Class-Wise Denoising for Robust Learning under Label Noise

Chen Gong, *Member, IEEE,* Yongliang Ding, Bo Han, Gang Niu, Jian Yang, *Member, IEEE,* Jane You, Dacheng Tao, *Fellow, IEEE,* Masashi Sugiyama, *Senior Member, IEEE*

Abstract—Label noise is ubiquitous in many real-world scenarios which often misleads training algorithm and brings about the degraded classification performance. Therefore, many approaches have been proposed to correct the loss function given corrupted labels to combat such label noise. Among them, a trend of works achieve this goal by unbiasedly estimating the data centroid, which plays an important role in constructing an unbiased risk estimator for minimization. However, they usually handle the noisy labels in different classes all at once, so the local information inherited by each class is ignored which often leads to unsatisfactory performance. To address this defect, this paper presents a novel robust learning algorithm dubbed "**C**lass-**W**ise **D**enoising" (CWD), which tackles the noisy labels in a class-wise way to ease the entire noise correction task. Specifically, two virtual auxiliary sets are respectively constructed by presuming that the positive and negative labels in the training set are clean, so the original false-negative labels and false-positive ones are tackled separately. As a result, an improved centroid estimator can be designed which helps to yield more accurate risk estimator. Theoretically, we prove that: 1) The variance in centroid estimation can often be reduced by our CWD when compared with existing methods with unbiased centroid estimator; and 2) The performance of CWD trained on the noisy set will converge to that of the optimal classifier trained on the clean set with a convergence rate $\mathcal{O}(\frac{1}{\sqrt{n}})$ where n is the number of the training examples. These sound theoretical properties critically enable our CWD to produce the improved classification performance under label noise, which is also demonstrated by the comparisons with ten representative state-of-the-art methods on a variety of benchmark datasets.

Index Terms—Label noise, Centroid estimation, Unbiasedness, Variance reduction.

1 Introduction

Traditional supervised machine learning algorithms usually require that training examples are all correctly labeled, otherwise their performance will decrease significantly due to the misleading supervision information. However, in many practical situations, the accurate labels of examples may be difficult to obtain due to various subjective or objective factors such as unavoidable human fatigue, limitation of human knowledge, measurement error of instruments, unreliable automatic labeling processes, etc. Therefore, it is highly desirable that some robust learning approaches can be designed to make the training process robust to noisy labels [1].

The existing methods for dealing with label noise can be roughly classified into three types, namely correctly-labeled data identification (or equivalently incorrectly-labeled data removal) [2], [3], [4], [5], robust loss design [6], [7], [8], and label-flip-rate based loss correction [9], [10], [11], [12]. Among them, correctly-labeled data identification or incorrectly-labeled data removal is perhaps the most straightforward way for tackling noisy labels, which aims to find the accurately labeled or mislabeled data to eliminate the negative impacts of noisy labels on training. The early-staged methods usually follow this idea which focus on noise detection and filtering [2], [13], namely the data are preprocessed to remove the possible noise ahead of conducting the standard algorithms. Due to the popularity of deep learning, correctly-labeled data identification is also adapted to various neural networks based on the "memorization" effect inherited by networks [14]. That is to say, the neural networks will fit correct and easy patterns in initial epochs and then move to the incorrect and difficult patterns in later epochs. Therefore, examples incurring small training loss values (*a.k.a.* small-loss data) can be selected in each epoch to reliably update the network. The typical methods include MentorNet [15], co-teaching [3], co-teaching+ [16], co-regularization [17], Search to Exploit [18], the curriculum

- *C. Gong was supported by NSF of China (No: 61973162), NSF of Jiangsu Province (No: BZ2021013), the Fundamental Research Funds for the Central Universities (Nos: 30920032202, 30921013114), and "111 Program" (No: B13022). B. Han was supported by the NSFC Young Scientists Fund (No: 62006202) and RGC Early Career Scheme (No: 22200720). J. You was supported by the Hong Kong Polytechnic University grants (Nos: YZ3K, UAJP/UAGK, and ZVRH). M. Sugiyama was supported by KAKENHI (No: 20H04206).*
- *C. Gong and Y. Ding are with the Key Laboratory of Intelligent Perception and Systems for High-Dimensional Information of Ministry of Education, the Jiangsu Key Laboratory of Image and Video Understanding for Social Security, and School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, P.R. China.*
  E-mail: {chen.gong, 1242388760}@njust.edu.cnn
- *B. Han is with the Department of Computer Science, Hong Kong Baptist University, Hong Kong, P.R. China.*
  E-mail: bhanml@comp.hkbu.edu.hk
- *G. Niu is with RIKEN Center for Advanced Intelligence Project, Tokyo, Japan.*
  E-mail: gang.niu.ml@gmail.com
- *J. Yang is with the College of Computer Science, Nankai University, Tianjin, P.R. China.*
  E-mail: csjyang@nankai.edu.cn
- *J. You is with the Department of Computing, Hong Kong Polytechnic University, Hong Kong, P.R. China.*
  E-mail: jane.you@polyu.edu.hk
- *D. Tao is with the JD Explore Academy, China and the University of Sydney, Australia.*
  E-mail: dacheng.tao@gmail.com
- *M. Sugiyama is with RIKEN Center for Advanced Intelligence Project, Tokyo, Japan; and is also with the Graduate School of Frontier Sciences, The University of Tokyo, Chiba, Japan.*
  E-mail: sugi@k.u-tokyo.ac.jp
- *Corresponding authors: C. Gong and J. Yang.*

loss [4], and SIGUA [19], etc. The main difference among these methods lies in how to find possible correctly labeled training data. For the identified mislabeled examples, some methods further conduct label correction via semi-supervised learning [5], label distribution learning [20], joint network optimization and true label estimation [21], conditional random field [22], etc.

However, the data selection or filtering process mentioned above are quite empirical and are short of theoretical guarantee, so this type of methods cannot stably generate good performance. Therefore, the second trend of research on label-noise learning tries to devise various noise-robust loss functions. These methods are usually largely based on the traditional cross entropy loss for learning with clean data. For example, Ghosh *et al.* [6] presented the Mean Absolute Error, which has been further extended by the Generalized Cross Entropy loss [23] that employs a negative Box-Cox transformation. Ma *et al.* [7] showed that the Mean Absolute Error can be made more robust to label noise by applying simple normalization, based on which they devised the Active Passive loss. Besides, Wang *et al.* [8] found that the plain use of the Cross Entropy loss can be class-biased, so they proposed the Symmetric Cross Entropy loss inspired by the symmetric Kullback-Leibler divergence. Recently, Hu *et al.* [24] devised Robust Clustering loss to make the deep networks focus on clean examples instead of noisy ones. Feng *et al.* [25] proposed the Taylor Cross Entropy Loss by explicitly controlling the order of Taylor Series for the cross entropy loss, so that the proposed loss function integrates the advantages of various robust loss functions.

The last type of methods for tackling noisy labels is label-flip-rate based loss correction, which has gained intensive attention recently and aims to correct the conventional loss functions based on the estimated label flip rate from one class to another. A pioneering work is [9] which proposed a simple weighted surrogate loss that is provably noise-tolerant. A key problem in [9] was to estimate the label flip rate, so [26] assumed that there exist a handful of clean data known as "anchor points" and further proposed an importance reweighing technique. Differently, [10] devised a backward correction operation specifically for deep neural networks to combat label noise. All the above methods require the anchor points to accurately estimate the label flip rate which may not be available in practical situations. Thereby, [27] and [28] proposed to decompose the original label flip matrix to some predictable matrices to relax such a requirement; and [29] estimated the label flip matrix by resorting to the simplex formed by its columns with the minimum volume. Other representative works belonging to this type include [30], [31], [32], [33], [34].

Among label-flip-rate based loss correction approaches, an important branch of works such as [11], [12], [35] aim to correct the loss by recovering the *centroid* of the training dataset with clean labels, in which an unbiased centroid estimator is critically built to form the unbiased empirical risk. Concretely, [11], [12], [35], [36] reveal that the commonly used loss functions (*e.g.*, the squared loss and hinge loss) can be decomposed as a label-independent term plus a label-dependent term, among which only the latter is affected by the corrupted labels. By noting that the label-dependent term is governed by the centroid of the training dataset which is related to the label values, the only thing we need to do is to precisely estimate the data centroid based on the observed noisy training set. However, the above-mentioned methods consider the noisy labels in all classes simultaneously when devising the centroid estimator, so the local information of individual class is not fully deployed, which often leads to the degraded performance.

Therefore, in this paper, we propose a novel algorithm dubbed "Class-Wise Denoising" (CWD) to progressively tackle the noisy labels class by class so that the difficulty of entire denoising process can be decreased. Taking binary classification as an example, we respectively treat the positively labeled examples and negatively labeled examples in the training set as clean, and separately correct noisy labels in the negative class and positive class via a class-wise way. By this way, two virtual auxiliary sets are built with actually false positive and false negative examples. After this, two unbiased centroid estimators based on these two virtual auxiliary sets can be accordingly obtained which are further combined in an appropriate way to form a final integrated estimator. Theoretically, we prove that: 1) the variance of centroid estimator involved in our CWD is often lower than that of the unbiased centroid estimator in existing methodologies, which means that the estimator of CWD is statistically more efficient than existing methods; and 2) the performance of our method trained on the noisy set will converge to that of the optimal classifier trained on the clean set, and the convergence rate is $\mathcal{O}(\frac{1}{\sqrt{n}})$ with n being the number of training examples. Experimentally, we show that CWD yields higher classification accuracy than existing typical label-noise learning algorithms on a variety of benchmark and real-world datasets under various noise types.

In fact, Lee *et al.* [37] also proposed to tackle the label noise in different classes separately. However, their work was based on an empirical observation that the hidden representations generated by neural networks exhibit clustering property, and the training examples with noisy labels were distributed like outliers. Therefore, they estimated the distribution parameters of each cluster by using minimum covariance determinant, which is very different from our strategy that aims to devise an unbiased centroid estimator. Besides, one recent work [12] also introduced two virtual auxiliary sets to achieve unbiased and statistically efficient centroid estimation. However, one of the virtual auxiliary sets in [12] pre-labels all examples as positive, which may incur the class imbalance problem and lead to the degraded learning results. Differently, CWD developed in this paper carefully builds two virtual auxiliary sets without introducing subjective pre-assumed labels, so the disadvantage of [12] can be avoided. The advantage of CWD over [12] is also observed in empirical studies (see Section 6).

The main contributions of this paper are summarized as follows:

1) We propose a new "Class-Wise Denoising" (CWD) algorithm to tackle label noise, which favors to sequentially cleanse the incorrect labels within different classes by establishing corresponding virtual auxiliary sets.
2) We theoretically show that the risk estimator induced by the proposed CWD is not only unbiased, but is also statistically more efficient than the existing methods based on centroid estimation.
3) We theoretically prove that the performance of our CWD classifier trained on the noisy set will get close to that of the optimal classifier trained on the corresponding clean set, as long as the amount of noisily-labeled training examples increases.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2022.3178690, IEEE Transactions on Pattern Analysis and Machine Intelligence

TABLE 1: The decomposition of some common loss functions [12], [35], [36], where $z = yh(\mathbf{x})$ is the functional margin, and $[\cdot]_+ = \max(\cdot, 0)$. The decomposition of the hinge loss is actually conducted on its upper bound as revealed by [36].

loss	$\ell(z)$	label-independent term	label-dependent term	g	Q
squared loss	$(z-1)^2$	z^2+1	$-2z$	h^2+1	-2
logistic loss	$\log(1+e^{-z})$	$\frac{1}{2}\log(2+e^{z}+e^{-z})$	$-z/2$	$\frac{1}{2}\log(2+e^{h}+e^{-h})$	$-1/2$
perceptron loss	$\max(0,-z)$	$\frac{1}{2}z\cdot\mathrm{sgn}(z\geq 0)$	$-z/2$	$\frac{1}{2}h\cdot\mathrm{sgn}[h\geq 0]$	$-1/2$
hinge loss	$[1-z]_+$	$\frac{1}{2}([1-z]_+ + [1+z]_+)$	$\frac{1}{2}(1-z)$	$\frac{1}{2}([1-h]_+ + [1+h]_+)$	$-1/2$

TABLE 2: Summary of main mathematical notations.

Notation	Mathematical meaning
(X,Y), $(X,\widetilde{Y})$	A pair of input random variables (X,Y) and the observed contaminated counterpart $(X,\widetilde{Y})$.
$S=\{(\mathbf{x}_i,y_i)\}_{i=1}^n$	The unobserved clean sample S with n data points $(\mathbf{x}_i,y_i)$.
$\widetilde{S}=\{(\mathbf{x}_i,\tilde{y}_i)\}_{i=1}^n$	The observed noisy sample $\widetilde{S}$ with n possible mislabeled training data $(\mathbf{x}_i,\tilde{y}_i)$.
$S_{\widetilde{\mathrm{P}}}$, $S_{\widetilde{\mathrm{N}}}$	The two introduced virtual auxiliary pseudo-labeled sets with actually false positive examples and false negative examples, respectively.
$\hat{\mu}(S)$, $\hat{\mu}(\widetilde{S})$, $\hat{\mu}(S_{\widetilde{\mathrm{P}}})$, $\hat{\mu}(S_{\widetilde{\mathrm{N}}})$	Empirical centroids of samples S, $\widetilde{S}$, $S_{\widetilde{\mathrm{P}}}$, and $S_{\widetilde{\mathrm{N}}}$, respectively.
$\tilde{\hat{\mu}}(S)$, $\tilde{\hat{\mu}}(S_{\widetilde{\mathrm{P}}})$, $\tilde{\hat{\mu}}(S_{\widetilde{\mathrm{N}}})$	Estimators of $\hat{\mu}(S)$, $\hat{\mu}(S_{\widetilde{\mathrm{P}}})$, and $\hat{\mu}(S_{\widetilde{\mathrm{N}}})$, respectively.
$\hat{\mathcal{R}}(h,S)$, $\widetilde{\mathcal{R}}(h,\widetilde{S})$	Empirical risks of hypothesis h on clean sample S and noisy sample $\widetilde{S}$, respectively.
$\widetilde{\hat{\mathcal{R}}}(h,\widetilde{S})$	Estimator of $\hat{\mathcal{R}}(h,S)$ based on the noisy sample $\widetilde{S}$.
η_{P}, η_{N}	Label flip rates.
π_{P}, π_{N}	Class priors for positive and negative classes, respectively.

2 Empirical Risk under Noisy Labels

In our paper, the superscript "$\tilde{}$" means that the variable is estimated or noisy, and the variable with superscript "$\hat{}$" means that it is an empirical quantity. The main notations that will be later used for algorithm description are listed in Table 2. In traditional supervised learning with clean labels, we let D be the underlying joint distribution of a pair of random variables $(X,Y)\in\mathcal{X}\times\mathcal{Y}$, where $\mathcal{X}\in\mathbb{R}^d$ (d denotes the dimensionality) is the input feature space and $\mathcal{Y}=\{1,-1\}$[1] is the output label space. In this case, a sample set $S=\{(\mathbf{x}_i,y_i)\}_{i=1}^n$ of (X,Y) containing n examples can be drawn independently and identically from D, where all $\{y_i\}_{i=1}^n$ are correct. However, in the task of classification under noisy labels, we are only accessible to a sample of n i.i.d. data points $\widetilde{S}=\{(\mathbf{x}_i,\tilde{y}_i)\}_{i=1}^n$ from a noisy distribution $\widetilde{D}$ of random variables $(X,\widetilde{Y})\in\mathcal{X}\times\mathcal{Y}$, where $\widetilde{Y}$ is a contaminated version of Y. Therefore, given the hypothesis space as $\mathcal{H}$, our task is to find a suitable decision function $h\in\mathcal{H}:\mathcal{X}\to\mathcal{Y}$ parameterized by $\mathbf{w}$ on $\widetilde{S}$, such that h can precisely predict the label Y of any $X\in\mathcal{X}$.

By defining $\ell:\mathbb{R}\times\mathcal{Y}\to\mathbb{R}$ as the loss function that penalizes the difference between the model output $h(X)$ and the groundtruth label Y, the empirical risk of h on a clean set S for traditional supervised learning is represented as

$$\hat{\mathcal{R}}(h,S)=\frac{1}{n}\sum_{i=1}^{n}\ell(h(\mathbf{x}_i),y_i). \tag{1}$$

Similarly to Eq. (1), due to the corruption of y to $\widetilde{y}$ in the presence of noisy labels, the empirical risk of any h on a noisy set $\widetilde{S}$ is written as

$$\widetilde{\mathcal{R}}(h,\widetilde{S})=\frac{1}{n}\sum_{i=1}^{n}\ell(h(\mathbf{x}_i),\widetilde{y}_i), \tag{2}$$

where $\widetilde{\mathcal{R}}(h,\widetilde{S})$ may be deviated from the real $\hat{\mathcal{R}}(h,S)$ because of the unavailability of groundtruth labels $\{y_i\}_{i=1}^n$.

Ideally, we hope to find an unbiased estimator $\widetilde{\hat{\mathcal{R}}}(h,\widetilde{S})$ for $\hat{\mathcal{R}}(h,S)$ given $\widetilde{S}$ so that the adverse impact caused by noisy $\widetilde{Y}_i$ can be removed. Following this idea, [11], [35] proposed to decompose the loss function ℓ (*e.g.*, the squared loss and logistic loss) into a label-independent part and a label-dependent part (see Table 1), where only the label-dependent part is influenced by label noise and needs further investigation. They showed that under loss decomposition and $h(\mathbf{x};\mathbf{w})=\langle\mathbf{w},\mathbf{x}\rangle^2$, Eq. (1) can be reformulated as

$$\begin{aligned}\hat{\mathcal{R}}(h,S)&=\frac{1}{n}\left[\sum_{i=1}^{n}g(h(\mathbf{x}_i;\mathbf{w}))+Q\sum_{i=1}^{n}y_ih(\mathbf{x}_i;\mathbf{w})\right]\\&=\frac{1}{n}\sum_{i=1}^{n}g(h(\mathbf{x}_i;\mathbf{w}))+Q\langle\mathbf{w},\hat{\mu}(S)\rangle,\end{aligned} \tag{3}$$

where $g:\mathbb{R}\to\mathbb{R}$ is some L_g-lipschitz continuous function, $Q\in\mathbb{R}$ is a constant, and $\hat{\mu}(S)=\frac{1}{n}\sum_{i=1}^{n}y_i\mathbf{x}_i$ is the empirical dataset centroid of S. Here the specific forms of g and Q depend on the adopted loss function as revealed in Table 1. Moreover, we may define the centroid of the entire distribution D as $\mu(D)=\mathbb{E}_{(X,Y)\sim D}[YX]$ with $\mathbb{E}[\cdot]$ computing the mathematical expectation.

From Eq. (3), we see that only the second term is related to the label value y_i. Consequently, if we want to find an unbiased $\widetilde{\hat{\mathcal{R}}}(h,\widetilde{S})$ to $\hat{\mathcal{R}}(h,S)$ to combat noisy labels, the core is to accurately estimate the dataset centroid $\hat{\mu}(S)$ based on $\widetilde{S}$, and the resulting estimator for $\hat{\mu}(S)$ is denoted by $\tilde{\hat{\mu}}(S)$ accordingly. Consequently, we have an unbiased empirical risk under label noise as

$$\widetilde{\hat{\mathcal{R}}}(h,\widetilde{S})=\frac{1}{n}\sum_{i=1}^{n}g(h(\mathbf{x}_i;\mathbf{w}))+Q\langle\mathbf{w},\tilde{\hat{\mu}}(S)\rangle. \tag{4}$$

The model of CWD in this paper can finally be achieved by combining the risk Eq. (4) with some techniques for preventing overfitting, such as the ℓ_2 regularizer for linear models, and the dropout operation [38] for neural networks.

1. For notational simplicity, we first describe our method under binary case. The extension to multi-class situations will be provided in Section 5.

2. In this paper, h, $h(\mathbf{w})$ and $h(\mathbf{x};\mathbf{w})$ refer to the same thing. We use their different forms in different places to avoid possible confusion as well as to simplify the notation.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2022.3178690, IEEE Transactions on Pattern Analysis and Machine Intelligence

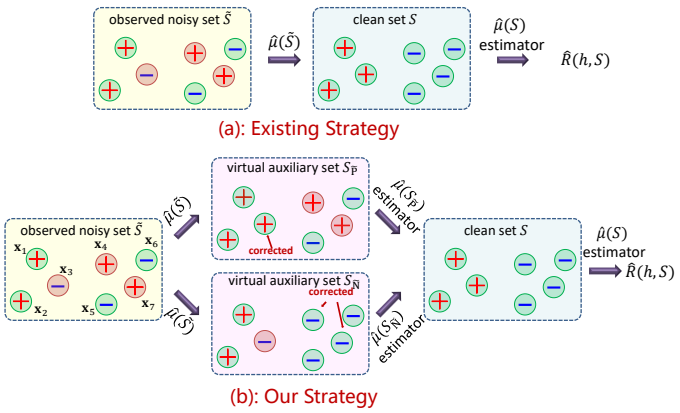

Fig. 1: The pipeline comparison of (a) existing methods and (b) our method. The existing strategy directly corrects all noisy labels in both the positive and negative classes contained in the observed noisy dataset $\widetilde{S}$. In contrast, our strategy introduces two virtual auxiliary sets $S_{\widetilde{\mathrm{P}}}$ and $S_{\widetilde{\mathrm{N}}}$ as a bridge from noisy $\widetilde{S}$ to clean S, and they help to correct the labels of false negative and false positive data points in $\widetilde{S}$, respectively.

3 The Proposed Centroid Estimator

As mentioned earlier, given the observed noisy dataset $\widetilde{S}$ as illustrated in Fig. 1(a), most of existing methods aim to cleanse the noisy labels in all classes simultaneously, and directly achieve clean S which induces the real empirical risk $\hat{\mathcal{R}}(h,S)$. In this case, the class-conditional label flip rates between the positive class and negative class can be defined as $\eta_{\mathrm{P}} = P(\widetilde{Y}=-1|Y=1)$ and $\eta_{\mathrm{N}} = P(\widetilde{Y}=1|Y=-1)$, where $P(\cdot)$ denotes the probability in this paper. The class priors in D are defined by $\pi_{\mathrm{P}} = P(Y=1)$ and $\pi_{\mathrm{N}} = P(Y=-1)$ with $\pi_{\mathrm{P}}+\pi_{\mathrm{N}}=1$. Here π_{P} can be estimated from η_{P} and η_{N} due to the following derivations:

$$\begin{aligned} &P(\widetilde{Y}=1)\\ &=P(\widetilde{Y}=1|Y=1)P(Y=1)+P(\widetilde{Y}=1|Y=-1)P(Y=-1)\\ &=\eta_{\mathrm{N}}+(1-\eta_{\mathrm{P}}-\eta_{\mathrm{N}})\pi_{\mathrm{P}}, \end{aligned} \tag{5}$$

which leads to $\pi_{\mathrm{P}} = \frac{P(\widetilde{Y}=1)-\eta_{\mathrm{N}}}{1-\eta_{\mathrm{P}}-\eta_{\mathrm{N}}}$ where $P(\widetilde{Y}=1)$ can be estimated from the given noisy dataset. By following [9], [11], [12], in this paper, we also assume that η_{P} and η_{N} are known. Practically, they can be easily estimated by some off-the-shelf methods such as [26], [27], [28], [29], [31].

In contrast, our CWD proposes to deal with the label noise in different classes one by one in order to acquire the improved estimation for the actual empirical risk $\hat{R}(h,S)$ (see Fig. 1(b)). To be specific, starting from the observed noisy $\widetilde{S}$, we respectively regard that the positively labeled data and negatively labeled data in $\widetilde{S}$ are correctly labeled, and arrive at two virtual auxiliary sets $S_{\widetilde{\mathrm{P}}}$ and $S_{\widetilde{\mathrm{N}}}$ accordingly. Note that due to label noise, not all positively labeled data in $\widetilde{S}$ are correctly labeled actually, therefore $S_{\widetilde{\mathrm{P}}}$ indeed contains false positive examples. Similarly, $S_{\widetilde{\mathrm{N}}}$ also contains false negative examples as some positive data are erroneously labeled as negative in $\widetilde{S}$. Here we say $S_{\widetilde{\mathrm{P}}}$ and $S_{\widetilde{\mathrm{N}}}$ are "virtual" as we do not explicitly construct these two datasets. Instead, they are simply fictitious with assumed labels and will not take additional storage space. Then we estimate their centroids $\hat{\mu}(S_{\widetilde{\mathrm{P}}}) = \frac{1}{n}\sum_{i=1}^{n}(y_{\widetilde{\mathrm{P}}})_i\mathbf{x}_i$ and $\hat{\mu}(S_{\widetilde{\mathrm{N}}}) = \frac{1}{n}\sum_{i=1}^{n}(y_{\widetilde{\mathrm{N}}})_i\mathbf{x}_i$ based on $\hat{\mu}(\widetilde{S}) = \frac{1}{n}\sum_{i=1}^{n}\widetilde{y}_i\mathbf{x}_i$, where $\hat{\mu}(\widetilde{S})$ is the centroid of $\widetilde{S}$; and $y_{\widetilde{\mathrm{P}}}$, $y_{\widetilde{\mathrm{N}}}$ and $\widetilde{y}$ are the (assumed) labels of examples in $S_{\widetilde{\mathrm{P}}}$, $S_{\widetilde{\mathrm{N}}}$ and $\widetilde{S}$ correspondingly. After that, the estimated $\hat{\mu}(S_{\widetilde{\mathrm{P}}})$ and $\hat{\mu}(S_{\widetilde{\mathrm{N}}})$ are further combined to acquire the centroid of clean S (*i.e.*, $\hat{\mu}(S)$) for recovering $\hat{R}(h,S)$. In this process, the two intermediate virtual auxiliary sets $S_{\widetilde{\mathrm{P}}}$ and $S_{\widetilde{\mathrm{N}}}$ serve as a bridge from the centroid of noisy $\widetilde{S}$ to that of clean S, which help to progressively remedy the incorrect labels in the positive class and negative class inherited by the observed $\widetilde{S}$. It can be proved[3] that for any $S_{\widetilde{\mathrm{P}}}$, $S_{\widetilde{\mathrm{N}}}$ and $\widetilde{S}$ degenerated from S, their centroids have the following relationship:

$$\hat{\mu}(S) = \hat{\mu}(S_{\widetilde{\mathrm{P}}}) + \hat{\mu}(S_{\widetilde{\mathrm{N}}}) - \hat{\mu}(\widetilde{S}), \tag{6}$$

where $\hat{\mu}(\widetilde{S}) = \frac{1}{n}\sum_{i=1}^{n}\widetilde{y}_i\mathbf{x}_i$ is directly computable as $\widetilde{S}$ is available. Therefore, to find $\tilde{\hat{\mu}}(S)$ that is the estimator of $\hat{\mu}(S)$, the core is to find estimators of $\hat{\mu}(S_{\widetilde{\mathrm{P}}})$ and $\hat{\mu}(S_{\widetilde{\mathrm{N}}})$, which are subsequently denoted by $\tilde{\hat{\mu}}(S_{\widetilde{\mathrm{P}}})$ and $\tilde{\hat{\mu}}(S_{\widetilde{\mathrm{N}}})$, respectively.

Firstly, we presume that all positive examples annotated in $\widetilde{S}$ are indeed positive, and only pay attention to correct the noisy labels in negatively labeled data. That is to say, we treat the virtual auxiliary set $S_{\widetilde{\mathrm{P}}}$ in Fig. 1(b) as clean and only the labels of false-negative data in $\widetilde{S}$ (*e.g.*, $\mathbf{x}_3$) are expected to be corrected. Therefore, we know that the positive class prior

$$\begin{aligned} &\pi_{\widetilde{\mathrm{P}}} = P(Y_{\widetilde{\mathrm{P}}}=1)\\ &=P(Y=1)P(Y_{\widetilde{\mathrm{P}}}=1|Y=1)+P(Y=-1)P(Y_{\widetilde{\mathrm{P}}}=1|Y=-1)\\ &=\pi_{\mathrm{P}}+\pi_{\mathrm{N}}\eta_{\mathrm{N}}, \end{aligned} \tag{7}$$

where $P(Y_{\widetilde{\mathrm{P}}}=1|Y=-1) = P(\widetilde{Y}=1|Y=-1) = \eta_{\mathrm{N}}$. As a result, the label flip rates from $S_{\widetilde{\mathrm{P}}}$ to $\widetilde{S}$ are

$$P(\widetilde{Y}=1|Y_{\widetilde{\mathrm{P}}}=-1) = \frac{P(\widetilde{Y}=1, Y_{\widetilde{\mathrm{P}}}=-1)}{P(Y_{\widetilde{\mathrm{P}}}=-1)} = 0 \tag{8}$$

$$\begin{aligned} P(\widetilde{Y}=-1|Y_{\widetilde{\mathrm{P}}}=1) &= \frac{P(\widetilde{Y}=-1, Y_{\widetilde{\mathrm{P}}}=1)}{P(Y_{\widetilde{\mathrm{P}}}=1)}\\ &= \frac{\pi_{\mathrm{P}}\eta_{\mathrm{P}}}{\pi_{\mathrm{P}}+\pi_{\mathrm{N}}\eta_{\mathrm{N}}} \triangleq \eta'_{\mathrm{P}}. \end{aligned} \tag{9}$$

Therefore, by considering Eqs. (8) and (9), we have

$$\begin{aligned} &\mathbb{E}_{\widetilde{Y}}[\widetilde{Y}X|(X,Y_{\widetilde{\mathrm{P}}})]\\ &=\pi_{\widetilde{\mathrm{P}}}\mathbb{E}_{\widetilde{Y}}[\widetilde{Y}X|(X,Y_{\widetilde{\mathrm{P}}}=1)]\\ &\quad+(1-\pi_{\widetilde{\mathrm{P}}})\mathbb{E}_{\widetilde{Y}}[\widetilde{Y}X|(X,Y_{\widetilde{\mathrm{P}}}=-1)]\\ &=\pi_{\widetilde{\mathrm{P}}}(1-2\eta'_{\mathrm{P}})Y_{\widetilde{\mathrm{P}}}X+(1-\pi_{\widetilde{\mathrm{P}}})Y_{\widetilde{\mathrm{P}}}X\\ &=(1-2\pi_{\widetilde{\mathrm{P}}}\eta'_{\mathrm{P}})Y_{\widetilde{\mathrm{P}}}X, \end{aligned} \tag{10}$$

which indicates that an unbiased estimate of $\hat{\mu}(S_{\widetilde{\mathrm{P}}})$ using $\widetilde{S}$ is $\tilde{\hat{\mu}}(S_{\widetilde{\mathrm{P}}}) = \frac{1}{1-2\pi_{\widetilde{\mathrm{P}}}\eta'_{\mathrm{P}}}\hat{\mu}(\widetilde{S})$.

Secondly, we presume that all negative examples annotated in $\widetilde{S}$ are indeed negative, and only focus on correcting the noisy labels in positively labeled data. In other words, we treat the virtual auxiliary set $S_{\widetilde{\mathrm{N}}}$ in Fig. 1(b) as clean and the labels of false-positive data in $\widetilde{S}$ (*e.g.*, $\mathbf{x}_4$ and $\mathbf{x}_7$) are expected to be corrected. Thereby, we have the positive class prior in $S_{\widetilde{\mathrm{N}}}$ as

$$\begin{aligned} &\pi_{\widetilde{\mathrm{N}}} = P(Y_{\widetilde{\mathrm{N}}}=1)\\ &=P(Y=1)P(Y_{\widetilde{\mathrm{N}}}=1|Y=1)+P(Y=-1)P(Y_{\widetilde{\mathrm{N}}}=1|Y=-1)\\ &=\pi_{\mathrm{P}}(1-\eta_{\mathrm{P}}), \end{aligned} \tag{11}$$

3. The detailed proof is deferred to **supplementary material**.

Algorithm 1 Summarization of our CWD algorithm for binary classification.

1: **Input:** label flip rates η_{P}, η_{N}; noisy set $\widetilde{S} = \{(\mathbf{x}_i, \widetilde{y}_i)\}_{i=1}^n$.
2: Compute $\pi_{\mathrm{P}} = \frac{P(\widetilde{Y}=1)-\eta_{\mathrm{N}}}{1-\eta_{\mathrm{P}}-\eta_{\mathrm{N}}}$ and $\pi_{\mathrm{N}} = 1-\pi_{\mathrm{P}}$;
3: Compute η'_{P} and η'_{N} via Eqs. (9) and (12), respectively;
4: Compute $\pi_{\widetilde{\mathrm{P}}} = \pi_{\mathrm{P}} + \pi_{\mathrm{N}}\eta_{\mathrm{N}}$ and $\pi_{\widetilde{\mathrm{N}}} = \pi_{\mathrm{P}}(1-\eta_{\mathrm{P}})$;
5: Compute $\hat{\mu}(\widetilde{S})$, $\widetilde{\hat{\mu}}(S_{\widetilde{\mathrm{P}}})$, and $\widetilde{\hat{\mu}}(S_{\widetilde{\mathrm{N}}})$;
6: Compute the estimated centroid of S via $\tilde{\hat{\mu}}(S) = \widetilde{\hat{\mu}}(S_{\widetilde{\mathrm{P}}}) + \widetilde{\hat{\mu}}(S_{\widetilde{\mathrm{N}}}) - \hat{\mu}(\widetilde{S})$;
7: Compute the unbiased risk estimator $\widetilde{\hat{\mathcal{R}}}(h, \widetilde{S})$ via Eq. (4);
8: Use any off-the-shelf solver to optimize the model (*e.g.*, SVM and CNN) by employing $\widetilde{\hat{\mathcal{R}}}(h, \widetilde{S})$ as the loss function.
9: **Output:** The optimal classifier parameter $\mathbf{w}^*$.

where $P(Y_{\widetilde{\mathrm{N}}} = 1|Y=-1) = P(\widetilde{Y} = -1|Y=1) = \eta_{\mathrm{P}}$. Consequently, the label flip rates from $S_{\widetilde{N}}$ to $\widetilde{S}$ are

$$P(\widetilde{Y}=1|Y_{\widetilde{\mathrm{N}}}=-1) = \frac{P(\widetilde{Y}=1, Y_{\widetilde{\mathrm{N}}}=-1)}{P(Y_{\widetilde{\mathrm{N}}}=-1)} = \frac{(1-\pi_{\mathrm{P}})\eta_{\mathrm{N}}}{1-\pi_{\mathrm{P}}+\pi_{\mathrm{P}}\eta_{\mathrm{P}}} \triangleq \eta'_{\mathrm{N}}, \tag{12}$$

$$P(\widetilde{Y}=-1|Y_{\widetilde{\mathrm{N}}}=1) = \frac{P(\widetilde{Y}=-1, Y_{\widetilde{\mathrm{N}}}=1)}{P(Y_{\widetilde{\mathrm{N}}}=1)} = 0. \tag{13}$$

By invoking Eqs. (12) and (13), we have

$$\begin{aligned}
&\mathbb{E}_{\widetilde{Y}}[\widetilde{Y}X|(X, Y_{\widetilde{\mathrm{N}}})] \\
&= \pi_{\widetilde{\mathrm{N}}}\mathbb{E}_{\widetilde{Y}}[\widetilde{Y}X|(X, Y_{\widetilde{\mathrm{N}}}=1)] \\
&\quad + (1-\pi_{\widetilde{\mathrm{N}}})\mathbb{E}_{\widetilde{Y}}[\widetilde{Y}X|(X, Y_{\widetilde{\mathrm{N}}}=-1)] \\
&= \pi_{\widetilde{\mathrm{N}}}Y_{\widetilde{\mathrm{N}}}X + (1-\pi_{\widetilde{\mathrm{N}}})(1-2\eta'_{\mathrm{N}}) \\
&= [1-2(1-\pi_{\widetilde{\mathrm{N}}})\eta'_{\mathrm{N}}]Y_{\widetilde{\mathrm{N}}}X,
\end{aligned} \tag{14}$$

which indicates that an unbiased estimate of $\hat{\mu}(S_{\widetilde{\mathrm{N}}})$ based on $\widetilde{S}$ is $\widetilde{\hat{\mu}}(S_{\widetilde{\mathrm{N}}}) = \frac{1}{1-2(1-\pi_{\widetilde{\mathrm{N}}})\eta'_{\mathrm{N}}}\hat{\mu}(\widetilde{S})$.

Therefore, by recalling Eq. (6), we learn that the centroid of S can be unbiasedly estimated as

$$\begin{aligned}
\tilde{\hat{\mu}}(S) &= \widetilde{\hat{\mu}}(S_{\widetilde{\mathrm{P}}}) + \widetilde{\hat{\mu}}(S_{\widetilde{\mathrm{N}}}) - \hat{\mu}(\widetilde{S}) \\
&= \left(\frac{1}{1-2\pi_{\widetilde{\mathrm{P}}}\eta'_{\mathrm{P}}} + \frac{1}{1-2(1-\pi_{\widetilde{\mathrm{N}}})\eta'_{\mathrm{N}}} - 1\right)\hat{\mu}(\widetilde{S}) \\
&= \left(\frac{1}{1-2\pi_{\mathrm{P}}\eta_{\mathrm{P}}} + \frac{1}{1-2\pi_{\mathrm{N}}\eta_{\mathrm{N}}} - 1\right)\hat{\mu}(\widetilde{S}),
\end{aligned} \tag{15}$$

which leads to an unbiased empirical risk estimator $\widetilde{\hat{R}}(h, \widetilde{S})$ by substituting Eq. (15) to Eq. (4). From Eq. (15), we see that if the training set is noise-free, namely $\eta_{\mathrm{P}} = \eta_{\mathrm{N}} = 0$ and $\widetilde{S} = S$, our proposed CWD model will directly degenerate to the traditional supervised model. Therefore, even we do not know whether the training set is clean before running our algorithm, our method can be safely used and the generated performance will not become too bad. The pseudo-code of our developed CWD algorithm is displayed in Algorithm 1.

4 Theoretical Analyses

In this section, we investigate theoretical aspects of the proposed CWD algorithm. To be specific, Section 4.1 reveals that our estimator is often statistically more efficient than existing methods, and Section 4.2 demonstrates that the expected risk of our CWD trained on noisy set $\widetilde{S}$ is upper-bounded under the clean distribution D.

4.1 Statistical Efficiency

Here we theoretically study the superiority of our proposed CWD to existing methods with unbiased centroid estimator in terms of statistical efficiency.

The centroid estimator proposed by [11] is[4]

$$\tilde{\hat{\mu}}_0(S) = \frac{1}{1-2\pi_{\mathrm{P}}\eta_{\mathrm{P}}-2\pi_{\mathrm{N}}\eta_{\mathrm{N}}}\hat{\mu}(\widetilde{S}). \tag{16}$$

Therefore, by respectively denoting the covariance matrices of $\tilde{\hat{\mu}}(S)$ and $\tilde{\hat{\mu}}_0(S)$ as $\Sigma[\tilde{\hat{\mu}}(S)]$ and $\Sigma[\tilde{\hat{\mu}}_0(S)]$, our target is to compare the values of $\mathrm{tr}(\Sigma[\tilde{\hat{\mu}}(S)])$ and $\mathrm{tr}(\Sigma[\tilde{\hat{\mu}}_0(S)])$ where "$\mathrm{tr}(\cdot)$" is the trace operator. The result is displayed in the following theorem:

Theorem 1. *Given the centroid estimators $\tilde{\hat{\mu}}(S)$ and $\tilde{\hat{\mu}}_0(S)$ respectively computed by Eq.* (15) *and Eq.* (16)*, we have their variances $\mathrm{tr}(\Sigma[\tilde{\hat{\mu}}(S)]) \leq \mathrm{tr}(\Sigma[\tilde{\hat{\mu}}_0(S)])$ with probability $\ln 2$ (≈ 0.693).*

Theorem 1 is proved in the **supplementary material**, from which we know that in most case with the probability $\ln 2 \approx 0.693$, our method is statistically equally or more efficient than [11].

4.2 Performance Bound

In this section, we show that although our classifier $\mathbf{w}^*$ is trained on the noisy set $\widetilde{S}$, its expected error on the clean distribution D can still be upper-bounded when compared with the optimal classifier $\mathbf{w}^{**}$ trained on the corresponding clean set S.

By respectively defining the expected risk of any h on $\widetilde{D}$ and D as $\mathcal{R}(h, \widetilde{D}) = \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{D}}\left[\ell(h(X;\mathbf{w}), \widetilde{Y})\right]$ and $\mathcal{R}(h, D) = \mathbb{E}_{(X,Y)\sim D}\left[\ell(h(X;\mathbf{w}), Y)\right]$, the expected classifier rendered by our CWD algorithm and the one trained on the clean set can be obtained by $\mathbf{w}^* = \arg\min_{\mathbf{w}} \mathcal{R}(h, \widetilde{D})$ and $\mathbf{w}^{**} = \arg\min_{\mathbf{w}} \mathcal{R}(h, D)$ accordingly, where h in this paper is a Multi-Layer Perceptron (MLP) network parameterized by $\mathbf{w}$.

Therefore, we may assume that the model h of CWD is consisted of l layers with parameter matrices $\mathbf{w}^{(1)}, \cdots, \mathbf{w}^{(l)}$ and activation functions $\sigma^{(1)}, \cdots, \sigma^{(l-1)}$ with $\sigma^{(i)}(\mathbf{0}) = 0$ for $i = 1, \cdots, l-1$. In this paper, we use ReLU as the activation function $\sigma^{(i)}$ for $i = 1, \cdots, l-1$. The size of the i-th parameter matrix $\mathbf{w}^{(i)}$ is $m^{(i)} \times m^{(i+1)}$ where $m^{(i)}$ denotes the number of the nodes in the i-th layer, and $m^{(1)} = d$. Then $\mathbf{w}^{(i)}_{jk}$, namely the (j,k)-th element of $\mathbf{w}^{(i)}$, represents the connecting weight from the j-th node of the i-th layer to the k-th node of the $(i+1)$-th layer. As a result, we have $h(\mathbf{x};\mathbf{w}) = \mathbf{w}^{(l)\top}\sigma^{(l-1)}(\mathbf{w}^{(l-1)\top}\sigma^{(l-2)}(\cdots\sigma^{(1)}(\mathbf{w}^{(1)\top}\mathbf{x})))$ as the real-valued network output. For simplicity, we compactly denote the network parameters as $\mathbf{w} = \{\mathbf{w}^{(1)}, \cdots, \mathbf{w}^{(l)}\}$ in this paper.

Based on the above facts, we may have the following main theorem:

4. The work [11] studied the case of $\eta_{\mathrm{P}} = \eta_{\mathrm{N}}$, and here we show the extended expression for both $\eta_{\mathrm{P}} = \eta_{\mathrm{N}}$ and $\eta_{\mathrm{P}} \neq \eta_{\mathrm{N}}$, which can be similarly derived via the method in [11].

Theorem 2. *(Performance bound of CWD) Suppose the backbone network for implementing CWD is MLP which has totally l layers with the parameter matrices $\mathbf{w}^{(i)}$ in the i-th ($i = 1, 2, \cdots, l$) layers satisfying $\left\|\mathbf{w}^{(i)}\right\|_{\mathrm{F}} \leq M^{(i)} < +\infty$, where $\|\cdot\|_{\mathrm{F}}$ denotes Frobenius norm. In MLP, the adopted activation function $\sigma(\cdot)$ is ReLU which is 1-Lipschitz. Besides, the ℓ_2 norm of the input feature vector $\mathbf{x} \in \mathbb{R}^d$ is bounded by $\bar{X}$, namely $\|\mathbf{x}\|_2 \leq \bar{X} < +\infty$. The function $g(\cdot)$ in Eq. (3) is L_g-Lipschitz continuous. Then for any $\delta > 0$, with probability at least $1 - 2\delta$, we have*

$$\begin{aligned}&\mathcal{R}(h(\mathbf{w}^*), D) - \mathcal{R}(h(\mathbf{w}^{**}), D)\\ &\leq \frac{2\bar{X}}{\sqrt{n}} \prod_{i=1}^{l} M^{(i)} \left(2\left|\Omega Q\right| \sqrt{2d \log\left(\frac{d}{\delta}\right)} + L_g(\sqrt{2l\log 2}+1)\right)\\ &\quad + 6\sqrt{\frac{\ln(2/\delta)}{2n}},\end{aligned} \tag{17}$$

where $\Omega = \frac{1}{1-2\pi_{\mathrm{P}}\eta_{\mathrm{P}}} + \frac{1}{1-2\pi_{\mathrm{N}}\eta_{\mathrm{N}}} - 1$.

The proof of this theorem can be found in the **supplementary material**. Theorem 2 indicates that although $h(\mathbf{w}^*)$ is trained on the noisy distribution $\widetilde{D}$, its performance will converge to that of the optimal classifier $h(\mathbf{w}^{**})$ which is trained on the clean distribution D, as long as the number of training data n increases, and the convergence rate is $\mathcal{O}(1/\sqrt{n})$.

5 Extension to Multi-Class Situations

The basic binary model developed in Section 3 can be directly extended to multi-class cases. Suppose there are C classes in total, the key is to respectively treat the observed labels of each of the C classes in $\widetilde{S}$ as clean, such that C virtual auxiliary sets $S_{\widetilde{1}}, \cdots, S_{\widetilde{C}}$ are built, and then the noisy labels in these C classes can be tackled via a class-wise way.

Suppose we have a noisy sample set $\widetilde{S} = \{(\mathbf{x}_i, \widetilde{\mathbf{y}}_i)\}_{i=1}^n$ where $\widetilde{\mathbf{y}}_i \in \{0,1\}^C$ is a C-dimensional label vector containing the one-hot encoding of class labels for $\mathbf{x}_i$. Concretely, by defining $\mathbf{e}_c$ as a C-dimensional vector with zero elements except for the c-th (c takes a value from $1, 2, \cdots, C$) element being 1, we have $\widetilde{\mathbf{y}}_i = \mathbf{e}_c$ if $\mathbf{x}_i$ has the observed noisy label c. The clean set is correspondingly denoted as $S = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^n$. By deploying the squared loss $\ell(h(\mathbf{x}_i), \widetilde{\mathbf{y}}_i) = \|\mathbf{y}_i - h(\mathbf{x}_i)\|^2$ where $h(\mathbf{x}_i) = \mathbf{W}^\top \mathbf{x}_i$ is decision function with $\mathbf{W} \in \mathbb{R}^{d\times C}$ being the coefficient matrix, we may follow Eq. (3) and decompose the fully-supervised empirical risk $\hat{\mathcal{R}}(h, S)$ as

$$\begin{aligned}\hat{\mathcal{R}}(h, S) &= \frac{1}{n}\sum_{i=1}^{n} \|\mathbf{y}_i - h(\mathbf{x}_i)\|^2\\ &= 1 + \frac{1}{n}\sum_{i=1}^{n} h^\top(\mathbf{x}) h(\mathbf{x}) - 2\langle \mathbf{W}, \hat{\mu}(S)\rangle,\end{aligned} \tag{18}$$

where we use the facts that $\mathbf{y}_i^\top \mathbf{y}_i = 1$ and $\mathbf{y}_i^\top h(\mathbf{x}_i) = \mathrm{tr}\left(h(\mathbf{x}_i)\mathbf{y}_i^\top\right) = \langle \mathbf{W}, \mathbf{x}_i \mathbf{y}_i^\top\rangle$; and $\hat{\mu}(S) = \frac{1}{n}\sum_{i=1}^n \mathbf{x}_i \mathbf{y}_i^\top$ is the centroid to be critically estimated based on the centroid of noisy set $\hat{\mu}(\widetilde{S}) = \frac{1}{n}\sum_{i=1}^n \mathbf{x}_i \widetilde{\mathbf{y}}_i^\top$.

To this end, we extend the previous label flip rates η_{P} and η_{N} to a label flip matrix $\boldsymbol{\eta}$ where the (i,j)-th element $\eta_{ij} = P(\widetilde{Y} = \mathbf{e}_j | Y = \mathbf{e}_i)$ encodes the label flip rate from the i-th class to the j-th class, therefore we have $\sum_{j=1}^C \eta_{ij} = 1$. Here the label flip matrix $\boldsymbol{\eta}$ can also be estimated by some existing works such as [26], [27], [28], [29], [31]. The class priors of C classes are respectively defined as $\pi_1 = P(Y = \mathbf{e}_1), \pi_2 = P(Y = \mathbf{e}_2), \cdots, \pi_C = P(Y = \mathbf{e}_C)$ with $\sum_{j=1}^C \pi_j = 1$ which can also be computed based on $\boldsymbol{\eta}$. Specifically, similarly to Eq. (5), we may have the following system of equations:

$$\begin{cases} P(\widetilde{Y} = \mathbf{e}_1) = \eta_{11}\pi_1 + \eta_{21}\pi_2 + \cdots + \eta_{C1}\pi_C\\ \qquad\qquad\vdots\\ P(\widetilde{Y} = \mathbf{e}_C) = \eta_{1C}\pi_1 + \eta_{2C}\pi_2 + \cdots + \eta_{CC}\pi_C \end{cases}, \tag{19}$$

from which the values of $\pi_1, \pi_2, \cdots, \pi_C$ can be easily solved.

After this, we need to build C virtual auxiliary sets where the c-th ($c = 1, \cdots, C$) virtual auxiliary set $S_{\widetilde{c}}$ is built by presuming that the examples with label c are all correctly annotated. Therefore, for a certain c-th virtual auxiliary set, its centroid is defined by $\hat{\mu}(S_{\widetilde{c}}) = \frac{1}{n}\sum_{i=1}^n \mathbf{x}_i (\mathbf{y}_{\widetilde{c}})_i^\top$ where $(\mathbf{y}_{\widetilde{c}})_i$ is the one-hot label vector of $\mathbf{x}_i$ in the set $S_{\widetilde{c}}$. Akin to Eq. (7), the class priors of the j-th ($j = 1, \cdots, C$) classes in $S_{\widetilde{c}}$ (denoted as $\pi_{\widetilde{c},j}$) can be calculated as

$$\begin{aligned}\pi_{\widetilde{c},j} &= P(Y_{\widetilde{c}} = \mathbf{e}_j)\\ &= P(Y = \mathbf{e}_1)P(Y_{\widetilde{c}} = \mathbf{e}_j | Y = \mathbf{e}_1) + \cdots\\ &+ P(Y = \mathbf{e}_c)P(Y_{\widetilde{c}} = \mathbf{e}_j | Y = \mathbf{e}_c) + \cdots\\ &+ P(Y = \mathbf{e}_C)P(Y_{\widetilde{c}} = \mathbf{e}_j | Y = \mathbf{e}_C),\end{aligned} \tag{20}$$

where $P(Y = \mathbf{e}_k) = \pi_k$ ($k = 1, \cdots, C$) as defined before. If $j = c$, Eq. (20) equals to

$$\pi_{\widetilde{c},j} = \pi_c + \sum\nolimits_{k=1,k\neq c}^{C} \pi_k \eta_{kj}, \tag{21}$$

where we use the facts that $P(Y_{\widetilde{c}} = \mathbf{e}_c | Y = \mathbf{e}_c) = 1$ and $P(Y_{\widetilde{c}} = \mathbf{e}_j | Y = \mathbf{e}_k) = \eta_{kj}$ under this situation.

If $j \neq c$, Eq. (20) equals to

$$\pi_{\widetilde{c},j} = \sum\nolimits_{k=1,k\neq c}^{C} \pi_k \eta_{kj}, \tag{22}$$

where we use the fact that $P(Y_{\widetilde{c}} = \mathbf{e}_j | Y = \mathbf{e}_c) = 0$ because we aim to correct all noisy labels in the c-th class at this time.

Next we need to establish a label flip matrix $\boldsymbol{\eta}'_{\widetilde{c}}$ from $S_{\widetilde{c}}$ to $\widetilde{S}$ encoding the label flip rates of pairs of classes, which acts as the same role with Eqs. (8) and (9) (or Eqs. (12) and (13)). For $j \neq c$, we have the label flip rates as

$$P(\widetilde{Y} = \mathbf{e}_c | Y_{\widetilde{c}} = \mathbf{e}_j) = \frac{P(\widetilde{Y} = \mathbf{e}_c, Y_{\widetilde{c}} = \mathbf{e}_j)}{P(Y_{\widetilde{c}} = \mathbf{e}_j)} = 0 \triangleq \eta'_{jc} \tag{23}$$

$$\begin{aligned}P(\widetilde{Y} = \mathbf{e}_j | Y_{\widetilde{c}} = \mathbf{e}_c) &= \frac{P(\widetilde{Y} = \mathbf{e}_j, Y_{\widetilde{c}} = \mathbf{e}_c)}{P(Y_{\widetilde{c}} = \mathbf{e}_c)}\\ &= \frac{\pi_c \eta_{cj}}{\pi_c + \sum_{j=1,j\neq c}^{C} \pi_j \eta_{jc}} \triangleq \eta'_{cj}\end{aligned} \tag{24}$$

$$P(\widetilde{Y} = \mathbf{e}_j | Y_{\widetilde{c}} = \mathbf{e}_j) = 1 \triangleq \eta'_{jj}. \tag{25}$$

Here Eq. (23) can be understood as label *flip-in* probability from other classes to the c-th class, while Eq. (24) can be understood as label *flip-out* probability from the c-th class to other classes. Based on Eq. (24), we further have

$$\begin{aligned}P(\widetilde{Y} = \mathbf{e}_c | Y_{\widetilde{c}} = \mathbf{e}_c) &= 1 - \sum\nolimits_{j=1,j\neq c}^{C} P(\widetilde{Y} = \mathbf{e}_j | Y_{\widetilde{c}} = \mathbf{e}_c)\\ &= 1 - \frac{\pi_c \sum_{j=1,j\neq c}^{C} \eta_{cj}}{\pi_c + \sum_{j=1,j\neq c}^{C} \pi_j \eta_{jc}} \triangleq \eta'_{cc}.\end{aligned} \tag{26}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2022.3178690, IEEE Transactions on Pattern Analysis and Machine Intelligence

Besides, for $j \neq i \neq c$, we have

$$P(\widetilde{Y} = \mathbf{e}_i | Y_{\widetilde{c}} = \mathbf{e}_j) = \frac{P(\widetilde{Y} = \mathbf{e}_i, Y_{\widetilde{c}} = \mathbf{e}_j)}{P(Y_{\widetilde{c}} = \mathbf{e}_j)} = 0 \triangleq \eta'_{ji}. \quad (27)$$

Based on the η'_{jc}, η'_{cj}, η'_{jj} and η'_{cc} defined in Eqs. (23), (24), (25) and (26) correspondingly, we obtain the label flip matrix $\boldsymbol{\eta}'_{\widetilde{c}}$ formatted as

$$\boldsymbol{\eta}'_{\widetilde{c}} = \begin{pmatrix} 1 & \cdots & \cdots & 0 & \cdots & \cdots & 0 \\ 0 & 1 & \cdots & 0 & \cdots & \cdots & 0 \\ \vdots & & & \vdots & & & \vdots \\ \cdots & \eta'_{cj} & \cdots & \eta'_{cc} & \cdots & \cdots & \cdots \\ \vdots & & & \vdots & & & \vdots \\ 0 & \cdots & \cdots & 0 & \cdots & 1 & 0 \\ 0 & \cdots & \cdots & 0 & \cdots & \cdots & 1 \end{pmatrix} \begin{matrix} \\ \\ \\ \leftarrow \text{the } c\text{-th row} \\ \\ \\ \\ \end{matrix}, \quad (28)$$

the c-th column $\downarrow$

Therefore, similarly to Eq. (14), we may compute the following conditional expectation based on the label flip matrix $\boldsymbol{\eta}'_{\widetilde{c}}$, namely

$$\begin{aligned} &\mathbb{E}_{\widetilde{Y}}[X\widetilde{Y}^\top | (X, Y_{\widetilde{c}})] \\ &= \sum_{j=1}^{C} P(Y_{\widetilde{c}} = \mathbf{e}_j) \mathbb{E}_{\widetilde{Y}}[X\widetilde{Y}^\top | (X, Y_{\widetilde{c}} = \mathbf{e}_j)] \\ &= \sum_{j=1}^{C} \pi_{\widetilde{c},j} [\eta'_{j1} X (\mathbf{K}_{j\to 1} Y_{\widetilde{c}})^\top + \cdots + \eta'_{jC} X (\mathbf{K}_{j\to C} Y_{\widetilde{c}})^\top] \\ &= \sum_{j=1}^{C} \pi_{\widetilde{c},j} \sum_{k=1}^{C} \eta'_{jk} X (\mathbf{K}_{j\to k} Y_{\widetilde{c}})^\top \\ &= X Y_{\widetilde{c}}^\top \sum_{j=1}^{C} \pi_{\widetilde{c},j} \sum_{k=1}^{C} \eta'_{jk} \mathbf{K}_{j\to k}^\top, \end{aligned} \quad (29)$$

where $\pi_{\widetilde{c},j}$ can be computed by Eq. (21) or Eq. (22) depending on whether $j = c$, and η'_{jk} is the (j,k)-th element of $\boldsymbol{\eta}'_{\widetilde{c}}$ in Eq. (28). Here $\mathbf{K}_{j\to k} \in \{0,1\}^{C\times C}$ is called *elementary row transformation matrix* which is formatted by exchanging the j-th row and the k-th row of an identity matrix, so that the i-th row and the j-th row of the column vector $Y_{\widetilde{c}}$ can be exchanged by computing $\mathbf{K}_{j\to k} Y_{\widetilde{c}}$. As a sequel, by letting $\mathbf{M}_c = \sum_{j=1}^{C} \pi_{\widetilde{c},j} \sum_{k=1}^{C} \eta'_{jk} \mathbf{K}_{j\to k}^\top$, we achieve the unbiased estimate of $\hat{\mu}(S_{\widetilde{c}})$ based on $\hat{\mu}(\widetilde{S})$ as $\tilde{\hat{\mu}}(S_{\widetilde{c}}) = \hat{\mu}(\widetilde{S}) \mathbf{M}_c^\dagger$ where $\mathbf{M}_c^\dagger$ computes the pseudo inverse of matrix $\mathbf{M}_c$. Thereby, similarly to Eq. (15), the estimator of $\hat{\mu}(S)$ (*i.e.*, $\tilde{\hat{\mu}}(S)$) based on $\tilde{\hat{\mu}}(S_{\widetilde{c}})$ ($c = 1, 2, \cdots, C$) is formulated as

$$\tilde{\hat{\mu}}(S) = \sum_{c=1}^{C} \tilde{\hat{\mu}}(S_{\widetilde{c}}) - (C-1)\hat{\mu}(\widetilde{S}). \quad (30)$$

The proof of Eq. (30) is similar to that of Eq. (6), which is also put into the **supplementary material**.

The pseudo code of our proposed CWD under multi-class case is presented in Algorithm 2.

Algorithm 2 Summarization of our CWD algorithm for multi-class classification.

Input: noisy training set $\widetilde{S} = \{(\mathbf{x}_i, \widetilde{\mathbf{y}}_i)\}_{i=1}^n$.
Estimate label flip matrix $\boldsymbol{\eta}$ via [29];
Compute class priors $\pi_1, \cdots, \pi_C$ by solving Eq. (19);
for $c = 1$ to C **do**
  Compute class priors $\pi_{\widetilde{c},j}$ ($j = 1, \cdots, C$) in the virtual auxiliary set $S_{\widetilde{c}}$ via Eqs. (21) and (22);
  Compute the label flip matrix $\boldsymbol{\eta}'_{\widetilde{c}}$ in Eq. (28) based on Eqs. (23), (24), (25) and (26);
  Compute $\mathbf{M}_c = \sum_{j=1}^{C} \pi_{\widetilde{c},j} \sum_{k=1}^{C} \eta'_{jk} \mathbf{K}_{j\to k}^\top$;
  Compute $\hat{\mu}(\widetilde{S}) = \hat{\mu}(\widetilde{S}) \mathbf{M}_c^\dagger$;
end for
Compute the estimated centroid of S via Eq. (30);
Compute the unbiased risk estimator of $\hat{\mathcal{R}}(h, S)$ via Eq. (18);
Use any off-the-shelf solver to optimize the model (*e.g.*, CNN) by employing $\hat{\mathcal{R}}(h, S)$ as the loss function.
Output: The optimal classifier parameter $\mathbf{W}^*$.

6 Experimental Results

In this section, we empirically investigate the performance of our proposed CWD method in dealing with noisy labels. Specifically, the compared methods include Unbiased Estimator (UE) [9], μSGD [35], Labeled Instance Centroid Smoothing (LICS) [11], Forward Correction (FC) [10], Determinant based Mutual Information ($\mathcal{L}_{\mathrm{DMI}}$) [39], Generalized Cross Entropy Loss (GCE) [23], Symmetric Cross Entropy (SCE) [8], f-Divergence (f-Div) [40], Sparse Regularization (SR) [41], and Centroid Estimation with Guaranteed Efficiency (CEGE) [12]. Among them, UE, μSGD, LICS and FC simply care about the unbiasedness of risk estimator, while CEGE also considers the statistical efficiency in addition to unbiasedness. Besides, $\mathcal{L}_{\mathrm{DMI}}$, GCE, SCE, f-Div and SR also try to design robust loss functions, so the comparison with them will validate the superiority of our proposed CWD.

TABLE 3: The characteristics of five adopted UCI datasets.

Dataset	$\bar{n}$	d	n_+	n_-
Heart	270	13	120	150
Blood	748	4	178	570
Diabetes	768	8	500	268
GermanCredit	1000	24	300	700
EEGEyeState	14980	16	8257	6723

In the following, we will test the classification ability of the compared approaches on five UCI benchmark datasets [42] (Section 6.1), two real-world binary classification datasets (Section 6.2), three real-world multi-class classification datasets (Section 6.3), and then study the effect of variance reduction brought by the consideration of statistical efficiency (Section 6.4).

6.1 Experiments on Benchmark Datasets

We first conduct the experiments on five benchmark datasets regarding binary classification from UCI machine learning repository [42], which include *Heart*, *BloodTransfusionServiceCenter* ("*Blood*" for short hereinafter), *Diabetes*, *GermanCredit*, and *EEGEyeState*. The brief configurations of these datasets are presented in Table 3, which contains the information such as the total number of examples $\bar{n}$, the feature dimensionality d, the number of positive examples n_+, and the number of negative examples n_- for each dataset. Moreover, The features for all methods have been normalized and standardized on every dataset. Five-fold cross validation is applied to all compared approaches on all datasets, and the mean test accuracy as well as standard deviation of the five independent trials on each dataset are reported for algorithm

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2022.3178690, IEEE Transactions on Pattern Analysis and Machine Intelligence

TABLE 4: Comparison of the mean test accuracies (%) of various approaches on five adopted UCI datasets. The best two records on each dataset are highlighted in red and blue, respectively. The "√" ("×") denotes that our CWD is significantly better (worse) than the corresponding existing methods revealed by the paired t-test with significance level 0.05.

Dataset	$(\bar{n}, d)$	$(\eta_{\mathrm{P}}, \eta_{\mathrm{N}})$	$\mathcal{L}_{\mathrm{DMI}}$ [39]	FC [10]	GCE [23]	SCE [8]	f-Div [40]	SR [41]	LICS [11]	μSGD [35]	ULE [9]	CEGE [12]	CWD
Heart	(270, 13)	(0.0, 0.0)	57.4 ± 8.8 √	50.5 ± 6.8 √	53.5 ± 8.7 √	82.6 ± 4.6	81.2±6.7	61.6±14.5 √	75.4 ± 4.8 √	57.3 ± 7.4 √	83.3 ± 3.3	73.9 ± 5.8 √	84.9 ± 4.1
		(0.2, 0.2)	37.3 ± 6.4 √	47.1 ± 5.7 √	49.8 ± 8.0 √	76.2 ± 9.9 √	80.4±6.8	52.0±10.1 √	62.2 ± 10.1 √	59.2 ± 8.5 √	81.8 ± 4.0	75.4 ± 4.3 √	81.1 ± 6.3
		(0.3, 0.1)	40.3 ± 7.9 √	53.9 ± 8.9 √	52.4 ± 11.0 √	60.0 ± 7.0 √	67.6±9.5 √	51.6±5.8 √	70.9 ± 9.2 √	55.8 ± 8.4 √	81.5 ± 7.8	75.8 ± 5.6 √	80.0 ± 8.5
		(0.4, 0.4)	45.2 ± 5.8 √	44.9 ± 6.2 √	53.9 ± 9.5 √	63.3 ± 7.9 √	76.0±6.5	56.4±7.0 √	57.3 ± 11.1 √	64.9 ± 3.0 √	75.8 ± 3.2	72.0 ± 5.6 √	76.2 ± 3.4
Blood	(748, 4)	(0.0, 0.0)	55.7 ± 24.9 √	78.7 ± 24.5	41.8 ± 23.5 √	75.0 ± 4.3	76.7±1.3	76.4±1.4	77.4 ± 2.4	76.2 ± 1.9	77.3 ± 2.4	76.2 ± 1.9	77.5 ± 2.4
		(0.2, 0.2)	76.1 ± 4.6 √	76.2 ± 4.4 √	42.2 ± 24.4 √	76.2 ± 4.3 √	77.5±2.8	71.1±5.2 √	76.3 ± 4.7 √	75.9 ± 6.1 √	76.7 ± 5.5 √	76.9 ± 4.4 √	79.1 ± 5.1
		(0.3, 0.1)	63.6 ± 22.5 √	76.1 ± 3.5 √	44.1 ± 24.2 √	76.1 ± 3.5 √	75.2±2.4	76.4±4.6	76.2 ± 2.2 √	75.7 ± 0.6 √	70.8 ± 7.0 √	76.5 ± 3.3 √	78.2 ± 2.9
		(0.4, 0.4)	44.1 ± 26.0 √	76.1 ± 0.3 √	42.0 ± 22.9 √	76.3 ± 3.6	74.8±4.3	68.5±9.9 √	76.1 ± 3.4 √	71.0 ± 4.7 √	71.0 ± 8.3 √	76.1 ± 3.9	76.4 ± 3.6
Diabetes	(768, 8)	(0.0, 0.0)	49.4 ± 15.4 √	58.8 ± 10.6 √	66.9 ± 3.7	65.3 ± 2.5 √	66.1±3.9 √	64.9±2.9	67.0 ± 2.6 √	65.2 ± 3.2 √	65.4 ± 3.6 √	74.1 ± 2.8 ×	70.8 ± 6.3
		(0.2, 0.2)	53.5 ± 15.4 √	57.2 ± 13.8 √	66.5 ± 4.8 √	71.5 ± 2.4 √	66.5±3.3 √	65.3±2.2 √	70.4 ± 3.5 √	65.2 ± 3.1 √	76.0 ± 3.1	73.9 ± 2.2 √	76.1 ± 2.0
		(0.3, 0.1)	37.3 ± 5.6 √	40.7 ± 12.0 √	56.6 ± 11.0 √	69.9 ± 3.0 √	63.7±5.9 √	47.2±14.3 √	64.7 ± 4.2 √	50.1 ± 9.9 √	72.2 ± 2.5	70.7 ± 2.3 √	73.2 ± 4.0
		(0.4, 0.4)	36.9 ± 13.2 √	39.2 ± 11.5 √	46.7 ± 13.3 √	66.9 ± 2.6 √	66.3±3.2 √	54.5±14.4 √	66.5 ± 6.4 √	65.2 ± 2.6 √	70.7 ± 3.0 √	67.8 ± 4.2 √	72.2 ± 4.2
GermanCredit	(1000, 24)	(0.0, 0.0)	51.3 ± 17.7 √	69.9 ± 1.5 √	70.4 ± 1.7 √	75.7 ± 3.1	75.1±2.0	70.2±2.0 √	66.2 ± 4.8 √	69.8 ± 1.9 √	76.3 ± 3.1	54.5 ± 3.4 √	76.8 ± 3.1
		(0.2, 0.2)	43.8 ± 17.3 √	68.1 ± 4.4 √	68.0 ± 4.1 √	70.5 ± 2.4 √	74.0±1.5	70.1±1.8	57.9 ± 5.9 √	70.0 ± 1.9 √	73.7 ± 3.7	71.5 ± 2.2 √	74.6 ± 3.3
		(0.3, 0.1)	55.9 ± 18.8 √	63.9 ± 5.4 √	70.6 ± 5.1 √	70.3 ± 3.9 √	71.7±4.5	71.2±3.6	59.7 ± 3.5 √	70.1 ± 5.6 √	71.7 ± 2.3	72.6 ± 3.5	72.8 ± 2.9
		(0.4, 0.4)	45.0 ± 15.2 √	62.2 ± 15.2 √	63.7 ± 14.2 ×	67.9 ± 4.0 ×	67.0±4.5	70.1±2.8 ×	52.6 ± 3.8 √	69.8 ± 4.0 ×	65.8 ± 4.7	62.3 ± 2.8 √	65.4 ± 4.2
EEGEyeState	(14980, 16)	(0.0, 0.0)	54.6 ± 3.9 √	55.2 ± 0.5 √	55.4 ± 0.8 √	58.2 ± 0.4	59.9±1.0	57.4±0.9	52.6 ± 4.5	55.1 ± 0.6	56.8 ± 4.6	57.8 ± 0.2	57.9 ± 0.7
		(0.2, 0.2)	51.1 ± 4.2 √	56.7 ± 1.3 √	55.0 ± 0.7 √	56.9 ± 0.9 √	58.5±1.4	55.3±0.7 √	55.1 ± 0.3 √	57.3 ± 0.3 √	47.4 ± 5.8 √	57.9 ± 1.0 √	61.5 ± 4.5
		(0.3, 0.1)	47.1 ± 3.3 √	51.3 ± 5.7 √	44.8 ± 0.4 √	49.4 ± 0.6 √	55.3±0.6	46.5±2.3 √	44.8 ± 0.4 √	46.0 ± 0.4 √	50.8 ± 4.4 √	51.8 ± 0.6 √	53.6 ± 2.2
		(0.4, 0.4)	46.8 ± 5.2 √	54.5 ± 2.9 √	55.6 ± 0.7 √	57.1 ± 0.9 √	56.7±0.8 √	55.5±0.8 √	55.1 ± 0.5 √	57.8 ± 0.3 √	48.9 ± 7.5 √	57.3 ± 0.3 √	60.2 ± 3.0
Average			**49.5**	**55.2**	**54.9**	**68.1**	**69.5**	**62.1**	**64.1**	**63.8**	**69.6**	**68.7**	**72.4**

evaluation. Besides, the paired t-test with significance level 0.05 is employed to check whether our method is statistically better or worse than various baseline approaches. The test accuracies of different methods are particularly reported under three label flip rates including $(\eta_{\mathrm{P}} = 0.2, \eta_{\mathrm{N}} = 0.2)$, $(\eta_{\mathrm{P}} = 0.4, \eta_{\mathrm{N}} = 0.4)$, and $(\eta_{\mathrm{P}} = 0.3, \eta_{\mathrm{N}} = 0.1)$, among which the first two cases are symmetric noise while the last case is asymmetric noise. Besides, to show our method can well deal with the noise-free situations as mentioned in Section 3, we also take the label flip rate $(\eta_{\mathrm{P}} = 0.0, \eta_{\mathrm{N}} = 0.0)$ into consideration. For fair comparison, the contaminated examples in each fold are kept identical for all compared methods on every dataset.

For some methods that require label flip rates as input such as μSGD, LICS, ULE, FC and our CWD, we directly send the real values of η_{P} and η_{N} to them. For LICS, the trade-off parameter λ is carefully tuned via searching the grid $\{10^{-2}, 10^{-1}, 10^{0}, 10^{1}, 10^{2}\}$ on every dataset, and the hyper-parameter β is set to 10^{-5} to achieve the top level performance. For GCE, the hyper-parameter q for the negative Box-Cox transformation is set to 0.7 as recommended by [23]. For SCE, we follow [8] and tune the trade-off parameters α and β within $\{10^{-2}, 10^{-1}, 10^{0}, 10^{1}, 10^{2}\}$ and $\{10^{-1}, 10^{0}\}$, respectively. For SR, by following [41], we set the regularization parameter $p = 0.1$, sharpening parameter $\tau = 0.5$, and tune the trade-off parameter λ from $\{0.1, 0.3, 1.0, 3.0, 5.0, 7.0, 10, 15, 20\}$. For FC, $\mathcal{L}_{\mathrm{DMI}}$, GCE, SCE, f-Div, SR, CEGE and our CWD, we employ the same MLP with three layers as backbone network, where the number of nodes in input layer equals to the data dimensionality d, and the number of nodes in output layer is 1. Therefore, the amount of nodes in hidden layer is decided as $\mathrm{round}(2/3 \times (d + 1))$ to achieve satisfactory performance. The Adam optimizer [43] with weight decay factor of 10^{-4} is employed for network training.

The experimental results of all methods on the adopted UCI datasets are presented in Table 4, which reveals that the proposed CWD generally ranks among the top two compared methods. For the average accuracy over all five datasets under different label flip rates, our CWD obtains a record of 72.4% which leads the second best method ULE by a margin of 2.8%. Although the average accuracies of CWD are lower than those of ULE on *Heart* dataset when $(\eta_{\mathrm{P}} = 0.2, \eta_{\mathrm{N}} = 0.2)$ and $(\eta_{\mathrm{P}} = 0.3, \eta_{\mathrm{N}} = 0.1)$, we see that the performances of these two methods are statistically comparable as revealed by the t-test. The same phenomena can also be observed on *EEGEyeState* dataset when comparing the accuracies of f-Div and our CWD.

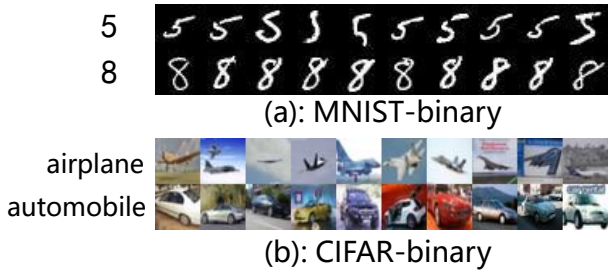

Fig. 2: Example images of binary classification datasets. (a) presents *MNIST-binary*, and (b) presents *CIFAR-binary*.

6.2 Experiments on Real-world Binary Datasets

To further validate the effectiveness of CWD, we study the performances of CWD as well as the ten baseline methods (*i.e.*, UE, μSGD, LICS, FC, $\mathcal{L}_{\mathrm{DMI}}$, GCE, SCE, f-Div, SR, and CEGE) on two practical two-class datasets, which include:

- *MNIST-binary*. This dataset originates from *MNIST* [44] dataset, which cares about handwritten digit recognition. The original *MNIST* contains 60,000 gray-scale images of size 28×28 belonging to the ten digits "0"∼"9", and each digit corresponds to a class. Here we follow [45] and select the images of "5" and "8" for classification (see Fig. 2(a)), and the size of dataset for our experiment is 12,000.
- *CIFAR-binary*. This dataset is a subset of *CIFAR* dataset [46], which is related to natural image classification. The original *CIFAR* consists of 60,000 colored natural images with size $32 \times 32 \times 3$. For our experiment, we follow [45] and pick up the image examples of two categories (*i.e.*, "airplane" and "automobile") for conduct binary classification (see Fig. 2(b)).

On both datasets, we also conduct five-fold cross validation to investigate the performances of all methods, and their mean test accuracies of five independent trials are particularly observed. To create label noise, we randomly flip the groundtruth labels of the images of every class to the opposite value under two symmetric cases including $(\eta_{\mathrm{P}} = 0.2, \eta_{\mathrm{N}} = 0.2)$ and $(\eta_{\mathrm{P}} = 0.4, \eta_{\mathrm{N}} =$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2022.3178690, IEEE Transactions on Pattern Analysis and Machine Intelligence

TABLE 5: Comparison of the mean test accuracies (%) of various approaches on *MNIST-binary* dataset. The best two records on each dataset are highlighted in red and blue, respectively. The "$\surd$" ("$\times$") denotes that our CWD is significantly better (worse) than the corresponding existing methods revealed by the paired t-test with significance level 0.05.

$(\eta_{\mathrm{P}}, \eta_{\mathrm{N}})$	$\mathcal{L}_{\mathrm{DMI}}$ [39]	FC [10]	GCE [23]	SCE [8]	f-Div [40]	SR [41]	LICS [11]	μSGD [35]	ULE [9]	CEGE [12]	CWD
(0.0, 0.0)	96.0 ± 0.6 $\surd$	96.1 ± 0.3 $\surd$	96.6 ± 0.3 $\surd$	97.0 ± 0.3 $\surd$	96.4 ± 0.3 $\surd$	98.5 ± 0.2	80.7 ± 3.2 $\surd$	86.5 ± 5.0 $\surd$	96.0 ± 0.2 $\surd$	89.8 ± 0.7 $\surd$	99.0 ± 0.1
(0.2, 0.2)	95.0 ± 0.7 $\surd$	95.0 ± 0.6 $\surd$	95.8 ± 0.4 $\surd$	94.5 ± 0.3 $\surd$	96.1 ± 0.4	95.9 ± 0.4	76.8 ± 3.4 $\surd$	86.2 ± 4.3 $\surd$	95.0 ± 0.5 $\surd$	89.8 ± 1.0 $\surd$	97.3 ± 0.1
(0.3, 0.1)	94.1 ± 0.6	92.0 ± 0.9 $\surd$	94.5 ± 0.8	91.5 ± 1.3 $\surd$	94.7 ± 0.6	83.9 ± 1.3 $\surd$	75.3 ± 1.4 $\surd$	68.0 ± 7.4 $\surd$	90.8 ± 0.6 $\surd$	89.5 ± 1.0 $\surd$	95.1 ± 0.2
(0.4, 0.4)	59.8 ± 11.5 $\surd$	91.4 ± 2.3 $\surd$	90.0 ± 1.3 $\surd$	85.0 ± 1.4 $\surd$	94.1 ± 1.0	88.6 ± 2.4 $\surd$	60.5 ± 2.4 $\surd$	76.0 ± 5.9 $\surd$	88.9 ± 1.5 $\surd$	87.5 ± 1.7 $\surd$	94.2 ± 0.1
Average	**86.1**	**93.6**	**94.2**	**91.9**	**95.3**	**91.7**	**73.2**	**79.1**	**92.6**	**89.1**	**96.3**

TABLE 6: Comparison of the mean test accuracies (%) of various approaches on *CIFAR-Binary* dataset. The best two records on each dataset are highlighted in red and blue, respectively. The "$\surd$" ("$\times$") denotes that our CWD is significantly better (worse) than the corresponding existing methods revealed by the paired t-test with significance level 0.05.

$(\eta_{\mathrm{P}}, \eta_{\mathrm{N}})$	$\mathcal{L}_{\mathrm{DMI}}$ [39]	FC [10]	GCE [23]	SCE [8]	f-Div [40]	SR [41]	LICS [11]	μSGD [35]	ULE [9]	CEGE [12]	CWD
(0.0, 0.0)	98.8 ± 0.3	64.1 ± 5.2 $\surd$	97.8 ± 0.2	98.6 ± 0.9	98.3 ± 0.2	96.4 ± 0.3 $\surd$	88.0 ± 9.5 $\surd$	98.4 ± 0.2	98.1 ± 0.4	96.5 ± 0.1 $\surd$	98.9 ± 0.1
(0.2, 0.2)	83.9 ± 17.0 $\surd$	66.9 ± 4.4 $\surd$	87.1 ± 1.2 $\surd$	86.5 ± 5.7 $\surd$	95.3 ± 0.2 $\surd$	96.1 ± 0.4	84.6 ± 1.7 $\surd$	94.7 ± 1.7 $\surd$	71.3 ± 4.2 $\surd$	96.2 ± 0.1	97.4 ± 0.2
(0.3, 0.1)	81.2 ± 15.0 $\surd$	65.3 ± 3.2 $\surd$	86.2 ± 0.5 $\surd$	85.7 ± 5.8 $\surd$	95.6 ± 0.3	95.7 ± 0.6	68.4 ± 2.9 $\surd$	75.1 ± 1.1 $\surd$	69.0 ± 9.6 $\surd$	96.3 ± 1.0	96.6 ± 0.2
(0.4, 0.4)	62.7 ± 4.3 $\surd$	55.5 ± 5.5 $\surd$	71.5 ± 3.5 $\surd$	71.3 ± 8.8 $\surd$	76.7 ± 0.7	74.2 ± 0.1 $\surd$	55.6 ± 2.6 $\surd$	65.1 ± 1.7 $\surd$	56.3 ± 4.3 $\surd$	65.5 ± 0.6 $\surd$	77.6 ± 4.9
Average	**81.6**	**62.9**	**85.6**	**85.4**	**91.5**	**90.6**	**74.1**	**83.3**	**73.6**	**88.6**	**92.6**

$0.4)$, and one asymmetric case which is $(\eta_{\mathrm{P}} = 0.3, \eta_{\mathrm{N}} = 0.1)$. The situation of $(\eta_{\mathrm{P}} = 0.0, \eta_{\mathrm{N}} = 0.0)$ is also investigated.

The hyper-parameters of various methods are tuned via the similar way as in Section 6.1. Concretely, the trade-off parameter λ in LICS is tuned via searching the grid $\{10^{-2}, 10^{-1}, 10^{0}, 10^{1}, 10^{2}\}$, and it is set to 10^{-1} and 10^{-2} to achieve the optimal results on *MNIST-binary* and *CIFAR-binary*, respectively. For GCE, the hyper-parameter q for the negative Box-Cox transformation is set to 0.7 as recommended by [23] on both datasets. For SCE, the parameters α and β are also tuned via grid search within $\{10^{-2}, 10^{-1}, 10^{0}, 10^{1}, 10^{2}\}$ and $\{10^{-1}, 10^{0}\}$, respectively. For SR, we set $\lambda = 3$ and $\lambda = 1$ on *MNIST-binary* and *CIFAR-binary* accordingly, and set $p = 0.1$ and $\tau = 0.5$ on both datasets. For nonlinear methods such as FC, $\mathcal{L}_{\mathrm{DMI}}$, GCE, SCE, f-Div, SR, CEGE and our CWD, we employ MLP with three layers as the backbone network on *MNIST-binary* dataset, and adopt ResNet-34 [47] as the backbone network on *CIFAR-binary* dataset. On *MNIST-binary*, the pixel-wise gray-scale intensity values are taken as image features for the input for all investigated methods. Differently, on *CIFAR-binary*, the deep methods such as FC, $\mathcal{L}_{\mathrm{DMI}}$, GCE, SCE, f-Div, SR, CEGE and CWD can automatically extract CNN features by ResNet-34. However, for other originally non-deep methods such as LICS, μSGD and ULE, we use a ResNet-34 network pre-trained on ImageNet dataset to extract image features, and then send them to the corresponding algorithm for performance evaluation. Adam [43] optimizer is deployed for training the MLP or ResNet-34 network. Specifically, on *MNIST-binary*, our CWD adopts the default parameters in Adam. On *CIFAR-binary*, we use the Adam optimizer with weight decay factor of 10^{-4} for CWD. The learning rate is 0.05 initially and is divided by 10 after 40 and 120 epochs (200 in total).

The experimental results of various methods on *MNIST-binary* and *CIFAR-binary* are presented in Table 5 and Table 6, respectively, which suggest that in most cases, our CWD obtains the highest average accuracy when compared with other baseline approaches on both datasets under different label flip rates. Particularly, we see that because ULE, μSGD, LICS and FC only take the unbiasedness into consideration when constructing the risk estimator, their acquired test accuracies are significantly lower than our CWD. This validates the importance of statistical efficiency which is considered by our CWD. Moreover, as mentioned in Introduction, CEGE [36] also introduces two virtual auxiliary sets to achieve unbiased and statistically efficient centroid estimation, but the way of CEGE in establishing the virtual auxiliary sets is inferior to that of the proposed CWD, so it yields worse performance than CWD under different label flip rates. Finally, we see that under clean set with $(\eta_{\mathrm{P}} = 0.0, \eta_{\mathrm{N}} = 0.0)$, our method can achieve the accuracy as high as 99.0% and 98.9% on *MNIST-binary* and *CIFAR-binary* correspondingly, which are significantly better than some noise-robust methods such as $\mathcal{L}_{\mathrm{DMI}}$, FC, GCE and SCE. This is because these noise-robust methods are designed under the assumption that the training set should contain label noise, which is obviously not satisfied when $(\eta_{\mathrm{P}} = 0.0, \eta_{\mathrm{N}} = 0.0)$.

6.3 Experiments on Real-World Multi-Class Datasets

To test the ability of our CWD in tackling multi-class classification with real-world label noise, here we adopt the following three practical datasets for performance evaluation:

- *Animal-10N* [48]. This dataset contains totally 55,000 images of 10 different animals, where 50,000 images are for training and 5,000 images are provided for testing. Since this dataset intentionally collects some pairs of visually confusing animals such as "cat" vs. "lynx" and "chimpanzee" vs. "orangutan" (see Fig. 3(a)), labeling errors would be naturally brought in during the annotation process.
- *Clothing-1M* [49]. This dataset contains 1 million clothing images belonging to 14 classes such as "T-shirt", "Jacket", and "Vest" (see Fig. 3(b)). Since the data is directly crawled from several online shopping websites, and the labels are automatically generated according to surrounding texts of these images, this dataset inevitably contains label noise.
- *CIFAR-100* [46]. This dataset consists of 60,000 colour images in 100 classes, with 600 images per class. There are 500 training images and 100 test images per class, and each image is associated with a fine-grained label (see Fig. 3(c)).

Among the above three datasets, *Animal-10N* and *Clothing-1M* contain the noisy labels naturally injected by human mistakes, and the divisions of training and test sets are also directly provided. For *CIFAR-100*, its original labels are all clean. To add label noise, we follow [4], [23], [50] and investigate two types of label

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2022.3178690, IEEE Transactions on Pattern Analysis and Machine Intelligence

Fig. 3: Example images of multi-class datasets. (a) presents *Animal-10N*, (b) presents *Clothing-1M*, and (c) presents *CIFAR-100*.

noise, namely: 1) symmetric noise with rates[5] $\eta = \{20\%, 40\%\}$ (denoted as "Symmetry-η"), which means that the corrupted label is uniformly assigned to one of $C-1$ incorrect classes with probability $\eta/(C-1)$; and 2) Pair flipping noise with rates $\eta = \{20\%, 40\%\}$ (denoted as "Pairflip-η"), which means that the label of each class is flipped into the next class circularly with probability η. Besides, the noise-free case with noise rate $\eta = 0\%$ is also studied (denoted as "Clean-0%").

The baseline methods in this section include the previously used $\mathcal{L}_{\mathrm{DMI}}$, FC, GCE, SCE, f-Div and SR. Here LICS, μSGD, ULE and CEGE are not compared as they can only handle binary classification. The hyper-parameters of various methods are tuned via the similar way as before. For GCE, the hyper-parameter q for the negative Box-Cox transformation is set to 0.6. For SCE, the parameters $\{\alpha, \beta\}$ are respectively adjusted to $\{0.5, 0.5\}$, $\{3.0, 0.1\}$ and $\{6.0, 0.1\}$ on *Animal-10N*, *Clothing-1M* and *CIFAR-100* to achieve optimal performance. For SR, the parameters p and τ are kept to 0.01 and 0.5 on the three datasets, and λ is set to 3, 5 and 10 on *Animal-10N*, *Clothing-1M* and *CIFAR-100* correspondingly. The backbone networks employed by all compared methods are VGG-19 for *Animal-10N*, ResNet-50 for *Clothing-1M*, and ResNet-34 for *CIFAR-100*. In our CWD, the label flip matrix $\boldsymbol{\eta}$ is estimated via [29] according to Algorithm 2.

The experimental results of various methods on *Animal-10N*, *Clothing-1M* and *CIFAR-100* are presented in Tables 7 and 8. For *Animal-10N*, we see that our CWD surpasses the second best method $\mathcal{L}_{\mathrm{DMI}}$ by approximately 2% in terms of test accuracy. For *Clothing-1M* dataset, here we follow the setting in [29] and do not include validation set during training. This is because some baseline methods such as FC, $\mathcal{L}_{\mathrm{DMI}}$ and f-Div rely on a validation set with clean labels for boosting the performance, which is usually not available in practice and will make the comparison not fair. The experimental results reveal that the compared methods achieve very similar performance, and f-Div is slightly better than our CWD by a margin of 0.24%. Regarding *CIFAR-100*, it can be observed that the proposed CWD is significantly better than other competitors in most cases under both symmetric noise and pair flipping noise with different noise rates. In a word, the above experimental results indicate that CWD is also effective in dealing with multi-class classification problems with real-world label noise.

5. Here we slightly abuse the notation η by referring it to label noise rate under multi-class case, as it will degenerate to the label flip rate under binary classification defined above.

TABLE 7: Comparison of test accuracies (%) of various approaches on *Animal-10N* and *Clothing-1M* datasets. The best two records on each dataset are highlighted in red and blue, respectively.

Method	*Animal-10N*	*Clothing-1M*
$\mathcal{L}_{\mathrm{DMI}}$ [39]	80.62	70.22
FC [10]	80.08	69.82
GCE [23]	79.62	69.19
SCE [8]	79.51	69.89
f-Div [40]	77.22	70.65
SR [41]	75.32	68.60
CWD	82.52	70.41

6.4 Effects of Variance Reduction

In section 4.1, we have theoretically proved that our CWD often has equal or lower variance on estimating the data centroid than LICS [11], which is critical for our method to obtain the improved results as illustrated in the above experiments. Here we empirically show this by comparing the variances and centroid estimation errors of LICS and our CWD on five adopted UCI datasets appeared in Section 6.1.

To be specific, we study a symmetric noise case ($\eta_{\mathrm{P}} = 0.4, \eta_{\mathrm{N}} = 0.4$) and an asymmetric noise case ($\eta_{\mathrm{P}} = 0.3, \eta_{\mathrm{N}} = 0.1$), and compare the variances and errors of our CWD and LICS in estimating the real centroid $\hat{\mu}(S)$ of clean training set. The results are displayed in Fig. 4. From the first column of Fig. 4, we can see that our CWD consistently yields lower variance in centroid estimation than LICS on all datasets under both noise types, which coincides with the theoretical finding in Section 4.1, and also verifies the necessity of considering statistical efficiency by our method in reducing the variance of centroid estimation. As a sequel, the proposed CWD obtains smaller or comparable estimation error when compared with LICS as revealed by the second column of Fig. 4, and this is beneficial for our method to achieve good robustness and classification performance.

7 Conclusion and Future Work

In this paper, we proposed a new "Class-Wise Denoising" (CWD) algorithm for robust learning under label noise. The key of CWD is to find an unbiased and statistically efficient data centroid

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2022.3178690, IEEE Transactions on Pattern Analysis and Machine Intelligence

TABLE 8: Comparison of the mean test accuracies (%) of various approaches on *CIFAR-100* dataset. The best two records on each dataset are highlighted in red and blue, respectively.

	f-Div [40]	SR [41]	$\mathcal{L}_{\mathrm{DMI}}$ [39]	FC [10]	GCE [23]	SCE [8]	CWD
Clean-0%	66.6	71.4	68.4	70.3	64.1	65.7	72.9
Symmetry-20%	65.2	63.7	57.6	56.7	62.6	56.2	65.8
Symmetry-40%	64.2	48.7	46.3	46.3	56.7	51.4	53.3
Pairflip-20%	53.0	61.6	57.6	58.7	59.2	58.2	62.8
Pairflip-40%	38.4	45.6	44.1	41.4	43.4	42.3	46.5
Average	**57.4**	58.2	**54.8**	**54.6**	**57.2**	**54.7**	60.2

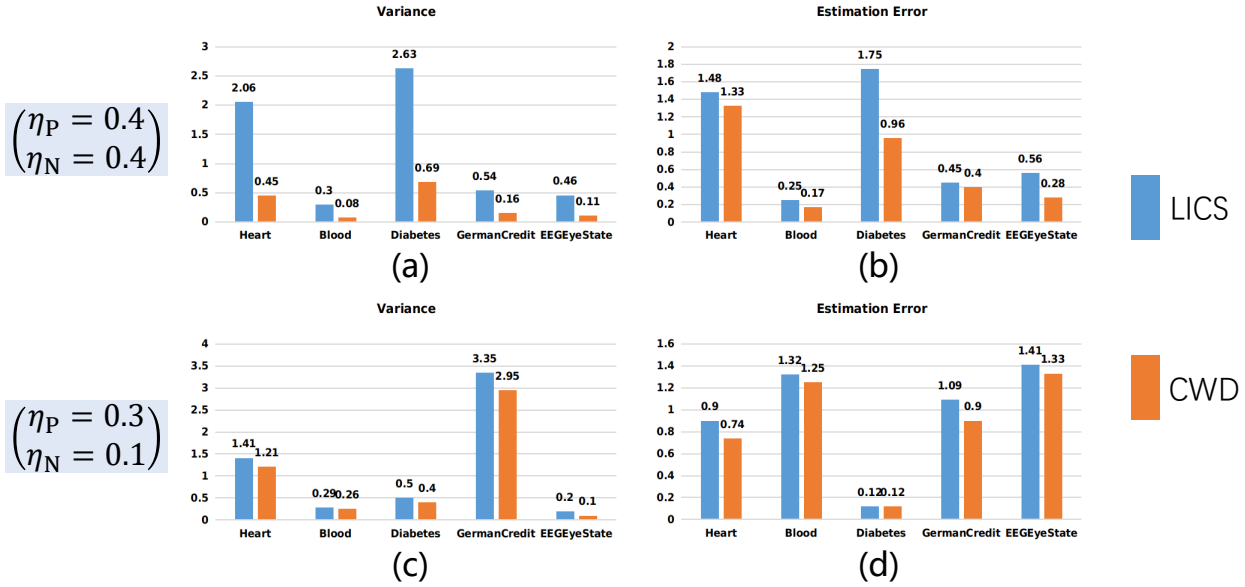

Fig. 4: Comparison of CWD and LICS on variance and error in centroid estimation, where the first row shows the case under label flip rate $(\eta_{\mathrm{P}} = 0.4, \eta_{\mathrm{N}} = 0.4)$, and the second row shows the case under label flip rate $(\eta_{\mathrm{P}} = 0.3, \eta_{\mathrm{N}} = 0.1)$. The **blue** bar and **orange** bar indicate LICS and CWD, respectively. The numerical values are annotated above the bars.

estimator to form a noise-robust empirical risk. To this end, CWD corrects the noisy labels class by class via establishing a series of intermediate virtual auxiliary sets, so that all attention is paid to the corrupted labels in one class at a time. Thanks to the progressive denoising strategy, the resulting centroid estimator is not only unbiased, but also shows equal or lower variance when compared with other state-of-the-art unbiased methods, so our CWD can produce more accurate and reliable classification results than them. The effectiveness of the proposed CWD has been confirmed from both theoretical and empirical aspects.

Regarding CWD, there are several problems to be further studied: 1) Theorem 1 suggests that the good statistical efficiency of CWD can be achieved with a probability of $\ln 2 \approx 0.693$, so whether there exists a better estimator that has a larger probability to be statistically efficient remains unclear; and 2) More advanced techniques are still needed to accurately estimate the label flip rates $(\eta_{\mathrm{P}}, \eta_{\mathrm{N}})$ or label flip matrix $\boldsymbol{\eta}$ for the input of our CWD.

REFERENCES

[1] B. Han, Q. Yao, T. Liu, G. Niu, I. W. Tsang, J. T. Kwok, and M. Sugiyama, "A survey of label-noise representation learning: Past, present and future," *arXiv preprint arXiv:2011.04406*, 2020.

[2] X. Zhu, X. Wu, and Q. Chen, "Eliminating class noise in large datasets," in *International Conference on Machine Learning*, vol. 3, 2003, pp. 920–927.

[3] B. Han, Q. Yao, X. Yu, G. Niu, M. Xu, W. Hu, I. Tsang, and M. Sugiyama, "Co-teaching: Robust training of deep neural networks with extremely noisy labels," in *Advances in Neural Information Processing Systems*, 2018, pp. 8527–8537.

[4] Y. Lyu and I. W. Tsang, "Curriculum loss: Robust learning and generalization against label corruption," in *International Conference on Learning Representations*, 2019.

[5] J. Li, R. Socher, and S. C. Hoi, "Dividemix: Learning with noisy labels as semi-supervised learning," in *International Conference on Learning Representations*, 2019.

[6] A. Ghosh, H. Kumar, and P. Sastry, "Robust loss functions under label noise for deep neural networks," in *AAAI Conference on Artificial Intelligence*, vol. 31, no. 1, 2017.

[7] X. Ma, H. Huang, Y. Wang, S. Romano, S. Erfani, and J. Bailey, "Normalized loss functions for deep learning with noisy labels," in *International Conference on Machine Learning*, 2020, pp. 6543–6553.

[8] Y. Wang, X. Ma, Z. Chen, Y. Luo, J. Yi, and J. Bailey, "Symmetric cross entropy for robust learning with noisy labels," in *IEEE International Conference on Computer Vision*, 2019, pp. 322–330.

[9] N. Natarajan, I. S. Dhillon, P. K. Ravikumar, and A. Tewari, "Learning with noisy labels," in *Advances in Neural Information Processing Systems*, 2013, pp. 1196–1204.

[10] G. Patrini, A. Rozza, A. Krishna Menon, R. Nock, and L. Qu, "Making deep neural networks robust to label noise: A loss correction approach," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1944–1952.

[11] W. Gao, L. Wang, Y.-F. Li, and Z.-H. Zhou, "Risk minimization in the presence of label noise," in *AAAI Conference on Artificial Intelligence*, 2016, pp. 1575–1581.

[12] C. Gong, J. Yang, J. J. You, and M. Sugiyama, "Centroid estimation with guaranteed efficiency: A general framework for weakly supervised learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[13] C. Brodley and M. Friedl, "Identifying mislabeled training data," *Journal of Artificial Intelligence Research*, vol. 11, pp. 131–167, 1999.

[14] D. Arpit, S. Jastrzebski, N. Ballas, D. Krueger, E. Bengio, M. Kanwal, T. Maharaj, A. Fischer, A. Courville, and Y. Bengio, "A closer look at memorization in deep networks," in *International Conference on Machine Learning*, 2017, pp. 233–242.

[15] L. Jiang, Z. Zhou, T. Leung, L.-J. Li, and L. Fei-Fei, "Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels," in *International Conference on Machine Learning*, 2018, pp. 2304–2313.

[16] X. Yu, B. Han, J. Yao, G. Niu, I. Tsang, and M. Sugiyama, "How does disagreement help generalization against label corruption?" in *International Conference on Machine Learning*, 2019, pp. 7164–7173.

[17] H. Wei, L. Feng, X. Chen, and B. An, "Combating noisy labels by agreement: A joint training method with co-regularization," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 13 726–13 735.

[18] Q. Yao, H. Yang, B. Han, G. Niu, and J. T.-Y. Kwok, "Searching to exploit memorization effect in learning with noisy labels," in *International Conference on Machine Learning*, 2020, pp. 10 789–10 798.

[19] B. Han, G. Niu, X. Yu, Q. Yao, M. Xu, I. Tsang, and M. Sugiyama, "SIGUA: Forgetting may make learning with noisy labels more robust," in *International Conference on Machine Learning*. PMLR, 2020, pp. 4006–4016.

[20] K. Yi and J. Wu, "Probabilistic end-to-end noise correction for learning with noisy labels," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 7017–7025.

[21] D. Tanaka, D. Ikami, T. Yamasaki, and K. Aizawa, "Joint optimization framework for learning with noisy labels," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5552–5560.

[22] A. Vahdat, "Toward robustness against label noise in training deep discriminative neural networks," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 2017, pp. 5601–5610.

[23] Z. Zhang and M. Sabuncu, "Generalized cross entropy loss for training deep neural networks with noisy labels," *Advances in neural information processing systems*, vol. 31, pp. 8778–8788, 2018.

[24] P. Hu, X. Peng, H. Zhu, L. Zhen, and J. Lin, "Learning cross-modal retrieval with noisy labels," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2021, pp. 5403–5413.

[25] L. Feng, S. Shu, Z. Lin, F. Lv, L. Li, and B. An, "Can cross entropy loss be robust to label noise," in *International Joint Conferences on Artificial Intelligence*, 2020, pp. 2206–2212.

[26] T. Liu and D. Tao, "Classification with noisy labels by importance reweighting," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 3, pp. 447–461, 2016.

[27] X. Xia, T. Liu, N. Wang, B. Han, C. Gong, G. Niu, and M. Sugiyama, "Are anchor points really indispensable in label-noise learning?" in *Advances in Neural Information Processing Systems*, 2019, pp. 6835–6846.

[28] Y. Yao, T. Liu, B. Han, M. Gong, J. Deng, G. Niu, and M. Sugiyama, "Dual t: Reducing estimation error for transition matrix in label-noise learning," in *Advances in Neural Information Processing Systems*, 2020, pp. 1–12.

[29] X. Li, T. Liu, B. Han, G. Niu, and M. Sugiyama, "Provably end-to-end label-noise learning without anchor points," in *International Conference on Machine Learning*, 2021, pp. 6403–6413.

[30] C. G. Northcutt, T. Wu, and I. L. Chuang, "Learning with confident examples: Rank pruning for robust classification with noisy labels," in *The Conference on Uncertainty in Artificial Intelligence*, 2017.

[31] C. Scott, "A rate of convergence for mixture proportion estimation, with application to learning from noisy labels," in *International Conference on Artificial Intelligence and Statistics*, 2015, pp. 838–846.

[32] J. Goldberger and E. Ben-Reuven, "Training deep neural-networks using a noise adaptation layer," in *International Conference on Learning Representations*, 2017, pp. 1–9.

[33] X. Xia, T. Liu, B. Han, N. Wang, M. Gong, H. Liu, G. Niu, D. Tao, and M. Sugiyama, "Parts-dependent label noise: Towards instance-dependent label noise," in *Advances in Neural Information Processing Systems*, 2020, pp. 1–12.

[34] N. Natarajan, I. S. Dhillon, P. Ravikumar, and A. Tewari, "Cost-sensitive learning with noisy labels," *Journal of Machine Learning Research*, vol. 18, no. 1, pp. 5666–5698, 2017.

[35] G. Patrini, F. Nielsen, R. Nock, and M. Carioni, "Loss factorization, weakly supervised learning and label noise robustness," in *International Conference on Machine Learning*, 2016, pp. 708–717.

[36] C. Gong, H. Shi, T. Liu, C. Zhang, J. Yang, and D. Tao, "Loss decomposition and centroid estimation for positive and unlabeled learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

[37] K. Lee, S. Yun, K. Lee, H. Lee, B. Li, and J. Shin, "Robust inference via generative classifiers for handling noisy labels," in *International Conference on Machine Learning*, 2019, pp. 3763–3772.

[38] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[39] Y. Xu, P. Cao, Y. Kong, and Y. Wang, "$\mathcal{L}_{\mathrm{DMI}}$: A novel information-theoretic loss function for training deep nets robust to label noise." in *Advances in Neural Information Processing Systems*, 2019, pp. 6222–6233.

[40] J. Wei and Y. Liu, "When optimizing f-divergence is robust with label noise," in *International Conference on Learning Representations*, 2021, pp. 1–11.

[41] X. Zhou, X. Liu, C. Wang, D. Zhai, J. Jiang, and X. Ji, "Learning with noisy labels via sparse regularization," in *IEEE International Conference on Computer Vision and Pattern Recognition*, 2021, pp. 72–81.

[42] D. Dua and C. Graff, "UCI machine learning repository," 2017. [Online]. Available: http://archive.ics.uci.edu/ml

[43] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *International Conference on Learning Representations*, 2015.

[44] D. C. Cireşan, U. Meier, J. Masci, L. M. Gambardella, and J. Schmidhuber, "High-performance neural networks for visual object classification," *arXiv preprint arXiv:1102.0183*, 2011.

[45] W. Hu, Z. Li, and D. Yu, "Simple and effective regularization methods for training on noisily labeled data with generalization guarantee," in *International Conference on Learning Representations*, 2019.

[46] A. Krizhevsky, G. Hinton *et al.*, "Learning multiple layers of features from tiny images," *Tech report*, 2009.

[47] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[48] H. Song, M. Kim, and J.-G. Lee, "SELFIE: Refurbishing unclean samples for robust deep learning," in *International Conference on Machine Learning*, 2019.

[49] T. Xiao, T. Xia, Y. Yang, C. Huang, and X. Wang, "Learning from massive noisy labeled data for image classification," in *IEEE International Conference on Computer Vision and Pattern Recognition*, 2015, pp. 2691–2699.

[50] D. T. Nguyen, C. K. Mummadi, T. P. N. Ngo, T. H. P. Nguyen, L. Beggel, and T. Brox, "SELF: Learning to filter noisy labels with self-ensembling," in *International Conference on Learning Representations*, 2020, pp. 1–10.

Chen Gong received his dual doctoral degree from Shanghai Jiao Tong University and the University of Technology Sydney in 2016 and 2017, respectively. Currently, he is a professor with Nanjing University of Science and Technology. His research interests mainly include machine learning and data mining. He has published more than 100 technical papers at prominent journals and conferences such as JMLR, IEEE T-PAMI, IEEE T-NNLS, IEEE T-IP, ACM T-IST, CVPR, ICML, NeurIPS, AAAI, IJCAI, ICDM, etc. He also serves as the reviewer for more than 20 international journals such as JMLR, AIJ, IEEE T-PAMI, IJCV, IEEE T-NNLS, IEEE T-IP, and also the (S)PC member of several top-tier conferences such as ICML, NeurIPS, ICLR, CVPR, ICCV, AAAI, IJCAI, ICDM, etc. He won the "Excellent Doctorial Dissertation Award" of Chinese Association for Artificial Intelligence, "Young Elite Scientists Sponsorship Program" of China Association for Science and Technology, "Wu Wen-Jun AI Excellent Youth Scholar Award", and the Science Fund for Distinguished Young Scholars of Jiangsu Province. He was also selected as the "Global Top Chinese Young Scholars in AI" released by Baidu.

Yongliang Ding received his bachelor degree from Nanjing University of Science and Technology (NJUST) in 2020. Currently, he is pursing master degree in NJUST under the supervision of Prof. Chen Gong. His research interests mainly lie in weakly-supervised learning.

Bo Han is an Assistant Professor of Computer Science at Hong Kong Baptist University, and a BAIHO Visiting Scientist at RIKEN Center for Advanced Intelligence Project (RIKEN AIP). He was a Postdoc Fellow at RIKEN AIP (2019-2020). He received his Ph.D. degree in Computer Science from University of Technology Sydney in 2019. He has served as area chairs of NeurIPS, ICML and ICLR, action editors of Transactions on Machine Learning Research and Neural Networks.

Gang Niu is currently an indefinite-term research scientist at RIKEN Center for Advanced Intelligence Project. He received the PhD degree in computer science from Tokyo Institute of Technology in 2013. Before joining RIKEN as a research scientist, he was a senior software engineer at Baidu and then an assistant professor at the University of Tokyo. He has published more than 90 journal articles and conference papers, including 31 ICML, 17 NeurIPS (1 oral and 3 spotlights), and 11 ICLR (1 outstanding paper honorable mention, 2 orals, and 1 spotlight) papers. He has co-authored the book “Machine Learning from Weak Supervision: An Empirical Risk Minimization Approach” (the MIT Press). On the other hand, he has served as an area chair 17 times, including ICLR 2021-2022, ICML 2019-2022, and NeurIPS 2019-2022. He also serves/has served as an action editor of TMLR and a guest editor of a special issue at MLJ. Moreover, he has served as a publication chair for ICML 2022, and has co-organized 9 workshops, 1 competition, and 2 tutorials.

Jian Yang received the PhD degree from Nanjing University of Science and Technology (NUST), on the subject of pattern recognition and intelligence systems in 2002. In 2003, he was a Postdoctoral researcher at the University of Zaragoza. From 2004 to 2006, he was a Postdoctoral Fellow at Biometrics Centre of Hong Kong Polytechnic University. From 2006 to 2007, he was a Postdoctoral Fellow at Department of Computer Science of New Jersey Institute of Technology. From 2006 to 2007, he was a Chang-Jiang professor in the School of Computer Science and Engineering of NUST. Now, he is a distinguished professor in the College of Computer Science of Nankai University. He is the author of more than 200 scientific papers in pattern recognition and computer vision. His papers have been cited over 30000 times in the Google Scholar. His research interests include pattern recognition, computer vision and machine learning. Currently, he is/was an associate editor of Pattern Recognition, Pattern Recognition Letters, IEEE Trans. Neural Networks and Learning Systems, and Neurocomputing. He is a Fellow of IAPR.

Jane You received the B.Eng. degree in electronics engineering from Xi’an Jiaotong University, Xi’an, China, in 1986, and the Ph.D. degree in computer science from La Trobe University, Melbourne, VIC, Australia, in 1992. She was a Lecturer with the University of South Australia, Adelaide SA, Australia, and a Senior Lecturer with Griffith University, Nathan, QLD, Australia, from 1993 to 2002. She is currently a Full Professor with The Hong Kong Polytechnic University, Hong Kong. Her current research interests include image processing, pattern recognition, medical imaging, biometrics computing, multimedia systems, and data mining.

Dacheng Tao (F’15) is the Inaugural Director of the JD Explore Academy and a Senior Vice President of JD.com. He is also an advisor and chief scientist of the digital science institute in the University of Sydney. He mainly applies statistics and mathematics to artificial intelligence and data science, and his research is detailed in one monograph and over 200 publications in prestigious journals and proceedings at leading conferences. He received the 2015 Australian Scopus-Eureka Prize, the 2018 IEEE ICDM Research Contributions Award, and the 2021 IEEE Computer Society McCluskey Technical Achievement Award. He is a fellow of the Australian Academy of Science, AAAS, ACM and IEEE.

Masashi Sugiyama received a Ph.D. degree in Computer Science from Tokyo Institute of Technology, Japan, in 2001. After experiencing assistant and associate professors at the same institute, he became a professor at the University of Tokyo in 2014. Since 2016, he has concurrently served as Director of RIKEN Center for Advanced Intelligence Project. His research interests include theories and algorithms of machine learning. He was a recipient of the Japan Academy Medal in 2017 and the Commendation for Science and Technology by the Minister of Education, Culture, Sports, Science and Technology in Japan in 2022.