbb{E}_{\mathcal{D}^*}[(1 - e_+(X) - e_-(X))\mathbb{1}(f(X), Y^*))] && \text{(Term-A)} \\
& + \mathbb{E}_X[e_+(X)\mathbb{1}(f(X), -1) + e_-(X)\mathbb{1}(f(X), +1)] && \text{(Term-B)} \\
& - (1 - e_+ - e_-) \cdot \mathbb{E}_{\mathcal{D}^*}[\mathbb{1}(f(X), Y^*_p))] && \text{(Term-C)} \\
& - \mathbb{E}_X[e_+ \cdot \mathbb{1}(f(X), -1) + e_- \cdot \mathbb{1}(f(X), +1)] && \text{(Term-D)}
\end{aligned}
$$

Term-B can be transformed to:

$$
\begin{aligned}
\mathbb{E}_X[e_+(X) \cdot \mathbb{1}(f(X), -1) &+ e_-(X) \cdot \mathbb{1}(f(X), +1)] \\
&= \mathbb{E}_X[e_+(X) \cdot \mathbb{1}(f(X), -1) + e_-(X) \cdot (1 - \mathbb{1}(f(X), -1))] \\
&= \mathbb{E}_X[(e_+(X) - e_-(X)) \cdot \mathbb{1}(f(X), -1) + e_-(X)].
\end{aligned}
$$

Similarly, Term-D turns to

$$\mathbb{E}_X[e_+ \cdot \mathbb{1}(f(X), -1) + e_- \cdot \mathbb{1}(f(X), +1)] = (e_+ - e_-) \cdot \mathbb{E}_X[\mathbb{1}(f(X), -1)] + e_-.$$

Define two random variables

$$Z_1(X) := 1 - e_+(X) - e_-(X), \ Z_2(X) = e_+(X) - e_-(X).$$

Then Term-A becomes

$$\begin{aligned}
&\mathbb{E}_{\mathcal{D}^*}[(1 - e_+(X) - e_-(X))\mathbb{1}(f(X), Y^*))] \\
&= \mathbb{E}[Z_1(X)] \cdot \mathbb{E}_{\mathcal{D}^*}[\mathbb{1}(f(X), Y^*))] + \mathrm{Cov}(Z_1(X), \mathbb{1}(f(X), Y^*)) \\
&= (1 - e_+ - e_-) \cdot \mathbb{E}_{\mathcal{D}^*}[\mathbb{1}(f(X), Y^*))] + \mathrm{Cov}(Z_1(X), \mathbb{1}(f(X), Y^*))
\end{aligned}$$

Similarly, Term-B can be further transformed to

$$\begin{aligned}
&\mathbb{E}_X[(e_+(X) - e_-(X)) \cdot \mathbb{1}(f(X), -1) + e_-(X)] \\
&= \mathbb{E}[Z_2(X)]\mathbb{E}_X[\mathbb{1}(f(X), -1)] + \mathrm{Cov}(Z_2(X), \mathbb{1}(f(X), -1)) + e_- \\
&= (e_+ - e_-)\mathbb{E}_X[\mathbb{1}(f(X), -1)] + \mathrm{Cov}(Z_2(X), \mathbb{1}(f(X), -1)) + e_-
\end{aligned}$$

Combining the above results, we have

$$\begin{aligned}
\mathbb{E}_{\widetilde{\mathcal{D}}}[\mathbb{1}_{\mathrm{PL}}(f(X), \widetilde{Y})] &= (1 - e_+ - e_-) \cdot \mathbb{E}_{\mathcal{D}^*}[\mathbb{1}(f(X), Y^*))] \\
&+ (e_+ - e_-)\mathbb{E}_X[\mathbb{1}(f(X), -1)] + e_- \\
&- (1 - e_+ - e_-) \cdot \mathbb{E}_{\mathcal{D}^*}[\mathbb{1}(f(X), Y_p^*))] \\
&- (e_+ - e_-) \cdot \mathbb{E}_X[\mathbb{1}(f(X), -1)] - e_- \\
&+ \mathrm{Cov}(Z_1(X), \mathbb{1}(f(X), Y)) + \mathrm{Cov}(Z_2(X), \mathbb{1}(f(X), -1)) \\
&= (1 - e_+ - e_-)\mathbb{E}_{\mathcal{D}^*}[\mathbb{1}_{\mathrm{PL}}(f(X), Y^*)] \\
&+ \mathrm{Cov}(Z_1(X), \mathbb{1}(f(X), Y^*)) + \mathrm{Cov}(Z_2(X), \mathbb{1}(f(X), -1))
\end{aligned}$$

$\square$

B.3. Proof for Theorem 3

Proof. From Theorem 2, we know

$$\left. \mathbb{E}_{\widetilde{\mathcal{D}}}[\mathbb{1}_{\mathrm{PL}}(f(X), \widetilde{Y})] - \mathrm{Cov}(Z_1(X), \mathbb{1}(f(X), Y^*)) - \mathrm{Cov}(Z_2(X), \mathbb{1}(f(X), -1)) \right]$$

$$=(1 - e_- - e_+) \cdot \mathbb{E}_{\mathcal{D}^*}[\mathbb{1}_{\mathrm{PL}}(f(X), Y^*)].$$

With Lemma 2, we can finish the proof. $\square$

B.4. Proof for Theorem 4

Proof. Recall $\tau \in [0, 1]$ is the expected ratio (a.k.a. probability) of correct examples in $\hat{D}^\tau$, i.e. $\tau = \mathbb{E}[\mathbb{1}\{(X, \hat{Y}) \in \hat{D}^\tau | (X, Y^*) \in D^*\}] = \mathbb{P}((X, \hat{Y}) \sim \hat{\mathcal{D}}^\tau | (X, Y^*) \sim \mathcal{D}^*)$. With $\hat{D}^\tau$, the classifier learned by minimizing the 0-1 CAL loss is

$$\tilde{f}^*_{\mathrm{CAL}\text{-}\tau} := \arg\min_f \mathbb{E}_{\widetilde{\mathcal{D}}}\left[\mathbb{1}_{\mathrm{PL}}(f(X), \widetilde{Y})] - \mathrm{Cov}_{\hat{\mathcal{D}}^\tau}(Z_1(X), \mathbb{1}(f(X), \hat{Y})) - \mathrm{Cov}_{\hat{\mathcal{D}}^\tau}(Z_2(X), \mathbb{1}(f(X), -1))\right].$$

 Note

$$\begin{aligned}
\mathrm{Cov}_{\hat{\mathcal{D}}^\tau}(Z_1(X), \mathbb{1}(f(X), Y)) &= \mathbb{E}_{\hat{\mathcal{D}}^\tau}\left[(Z_1(X) - \mathbb{E}_{\hat{\mathcal{D}}^\tau}[Z_1(X)])\left(\mathbb{1}(f(X), Y) - \mathbb{E}_{\hat{\mathcal{D}}^\tau}[\mathbb{1}(f(X), Y)]\right)\right] \\
&= \mathbb{E}_{\hat{\mathcal{D}}^\tau}\left[(Z_1(X) - \mathbb{E}_{\hat{\mathcal{D}}^\tau}[Z_1(X)])\mathbb{1}(f(X), Y)\right] \\
&= \mathbb{P}((X, Y) \in D^*|(X, Y) \in \hat{D}^\tau)\mathbb{E}_{\hat{\mathcal{D}}^\tau}\left[(Z_1(X) - \mathbb{E}_{\hat{\mathcal{D}}^\tau}[Z_1(X)])\mathbb{1}(f(X), Y)|(X, Y) \in D^*\right] \\
&+ \mathbb{P}((X, Y) \notin D^*|(X, Y) \in \hat{D}^\tau)\mathbb{E}_{\hat{\mathcal{D}}^\tau}\left[(Z_1(X) - \mathbb{E}_{\hat{\mathcal{D}}^\tau}[Z_1(X)])\mathbb{1}(f(X), Y)|(X, Y) \notin D^*\right].
\end{aligned}$$

Similarly,

$$
\begin{aligned}
\mathrm{Cov}_{\mathcal{D}^*}(Z_1(X), \mathbb{1}(f(X), Y)) &= \mathbb{E}_{\mathcal{D}^*}\left[(Z_1(X) - \mathbb{E}_{\mathcal{D}^*}[Z_1(X)])\,\mathbb{1}(f(X), Y)\right] \\
&= \mathbb{P}((X, Y) \in \hat{D}^\tau | (X, Y) \in D^*)\mathbb{E}_{\mathcal{D}^*}\left[(Z_1(X) - \mathbb{E}_{\mathcal{D}^*}[Z_1(X)])\,\mathbb{1}(f(X), Y) | (X, Y) \in \hat{D}^\tau\right] \\
&\quad + \mathbb{P}((X, Y) \notin \hat{D}^\tau | (X, Y) \in D^*)\mathbb{E}_{\mathcal{D}^*}\left[(Z_1(X) - \mathbb{E}_{\mathcal{D}^*}[Z_1(X)])\,\mathbb{1}(f(X), Y) | (X, Y) \notin \hat{D}^\tau\right].
\end{aligned}
$$

When D^*, $\hat{D}^\tau$ and $\tilde{D}$ have the same feature set, we have

$$
\mathbb{P}((X, Y) \in D^* | (X, Y) \in \hat{D}^\tau) = \mathbb{P}((X, Y) \in \hat{D}^\tau | (X, Y) \in D^*) = \tau,
$$
$$
\mathbb{P}((X, Y) \notin D^* | (X, Y) \in \hat{D}^\tau) = \mathbb{P}((X, Y) \notin \hat{D}^\tau | (X, Y) \in D^*) = 1 - \tau.
$$

Therefore,

$$
\mathrm{Cov}_{\hat{D}^\tau}(Z_1(X), \mathbb{1}(f(X), Y)) - \mathrm{Cov}_{\mathcal{D}^*}(Z_1(X), \mathbb{1}(f(X), Y)) \le 2(1 - \tau)(\epsilon_+ + \epsilon_-).
$$

The rest of the proof can be accomplished by following the proof of Theorem 1. $\qquad\square$

C. Proof for Corollaries

C.1. Proof for Corollary 1

Proof.

$$
\mathbb{E}_{\tilde{\mathcal{D}}}[\ell_{\mathrm{PL}}(f(X), \tilde{Y})] = \mathbb{E}_{\tilde{\mathcal{D}}}[\ell(f(X), \tilde{Y})] - \mathbb{E}_{\tilde{\mathcal{D}}_Y}\left[\mathbb{E}_{\mathcal{D}_X}[\ell(f(X_p), \tilde{Y}_p)]\right]. \tag{5}
$$

The first term in (5) is

$$
\begin{aligned}
&\mathbb{E}_{\tilde{\mathcal{D}}}[\ell(f(X), \tilde{Y})] \\
=&\mathbb{E}_{\mathcal{D}^*}\left[\sum_{j \in [K]} \mathbb{P}(\tilde{Y} = j | X, Y^*)\ell(f(X), j)\right] \\
=&\sum_{j \in [K]}\sum_{i \in [K]} \mathbb{P}(Y^* = i)\mathbb{E}_{\mathcal{D}^* | Y^* = i}[T_{ij}(X)\ell(f(X), j)] \\
=&\sum_{j \in [K]}\sum_{i \in [K]} \mathbb{P}(Y^* = i)\left[T_{ij}\mathbb{E}_{\mathcal{D}^* | Y^* = i}[\ell(f(X), j)] + \mathrm{Cov}_{\mathcal{D}^* | Y^* = i}[T_{ij}(X), \ell(f(X), j)]\right] \\
=&\sum_{j \in [K]}\left[\mathbb{P}(Y^* = j)\left(1 - \sum_{i \ne j, i \in [K]} T_{ji}\right)\mathbb{E}_{\mathcal{D}^* | Y^* = j}[\ell(f(X), j)] + \sum_{i \in [K], i \ne j} \mathbb{P}(Y^* = i)T_{ij}\mathbb{E}_{\mathcal{D}^* | Y^* = i}[\ell(f(X), j)]\right] \\
&\quad + \sum_{j \in [K]}\sum_{i \in [K]} P(Y^* = i)\mathrm{Cov}_{\mathcal{D}^* | Y^* = i}[T_{ij}(X), \ell(f(X), j)] \\
=&\sum_{j \in [K]}\left[\mathbb{P}(Y^* = j)\left(1 - \sum_{i \ne j, i \in [K]} e_i\right)\mathbb{E}_{\mathcal{D}^* | Y^* = j}[\ell(f(X), j)] + \sum_{i \in [K], i \ne j} \mathbb{P}(Y^* = i)e_j\mathbb{E}_{\mathcal{D}^* | Y^* = i}[\ell(f(X), j)]\right] \\
&\quad + \sum_{j \in [K]}\sum_{i \in [K]} P(Y^* = i)\mathrm{Cov}_{\mathcal{D}^* | Y^* = i}[T_{ij}(X), \ell(f(X), j)] \\
=&\left(1 - \sum_{i \in [K]} e_i\right)\mathbb{E}_{\mathcal{D}^*}[\ell(f(X), Y^*)] + \sum_{j \in [K]}\sum_{i \in [K]} \mathbb{P}(Y^* = i)e_j\mathbb{E}_{\mathcal{D}^* | Y^* = i}[\ell(f(X), j)] \\
&\quad + \sum_{j \in [K]}\sum_{i \in [K]} P(Y^* = i)\mathrm{Cov}_{\mathcal{D}^* | Y^* = i}[T_{ij}(X), \ell(f(X), j)]
\end{aligned}
$$

The rest of proofs can be done following standard multi-class peer loss derivations [24].

$\qquad\square$

D. More Discussions

D.1. Setting Thresholds $L_{\min}$ and $L_{\max}$

In a high level, there are two strategies for setting $L_{\min}$ and $L_{\max}$: 1) $L_{\min} < L_{\max}$ and 2) $L_{\min} = L_{\max}$.

Strategy-1: $L_{\min} < L_{\max}$: This strategy may provide a higher ratio of true Bayes optimal labels among feasible examples in $\hat{D}$ since some ambiguous examples are dropped. However, dropping examples changes the distribution of X (as well as the distribution of the unobservable Y^*), a.k.a. covariate shift [14, 6]. Importance re-weighting with weight $\gamma(X)$ is necessary for correcting the covariate shift, i.e. the weight of each feasible example $(x, \hat{y}) \in \hat{D}$ should be changed from 1 to $\gamma(x)$. Let $\mathcal{D}_X$ and $\hat{\mathcal{D}}_X$ be the marginal distributions of $\mathcal{D}$ and $\hat{\mathcal{D}}$ on X. With a particular kernel $\Phi(X)$, the optimization problem is:

$$
\begin{aligned}
\min_{\gamma(X)} \quad & \|\mathbb{E}_{\mathcal{D}_X}[\Phi(X)] - \mathbb{E}_{\hat{\mathcal{D}}_X}[\gamma(X)\Phi(X)]\| \\
\text{s.t.} \quad & \gamma(X) > 0 \text{ and } \mathbb{E}_{\hat{\mathcal{D}}_X}[\gamma(X)] = 1.
\end{aligned}
\tag{6}
$$

The optimal solution is supposed to be $\gamma^*(X) = \frac{\mathbb{P}_{\mathcal{D}_X}(X)}{\mathbb{P}_{\hat{\mathcal{D}}_X}(X)}$. Note the selection of kernel $\Phi(\cdot)$ is non-trivial, especially for complicated features [7] in DNN solutions. Using this strategy, with appropriate $L_{\min}$ and $L_{\max}$ such that all the examples in $\hat{D}$ are Bayes optimal, the covariance could be guaranteed to be optimal when each example in $\hat{D}$ is re-weighted by $\gamma^*(X)$.

Strategy-2: $L_{\min} = L_{\max}$: Compared with Strategy-1, we effectively lose one degree of freedom for getting a better $\hat{D}$. However, this is not entirely harmful since $\hat{D}$ and D^* have the same feature set, indicating estimating $\gamma(X)$ is no longer necessary and $\gamma(X) = 1$ is an optimal solution for (6) with this strategy.

Strategy selection When we can get a high-quality $\hat{D}$ by fine-tuning $L_{\min}$ and $L_{\max}$ or $\hat{D}$ is already provided from other sources, we may solve the optimization problem in (6) to find the optimal weight $\gamma(X)$. However, considering the fact that estimating $\gamma(X)$ introduces extra computation and potentially extra errors, we focus on Strategy-2 in this paper. Using Strategy-2 also reduces the effort on tuning hyperparameters. Besides, the proposed CAL loss is tolerant of an imperfect $\hat{D}$ (shown theoretically in Section 4.3).

D.2. Generation of Instance-Dependent Label Noise

Pseudo codes for generate instance-based label noise are provided in Algorithm 2. This algorithm follows the state-of-the-art method [40]. Define the overall noise rate as η.

Algorithm 2: Generating Instance-Dependent Label Noise

    Input: Clean examples $(x_n, y_n)_{n=1}^N$; Noise rate: η; Number of classes: K; Shape of each feature x_n: $S \times 1$.
1 Sample instance flip rates q_n from the truncated normal distribution $\mathcal{N}(\eta, 0.1^2, [0, 1])$;   *// mean η, variance 0.1^2, range $[0, 1]$*
2 Sample $W \in \mathcal{R}^{S \times K}$ from the standard normal distribution $\mathcal{N}(0, 1^2)$;
3 **for** $n \in [N]$ **do**
4      $p = x_n^\top W$ *// Generate instance dependent flip rates. The size of p is $1 \times K$.*
5      $p_{y_n} = -\infty$ *// Only consider entries that are different from the true label*
6      $p = q_n \cdot \text{softmax}(p)$ *// Let q_n be the probability of getting a wrong label*
7      $p_{y_n} = 1 - q_n$ *// Keep clean w.p. $1 - q_n$*
8   Randomly choose a label from the label space as noisy label $\tilde{y}_n$ according to p;
9 **end**
    Output: Noisy examples $\{(x_i, \tilde{y}_n), n \in [N]\}$.

Note Algorithm 2 cannot ensure $T_{ii}(X) > T_{ij}(X)$ when $\eta > 0.5$. To generate an informative dataset, we set $0.9 \cdot T_{ii}(X)$ as the upper bound of $T_{ij}(X)$ and distribute the remaining probability to other classes.

D.3. Performance without Data Augmentations

For a fair comparison with the recent work on instance-dependent label noise [40], we adopt the same data augmentations as [40] and re-produce their results using the same noise file as we employed in Table 1. Each noise rate is tested 5 times with a different generation matrix W (defined in Algorithm 2). Table 4 shows the advantages of our second-order approach.

Table 4. Performance comparisons without data augmentations

Method	$\eta = 0.2$	$\eta = 0.4$
PTD-R-V[40]	69.62 ± 3.35	64.73 ± 3.64
CAL	75.52 ± 3.94	70.30 ± 2.96

D.4. More Implementation Details on Clothing1M

Construct $\hat{D}$ We first train the network for 120 epochs on 1 million noisy training images using the method in [5]. The batch-size is set to 32. The initial learning rate is set as 0.01 and reduced by a factor of 10 at 30, 60, 90 epochs. We sample 1000 mini-batches from the training data for each epoch while ensuring the (noisy) labels are balanced. Mixup [48] is adopted for data augmentations. Hyperparameter β is set to 0 at first 80 epochs, and linearly increased to 0.4 for next 20 epochs and kept as 0.4 for the rest of the epochs. We construct $\hat{D}$ with the best model.

Train with CAL We change the loss to the CAL loss after getting $\hat{D}$ and continue training the model (without mixup) with an initial learning rate of 10^{-5} for 120 epochs (reduced by a factor of 10 at 30, 60, 90 epochs). We also tested re-train the model with $\hat{D}$ and get an accuracy of 73.56. A randomly-collected balanced dataset with $18,976$ noisy examples in each class is employed in training with CAL. Examples that are not in this balanced dataset are removed from $\hat{D}$ for ease of implementation.