Estimating Per-Class Statistics for Label Noise Learning

Wenshui Luo, Shuo Chen, Tongliang Liu, *Senior Member, IEEE,* Bo Han, *Senior Member, IEEE,* Gang Niu, *Senior Member, IEEE,* Masashi Sugiyama, *Senior Member, IEEE,* Dacheng Tao, *Fellow, IEEE,* Chen Gong, *Senior Member, IEEE*

Abstract—Real-world data may contain a considerable amount of noisily labeled examples, which usually mislead the training algorithm and result in degraded classification performance on test data. Therefore, Label Noise Learning (LNL) was proposed, of which one popular research trend focused on estimating the critical statistics (e.g., sample mean and sample covariance), to recover the clean data distribution. However, existing methods may suffer from the unreliable sample selection process or can hardly be applied to multi-class cases. Inspired by the centroid estimation theory, we propose Per-Class Statistic Estimation (PCSE), which establishes the quantitative relationship between the clean (first-order and second-order) statistics and the corresponding noisy statistics for every class. This relationship is further utilized to induce a generative classifier for model inference. Unlike existing methods, our approach does not require sample selection from the instance level. Moreover, our PCSE can serve as a general post-processing strategy applicable to various popular networks pre-trained on the noisy dataset for boosting their classification performance. Theoretically, we prove that the estimated statistics converge to their ground-truth values as the sample size increases, even if the estimated label transition matrix is biased. Empirically, we conducted intensive experiments on various binary and multi-class datasets, and the results demonstrate that PCSE achieves more precise statistic estimation as well as higher classification accuracy when compared with state-of-the-art methods in LNL.

Index Terms—Label noise, Statistic estimation, Unbiasedness.

1 Introduction

The training of modern machine learning or pattern recognition models usually relies heavily on large-scale datasets with accurate label annotations. However, in many practical applications, acquiring high-quality labels for training data can be challenging due to various subjective or objective factors such as the limitation of human knowledge, the measurement error of instruments, the unreliable automatic labeling processes, etc. The previous study [35] revealed that label noise even exists in many well-curated datasets such as *CIFAR-10*, *CIFAR-100* [24], and *ImageNet* [8]. It has also been shown that deep neural networks can easily be misled by such noisily labeled examples, leading to dramatic performance degradation [54]. Therefore, developing robust Label Noise Learning (LNL) algorithms is highly demanded in various real-world applications.

The existing methods for handling label noise can be roughly classified into three categories, namely sample selection based methods, robust loss function design, and statistic estimation based methods [6], [41]. Among them, sample selection aims to pick up clean examples or remove noisy examples from the original training set. Representative methods include MentorNet [22], Co-teaching [18], and Co-teaching+ [53]. However, most sample selection based methods cannot theoretically guarantee the label correctness of selected clean examples, so they can hardly obtain stable performance in practical uses. Therefore, the second line of research focuses on designing robust loss functions for tackling noisy labels. Representative methods are Generalized Cross Entropy (GCE) [57], Determinant based Mutual Information (DMI) [50], Symmetric Cross Entropy (SCE) [43], and Active Passive Loss (APL) [32]. Nevertheless, the above two types of methods do not explicitly characterize the generation process of the label noise, so they inevitably become weak in some complicated noisy scenarios [6].

The third trend of research is methods based on esti-

- *This research is supported by NSF of China (Nos: 62336003, 12371510, 62376235), NSF of Jiangsu Province (No: BZ2021013), NSF for Distinguished Young Scholar of Jiangsu Province (No: BK20220080), "111" Program (No: B13022), Guangdong Basic and Applied Basic Research Foundation (Nos: 2022A1515011652, 2024A1515012399), HKBU Faculty Niche Research Areas (No: RC-FNRA-IG/22-23/SCI/04), and HKBU CSD Departmental Incentive Grant. This work was done when Wenshui Luo was an intern at RIKEN.*
- *W. Luo and C. Gong are with the Key Laboratory of Intelligent Perception and Systems for High-Dimensional Information of Ministry of Education, 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, 210094, P.R. China. E-mail: {randylo, chen.gong}@njust.edu.cn*
- *S. Chen and G. Niu are with RIKEN Center for Advanced Intelligence Project, Tokyo, Japan. E-mail: shuo.chen.ya@riken.jp; gang.niu.ml@gmail.com*
- *T. Liu is with the School of Computer Science, Faculty of Engineering, the University of Sydney, Australia. E-mail: tongliang.liu@sydney.edu.au*
- *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*
- *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*
- *D. Tao is with Nanyang Technological University, Singapore. E-mail: dacheng.tao@gmail.com*
- *Corresponding authors: C. Gong and S. Chen.*

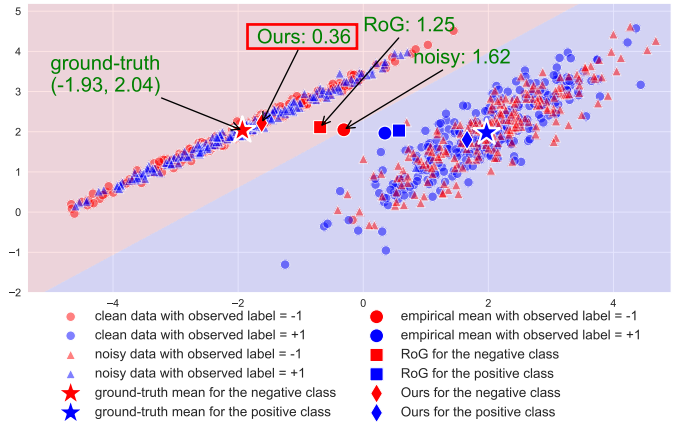


Fig. 1. Sample mean estimations output by RoG and our PCSE, where the empirical mean of the observed noisy dataset is also illustrated. The number of examples is 1000 and the noise rate is 30%. The circles represent clean examples, while the triangles represent noisy examples. The colors of data points indicate the observed labels (i.e., $\widetilde{Y}$ throughout this paper), and the colors of the shaded areas indicate the ground-truth labels of instances. Different markers represent different methods. Their estimation errors regarding positive class based on the Euclidean distance are also shown along the names of different methods. This figure suggests that our PCSE method achieves more accurate estimations on per-class sample mean than RoG.

mation of some statistics. These methods can be further partitioned based on the estimated statistics such as the noise transition matrix [29], [30], [46], [51], [52], dataset centroid [9], [15], [16], [36], and mean/covariance of data [12], [25]. In transition matrix estimation methods, the transition matrix consists of label flip rates of pairwise classes, which approximates the probabilities that each ground-truth label converts to other incorrect ones. Such a matrix can be utilized to estimate the clean class prior [9], [12], [30] and learn statistically consistent classifiers [37], [52]. To this end, plenty of methods have been proposed to estimate the transition matrix, e.g., Importance Reweighting [30], T-revision [46], VolMinNet [29], ROBOT [52], and Total Variation Regularization [56]. For centroid estimation methods, a series of methods [9], [11], [15], [16], [36] decompose the empirical loss into two parts, one of which is label-independent while the other is label-dependent. Then the learning objective is to estimate the centroid of the clean dataset by directly utilizing the noisy centroid and a transition matrix. Among them, Class-Wise Denoising (CWD) [15] proposed a global centroid estimator on the entire training set based on class-wise transformation, which was proved to be statistically more efficient than the estimator of Labeled Instance Centroid Smoothing (LICS) [11].

However, the above methods usually directly estimate one global centroid over the whole training set, so the local statistical property inherited by each class is ignored. To the best of our knowledge, there are only two studies devoted to estimating the unbiased sample mean and covariance. Specifically, Noise Estimation Statistics with Clusters (NESC) [12] proposed unbiased estimators using noisy first- and second-order statistics. However, this approach was only designed for binary classification and can hardly be applied to multi-class cases. Besides, Robust Generative (RoG) classifier [25] assumes that both clean and noisy examples for each class follow the isotropic Gaussian distribution [4], and the noisy examples are more widely scattered than the clean examples. Based on this assumption, the approximate Minimum Covariance Determinant [38] (MCD) estimators were then adopted to estimate class-wise statistics. Nevertheless, the Gaussian distribution assumption cannot be necessarily satisfied in lots of real-world data, and the effectiveness of RoG may depend on the reliability of the sample selection process.

In view of the above problems, we propose a new method called **P**er-**C**lass **S**tatistics **E**stimation (PCSE) to obtain unbiased estimators for per-class statistics in multi-class situations. Inspired by [9], which only estimates a single global centroid on the entire dataset, we further decompose the mathematical expectation of such global centroid into a series of conditional expectations of the mean conditioned on each class. Subsequently, the quantitative relationship between the noisy mean and clean mean (as well as the relationship between the noisy and clean covariances) within each class can be successfully established. Thanks to the broad generality of such a quantitative relationship, our PCSE makes full use of all the examples, and there is no need to identify the clean examples from instance level. Therefore, the unreliable instance selection process widely used in existing LNL approaches [17], [18], [25] can be avoided. We utilized a toy dataset to showcase the superiority of our PCSE in estimating clean statistics compared with RoG and empirical estimations derived directly from the noisy dataset. The sample means estimated by different methods are shown in Fig. 1, where the sample means estimated by PCSE are clearly more accurate than those estimated by other strategies. After obtaining the estimation results for these key statistics, we follow the common practice [25] and also apply Gaussian Discriminant Analysis (GDA) [19] to construct a generative classifier for clean label inference. Similar to RoG [25], our method can also be used as a post-processing strategy for boosting the classification performance of various existing label noise learning methods such as Co-teaching [18].

Theoretically, we derive error upper bounds for the proposed estimators of the first- and second-order statistics, which reveals that our estimation results effectively converge to the corresponding true values with the increase of the sample size. The empirical results across different datasets also demonstrate that the estimation errors of our PCSE are lower than those of NESC and RoG. Finally, the experiments on synthetic and real-world noisy datasets indicate that our PCSE can achieve higher classification accuracy than the state-of-the-art LNL algorithms in both binary and multi-class cases.

In summary, we propose a novel approach called PCSE to obtain unbiased estimators for per-class statistics in the context of multi-class label noise learning. Here, it should be highlighted that our PCSE does not require sample selection, and thereby it is sample-efficient, and it can handle both binary and multi-class noisy scenarios. Moreover, it is provable that the estimated statistics by our method converge to their ground-truth values as the sample size increases, even if the noise transition matrix is slightly biased.

Before delving into details, we highlight the main contributions of our paper in three folds:

- Algorithmically, we propose PCSE, which conducts per-class estimation for both the mean and covariance to fully

characterize the data distribution of different classes.
- Theoretically, we are the first to provide estimation error bounds of per-class statistics for multi-class classification under noisy supervision.
- Empirically, we conducted intensive experiments on synthetic and real-world noisy datasets, which demonstrate that PCSE outperforms existing methods on both classification accuracy and the precision of estimated statistics.

The rest of this paper is organized as follows. In Section 2, we review related works on learning under label noise. In Section 3, we introduce necessary notations and useful preliminary knowledge, which is followed by Section 4 that presents the proposed method in detail. Theoretical analyses and experimental results are provided in Section 5 and Section 6, respectively. Finally, we conclude our paper in Section 7.

2 Related Work

In this section, we review three major types of existing LNL methods, including sample selection based methods, robust loss function design, and statistic estimation based methods.

2.1 Sample Selection Based Methods

Sample selection based methods aim to select clean examples from the noisy dataset for reliable classifier training. Collaborative learning and co-training have been widely adopted by this type of method. Due to the memorization effect [1] of Deep Neural Networks (DNNs), these methods regard examples with small loss values as clean ones during training. A representative approach is Co-teaching [18], which trains two networks collaboratively, and each network exchanges its small-loss examples with its peer network for updating network parameters. After that, Co-teaching+ [53] was proposed, which further employs the disagreement strategy of decoupling when compared with the original Co-teaching. Meanwhile, Joint Training with Co-Regularization (JoCoR) [44] reduces the inconsistency between two networks via the co-regularization technique, and a joint loss is utilized to select small-loss examples so that the error from the biased selection would not be accumulated in a single network. Moreover, some hybrid models were proposed to combine sample selection with other techniques, such as semi-supervised learning [27], data augmentation [34], and label correction [3], [27], further improving the reliability of the sample selection results.

2.2 Robust Loss Function Design

Since sample selection based methods usually fall short of the theoretical guarantee and empirical guidance on clean data selection, they cannot always obtain stable performance on lots of real-world data. Therefore, the second line of research focuses on designing risk-consistent loss functions for tackling noisy labels. For example, Ghosh et al. [14] proposed a sufficient condition for robust loss functions and proved the robustness of the Mean Absolute Error (MAE) loss. However, since MAE treats every example equally (reflected in the gradient), it may incur difficulty in optimization when complicated data are involved. Therefore, Zhang et al. [57] proposed the Generalized Cross Entropy (GCE) loss, which is an extension of Cross Entropy (CE) and MAE. Recently, some other loss functions, such as Active Passive Loss (APL) [32], Taylor-CE [10], Peer Loss [31], and Regularly Truncated M-estimators [47], are also proved to be noise-robust under certain assumptions. However, this type of methods lacks explicit characterization of the generation process of label noise, leading to inherent limitations in addressing complex noise scenarios [6].

2.3 Statistic Estimation Based Methods

The third trend of research can be summarized as the statistic estimation based methods. These methods target to recover clean data distribution by estimating some critical statistics such as transition matrix [5], [29], [30], [46], [51], [52], [56], dataset centroid [9], [15], [16], [36], and the mean/covariance of data [12], [25]. The methods based on transition matrix estimation usually aim to learn statistically consistent classifiers [37], [52], and the critical task is to estimate the transition matrix. To this end, Liu et al. [30] proposed to explore anchor points for calibrations, which are defined as examples that belong to a specified class almost surely. Besides, Li et al. [29] proposed the Sufficiently Scattered Assumption, which is less stringent when compared with the Anchor Point Assumption.

Another line of research estimates some critical statistics regarding the whole dataset, such as the centroid, mean, and covariance, and this type of method is usually combined with the loss factorization technique. For centroid estimation, the common practice is to decompose the loss into two parts, one of which is label-independent while the other one is label-dependent [9], [11], [15], [16], [36]. Then the problem is transformed into that of estimating the centroid of the clean dataset by utilizing the noisy one and a transition matrix. Representative methods include Labeled Instance Centroid Smoothing (LICS) [11], μSGD [36], and Centroid Estimation with Guaranteed Efficiency (CEGE) [16]. However, they mainly focus on the binary classification. In order to handle label noise in the multi-class situation, Ding et al. [9] extended LICS and CEGE by defining a new form of centroid, so that the global centroid over the multi-class dataset can be consequently estimated. Recently, Gong et al. [15] proposed Class-Wise Denoising (CWD), which estimates the centroid of the entire training set by handling the label noise via a class-by-class way. This method was proved to be statistically more efficient than LICS [11]. However, the above methods primarily aim at estimating a single global centroid over the entire training set, so the local statistical property inherited by each class is usually overlooked.

To the best of our knowledge, there are only two existing studies attempting to estimate unbiased per-class statistics (i.e., the sample mean and sample covariance). Specifically, Noise Estimation Statistics with Clusters (NESC) [12] proposed unbiased estimators of the first- and second-order statistics based on the observed noisy data. However, this approach can hardly deal with multi-class cases. Besides, Robust Generative Classifier (RoG) [25] estimated per-class statistics from the perspective of outlier removal. It assumes that both clean and noisy examples for each class obey the isotropic Gaussian distribution, with the noisy examples being more widely scattered than the clean ones. Then, the Minimum Covariance Determinant [38] (MCD) estimators were adopted to estimate the class-wise statistics. However,

TABLE 1
Summary of main mathematical notations.

Notation	Interpretation
$(X, Y), (X, \widetilde{Y})$	A pair of input random variable and the observed contaminated counterpart. Here $X \in \mathcal{X} \subseteq \mathbb{R}^d$ represents the feature and $Y \in \mathcal{Y} = \{0, 1\}^C$ represents the one-hot label, where d is the dimension of feature space and C denotes the number of classes.
$S = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^n$	The unobserved clean sample of (X, Y) with n data points.
$\widetilde{S} = \{(\mathbf{x}_i, \widetilde{\mathbf{y}}_i)\}_{i=1}^n$	The observed noisy sample of $(X, \widetilde{Y})$ with n possible mislabeled training data.
$\pi_i, \widetilde{\pi}_i \in (0, 1)$	The clean and noisy class priors for the i-th class.
$\boldsymbol{\mu}_i \in \mathbb{R}^d$	The sample mean (first-order statistic) for the i-th class.
$\boldsymbol{\sigma}_i \in \mathbb{R}^{d \times d}$	The second-order statistic for the i-th class.
$\boldsymbol{\Sigma}_i \in \mathbb{R}^{d \times d}$	The covariance matrix for the i-th class.
$\mathbf{T} = [T_{ij}]$	The label transition matrix, of which the (i, j)-th element is T_{ij}.

the assumption of the isotropic Gaussian distribution may not hold in many real-world scenarios, and the instance selection process of RoG is not reliable as the selected examples may still contain noisy labels, leading to biased estimation on the mean and covariance.

To tackle the aforementioned issues, we propose a novel method named PCSE to estimate per-class statistics. Our method makes full use of all the training examples when estimating statistics, and there is no need to identify the clean examples at the instance level.

3 PRELIMINARIES

In this section, we first introduce some mathematical notations which will be used in this paper. Specifically, the superscript "$\widetilde{\ }$" indicates that the variable is calculated based on noisy observations, and the variable with a superscript "$\widehat{\ }$" is the corresponding empirical estimation. Note that a statistic accompanied by the term "clean" means that this statistic is calculated using underlying clean labels, whereas a statistic accompanied by the term "noisy" implies that it is calculated using observed noisy labels. We use the notation $[\![K]\!]$ to represent the set $\{1, 2, \cdots, K\}$ for any $K \in \mathbb{Z}$. Besides, the one-hot vector with a value of 1 in its j-th element is denoted by $\mathbf{e}_j$. The mathematical expectation is denoted by $\mathbb{E}[\cdot]$. Here we also introduce some norms used in this paper. Specifically, $\|\mathbf{v}\|_2$ represents the ℓ_2-norm of a vector $\mathbf{v}$, defined as $\|\mathbf{v}\|_2 = \sqrt{\sum_i v_i^2}$, with v_i being the i-th element of $\mathbf{v}$. We use $\|\mathbf{Q}\|_2$ to represent the spectral norm of a matrix $\mathbf{Q}$, which is defined as the largest singular value of $\mathbf{Q}$. Besides, $\text{cond}_2(\mathbf{Q}) = \|\mathbf{Q}\|_2 \|\mathbf{Q}^{-1}\|_2$ [21] means the condition number of a nonsingular matrix $\mathbf{Q}$. Unless otherwise stated, the spectral norm is used throughout this paper for the calculation of the condition number. For a matrix $\mathbf{Q}$, the norm $\|\mathbf{Q}\|_1$ is defined as $\|\mathbf{Q}\|_1 = \max_j \sum_i |Q_{ij}|$ with Q_{ij} being the (i, j)-th element of $\mathbf{Q}$. Additionally, we use $\mathbf{1}\{\cdot\}$ to denote the indicator function. Here $\mathbf{1}\{\cdot\} = 1$ if and only if the event within the bracket is satisfied. The main mathematical notations that will be later used for algorithm description are listed in Table 1.

We consider a typical classification problem with C classes of data. Let $\mathcal{X} \subseteq \mathbb{R}^d$ and $\mathcal{Y}$ be the input feature space and output label space, respectively, where d denotes the dimension of features and $\mathcal{Y} = \{0, 1\}^C$. We define the clean joint distribution of a pair of random variables $(X, Y) \in \mathcal{X} \times \mathcal{Y}$ as $\mathcal{D}$. The clean sample set $S = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^n$ of (X, Y) containing n data points is drawn identically and independently from $\mathcal{D}$, where the one-hot vector $\mathbf{y}_i$ is the ground-truth label of $\mathbf{x}_i$. However, under label noise learning, we are only accessible to a sample of n independent and identically distributed data points $\widetilde{S} = \{(\mathbf{x}_i, \widetilde{\mathbf{y}}_i)\}_{i=1}^n$ from a noisy distribution $\widetilde{\mathcal{D}}$ of random variables $(X, \widetilde{Y}) \in \mathcal{X} \times \mathcal{Y}$, where $\widetilde{Y}$ is the contaminated version of Y. Let $\widetilde{n}_c$ be the number of examples for the c-th class in $\widetilde{S}$. The task of LNL is to learn a robust classifier by leveraging $\widetilde{S}$ that can approximate the optimal classifier trained on S.

In this paper, we consider the class-dependent label noise, which is a widely used setting in label noise learning [30], [46], [56]. In this setting, the observed noisy label for each $\mathbf{x} \in \mathcal{X}$ only depends on its underlying clean label. To be more specific, the transition probability of class i to class j is $P(\widetilde{Y} = \mathbf{e}_j | Y = \mathbf{e}_i, X = \mathbf{x}) = P(\widetilde{Y} = \mathbf{e}_j | Y = \mathbf{e}_i) = T_{ij}, \forall i, j \in [\![C]\!]$, where $\mathbf{T} = [T_{ij}] \in [0, 1]^{C \times C}$ is the noise transition matrix. We denote the noise rate by ϵ, then for symmetric label noise and $\forall i, j \in [\![C]\!]$, $T_{ii} = 1 - \epsilon$ and $T_{ij} = \frac{\epsilon}{C-1}$ with $j \neq i$. We can estimate the transition matrix by solving the following problem [29]:

$$\min_{\boldsymbol{\theta}, \widehat{\mathbf{T}}} L(\boldsymbol{\theta}, \widehat{\mathbf{T}}) = \frac{1}{n} \sum_{i=1}^{n} \ell(\widehat{\mathbf{T}}^\top h_{\boldsymbol{\theta}}(\mathbf{x}_i), \widetilde{\mathbf{y}}_i) + \lambda \cdot \log\det(\widehat{\mathbf{T}}), \quad (1)$$

where ℓ is the loss function (cross-entropy loss is typically used). The function $h_{\boldsymbol{\theta}}(\cdot)$ is the output of a neural network parameterized by $\boldsymbol{\theta}$. The regularizer $\log\det(\widehat{\mathbf{T}})$ stands for the natural logarithm of determinant of $\widehat{\mathbf{T}}$, which ensures that the simplex formed by $\widehat{\mathbf{T}}$ has the minimum volume, and $\lambda > 0$ is the trade-off hyperparameter.

Before formally introducing our proposed algorithm, here we first discuss two related methods which also focus on statistic estimation.

3.1 Loss Decomposition and Centroid Estimation

Loss Decomposition and Centroid Estimation (LDCE) [11], [15], [36] has been used as an effective technique to construct unbiased loss functions by leveraging noisy training set $\widetilde{S}$. For multi-class cases, Ding et al. [9] proposed an extension of LDCE (termed "MC-LDCE"), by defining a generalized form of data centroid. Suppose that we use the linear scoring function $h_{\mathbf{W}}(\mathbf{x}) = \mathbf{W}^\top \mathbf{x}$ parameterized by $\mathbf{W} \in \mathbb{R}^{d \times C}$, and use the ℓ_2 loss as the loss function. Then the empirical risk over the clean set S can be formulated by

$$\widehat{\mathcal{R}}(h, S) = \frac{1}{n} \sum_{i=1}^{n} \left\| \mathbf{y}_i - \mathbf{W}^\top \mathbf{x}_i \right\|_2^2. \quad (2)$$

This loss function can be further decomposed to the sum of a label-independent term and a label-dependent term, namely

$$\widehat{\mathcal{R}}(h, S) = \underbrace{\left(1 + \frac{1}{n} \mathbf{x}_i^\top \mathbf{W} \mathbf{W}^\top \mathbf{x}_i\right)}_{\text{label-independent term}} \underbrace{- 2 \cdot \text{trace}\big(\mathbf{W}^\top \widehat{\boldsymbol{\mu}}(S)\big)}_{\text{label-dependent term}}, \quad (3)$$

where $\widehat{\boldsymbol{\mu}}(S)=\frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i\mathbf{y}_i^\top \in \mathbb{R}^{d\times C}$ is dubbed as the empirical clean centroid, and its expectation is $\mathbb{E}_{(X,Y)\sim\mathcal{D}}[XY^\top]$, which is called clean centroid. We use $\mathrm{trace}(\mathbf{M})$ to denote the trace of a square matrix $\mathbf{M}$, which is defined as the sum of all diagonal elements of $\mathbf{M}$. Note that we do not have access to the clean set S and thus we cannot obtain $\widehat{\boldsymbol{\mu}}(S)$. Instead, we can derive the empirical noisy centroid $\widehat{\boldsymbol{\mu}}(\widetilde{S})=\frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i\widetilde{\mathbf{y}}_i^\top$ based on $\widetilde{S}$. The corresponding expectation $\mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}[X\widetilde{Y}^\top]$ is called noisy centroid. To obtain the unbiased ℓ_2 loss based only on $\widetilde{S}$, we need to find an unbiased estimator of the clean centroid according to the noisy centroid $\widehat{\boldsymbol{\mu}}(\widetilde{S})$.

Since the relationship between the clean centroid and noisy centroid is crucial for our proposed method, here we briefly outline the derivation process. MC-LDCE begins with studying the conditional expectation of $X\widetilde{Y}^\top$ over the noisy set, which is formulated as

$$\mathbb{E}_{\widetilde{Y}}[X\widetilde{Y}^\top|(X,Y)]=\sum_{i=1}^{C}\pi_i\mathbb{E}_{\widetilde{Y}}[X\widetilde{Y}^\top|(X,Y=\mathbf{e}_i)], \tag{4}$$

where $\pi_i = P(Y=\mathbf{e}_i)$ is the class prior for the i-th class. For the one-hot vectors $\mathbf{e}_i$ and $\mathbf{e}_j$, we can use a permutation matrix $\mathbf{K}_{i\to j}$ to convert $\mathbf{e}_i$ to $\mathbf{e}_j$, namely $\mathbf{e}_j=\mathbf{K}_{i\to j}\mathbf{e}_i$. Here, the permutation matrix $\mathbf{K}_{i\to j}$ is constructed by exchanging the i-th and the j-th rows of an identity matrix. This equation allows us to calculate the conditional expectation as

$$\mathbb{E}_{\widetilde{Y}}[X\widetilde{Y}^\top|(X,Y=\mathbf{e}_i)]=\sum_{j=1}^{C}T_{ij}XY^\top\mathbf{K}_{i\to j}^\top. \tag{5}$$

Therefore, the conditional expectation in Eq. (4) can be further transformed to

$$\mathbb{E}_{\widetilde{Y}}[X\widetilde{Y}^\top|(X,Y)]=XY^\top\underbrace{\left[\sum_{i=1}^{C}\pi_i\sum_{j=1}^{C}T_{ij}\mathbf{K}_{i\to j}^\top\right]}_{\mathbf{M}}, \tag{6}$$

where we define $\mathbf{M}=\sum_{i=1}^{C}\sum_{j=1}^{C}\pi_iT_{ij}\mathbf{K}_{i\to j}^\top$, with M_{ij} being its (i,j)-th element. Then, the relationship between the clean global centroid and noisy global centroid can be directly derived from Eq. (6), which is given by

$$\mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}[X\widetilde{Y}^\top]\mathbf{M}^{-1}=\mathbb{E}_{(X,Y)\sim\mathcal{D}}[XY^\top], \tag{7}$$

where we use the fact that $\mathbf{M}$ does not depend on the clean distribution $\mathcal{D}$. This relationship is then used to obtain $\widehat{\boldsymbol{\mu}}(S)=\widehat{\boldsymbol{\mu}}(\widetilde{S})\mathbf{M}^{-1}$, which will be further substituted into Eq. (3) for recovering the original empirical risk on the clean set S. Therefore, we only need noisy labels (instead of the unobservable clean ones) to construct the unbiased ℓ_2 loss.

Note that in [9], the invertibility of $\mathbf{M}$ is not proved and thus they use the pseudo-inverse $\mathbf{M}^\dagger$ instead in the above derivations. However, in this paper, we theoretically reveal that its invertibility can be guaranteed by Lemma 1 under mild conditions.

Lemma 1. ***(Invertibility of matrix*** $\mathbf{M}$***)*** *Let* $\mathbf{M}=\sum_{i=1}^{C}\sum_{j=1}^{C}\pi_iT_{ij}\mathbf{K}_{i\to j}$. *If one of the following conditions is satisfied, then* $\mathbf{M}$ *is invertible:*

1) *For symmetric label noise, the noise rate* $\epsilon<\min\left\{\frac{C-1}{\max_{i\in[\![C]\!]}\pi_i(2C-4)+2},\frac{C-1}{C}\right\}$.
2) *For asymmetric label noise, the class priors are uniform, i.e.,* $\pi_i=\frac{1}{C},\forall i\in[\![C]\!]$, *and the noise rate* $\epsilon<\frac{1}{2}$.

The proof of Lemma 1 has been put to the **supplementary material**. Note that for the first condition, if $\max_{i\in[\![C]\!]}\pi_i\le\frac{1}{2}$, then $\max_{i\in[\![C]\!]}\pi_i(2C-4)+2<C$, and thus the condition for invertibility of $\mathbf{M}$ becomes $\epsilon<\frac{C-1}{C}$, which is a commonly used assumption in LNL [10], [14], [57].

3.2 Robust Generative Classifier

Robust Generative Classifier [25] (RoG) is a label inference method that can be regarded as a general post-processing strategy applicable to many robust classifiers pre-trained on noisy datasets. It induces a generative classifier on top of hidden feature spaces of the pre-trained DNNs, for obtaining a more robust decision boundary with boosted classification accuracy.

We denote the sample means w.r.t. the ground-truth labels by

$$\boldsymbol{\mu}_i=\mathbb{E}_{X|Y=\mathbf{e}_i}[X],\quad \forall i\in[\![C]\!], \tag{8}$$

which is also known as the first-order statistic for each class. The covariance matrix w.r.t. the ground-truth labels for the i-th class is denoted by

$$\boldsymbol{\Sigma}_i=\mathbb{E}_{X|Y=\mathbf{e}_i}[(X-\boldsymbol{\mu}_i)(X-\boldsymbol{\mu}_i)^\top]. \tag{9}$$

The empirical sample mean and covariance for the i-th class are denoted by $\widehat{\boldsymbol{\mu}}_i$ and $\widehat{\boldsymbol{\Sigma}}_i$, respectively.

To estimate per-class statistics, RoG constructs the set $\widetilde{S}_i=\{\mathbf{x}_j\,|\,(\mathbf{x}_j,\widetilde{\mathbf{y}}_j)\in\widetilde{S},\,\widetilde{\mathbf{y}}_j=\mathbf{e}_i,\,j\in[\![n]\!]\}$, which contains the examples of which the observed noisy labels are $\mathbf{e}_i$. Then RoG proposes to use Minimum Covariance Discriminant (MCD) method to select clean examples for statistics estimation. Specifically, the MCD estimator considers noisy examples as outliers and removes them via a class-by-class way. For the i-th class, this method seeks for a subset $\widetilde{S}_i^{\text{sub}}$ from $\widetilde{S}_i$ by the following objective

$$\min_{\widetilde{S}_i^{\text{sub}}\subseteq\widetilde{S}_i}\det(\widehat{\boldsymbol{\Sigma}}_i)\quad\text{subject to}\quad|\widetilde{S}_i^{\text{sub}}|=K_i, \tag{10}$$

where the constant K_i is the number of selected examples for class i, and $\widehat{\boldsymbol{\Sigma}}_i$ is the covariance matrix for examples in $\widetilde{S}_i^{\text{sub}}$. Once the optimal $\widetilde{S}_i^{\text{sub}}$ is obtained, the examples in this set will be used to calculate the sample mean and covariance for class i directly, while the excluded examples in $\widetilde{S}_i$ will be perceived as outliers or noisy examples.

Subsequently, RoG employs Gaussian Discriminant Analysis (GDA) [19] to train a generative classifier. Actually, the output of the l-th layer ϕ_l of a DNN can be regarded as the hidden-layer features for the examples in $\widetilde{S}$. RoG induces a generative classifier by assuming that $\phi_l(\mathbf{x})$ (conditioned on Y) obeys multivariate Gaussian distributions with the identical covariance across all classes, i.e., $P(\phi_l(\mathbf{x})|Y=\mathbf{e}_c)=\mathcal{N}(\phi_l(\mathbf{x})|\boldsymbol{\mu}_c^{(l)},\boldsymbol{\Sigma}^{(l)})$, where $\boldsymbol{\Sigma}^{(l)}$ is the shared covariance based on the assumptions of GDA. Then based on the Bayesian rule, for a test example $\mathbf{x}$, the class posterior is

$$\begin{aligned}P(Y=\mathbf{e}_c|\phi_l(\mathbf{x}))&=\frac{\widehat{\pi}_cP(\phi_l(\mathbf{x})|Y=\mathbf{e}_c)}{\sum_{c'}\widehat{\pi}_{c'}P(\phi_l(\mathbf{x})|Y=\mathbf{e}_{c'})}\\&=\frac{\exp(\widehat{\boldsymbol{\mu}}_c^\top\widehat{\boldsymbol{\Sigma}}^{-1}\phi_l(\mathbf{x})-\frac{1}{2}\widehat{\boldsymbol{\mu}}_c\widehat{\boldsymbol{\Sigma}}^{-1}\widehat{\boldsymbol{\mu}}_c+\log\widehat{\pi}_c)}{\sum_{c'}\exp(\widehat{\boldsymbol{\mu}}_{c'}^\top\widehat{\boldsymbol{\Sigma}}^{-1}\phi_l(\mathbf{x})-\frac{1}{2}\widehat{\boldsymbol{\mu}}_{c'}\widehat{\boldsymbol{\Sigma}}^{-1}\widehat{\boldsymbol{\mu}}_{c'}+\log\widehat{\pi}_{c'})},\end{aligned} \tag{11}$$

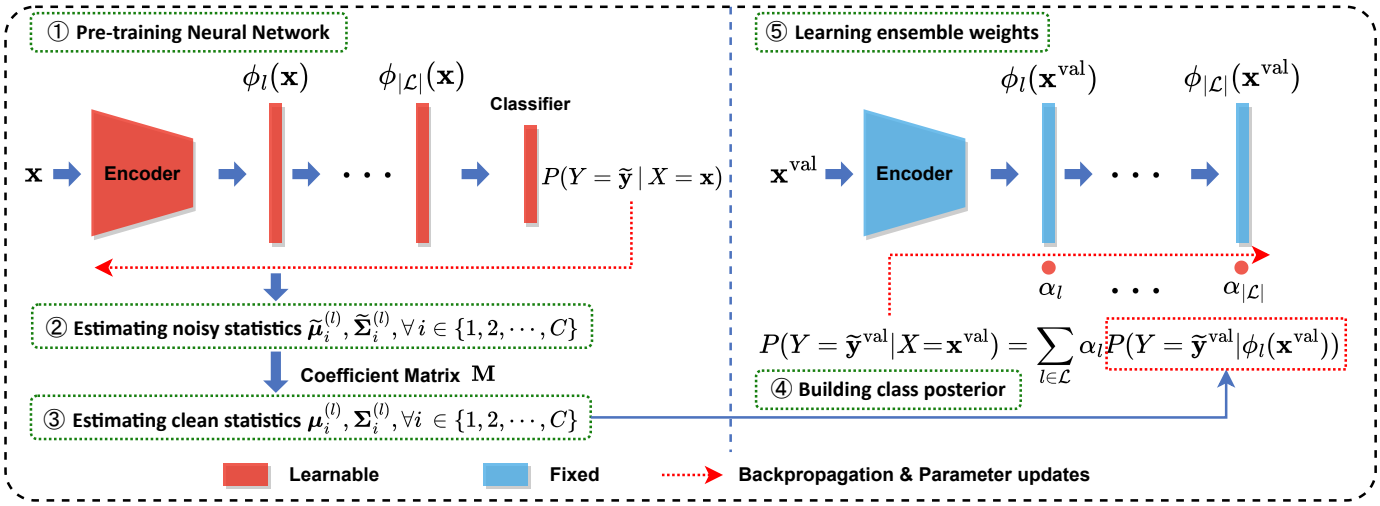


Fig. 2. The overall framework of our PCSE method. In the framework, PCSE first pre-trains the neural network using any existing method (①), and the hidden-layer features generated by this network are utilized to calculate the noisy per-class statistics $\widetilde{\boldsymbol{\mu}}_i^{(l)}$ and $\widetilde{\boldsymbol{\Sigma}}_i^{(l)}, \forall i \in [\![C]\!]$ (②). Subsequently, the noisy statistics and the coefficient matrix $\mathbf{M}$ are employed to estimate per-class clean statistics $\boldsymbol{\mu}_i^{(l)}$ and $\boldsymbol{\Sigma}_i^{(l)}, \forall i \in [\![C]\!]$ (③), which are then utilized to build the class posterior $P(Y=\tilde{\mathbf{y}}\,|\,\phi_l(\mathbf{x}))$ (④). Finally, the examples $(\mathbf{x}^{\text{val}}, \tilde{\mathbf{y}}^{\text{val}})$ from the noisy validation set $\widetilde{S}_{\text{val}}$ are used to learn the ensemble weights, and the generative classifier induced by $\{P(Y\,|\,\phi_l(\mathbf{x}))\}_{l\in\mathcal{L}}$ takes the place of the pre-trained classifier for model inference (⑤).

where $\widehat{\boldsymbol{\Sigma}} = \sum_i \widehat{\pi}_i \widehat{\boldsymbol{\Sigma}}_i$ is the overall covariance. Here uniform class priors are assumed, which means $\widehat{\pi}_i = \frac{1}{C}$ for all $i \in [\![C]\!]$. Note that for brevity, we omit the superscript "(l)" for $\widehat{\boldsymbol{\mu}}_c$ and $\widehat{\boldsymbol{\Sigma}}$ in Eq. (11). To further boost performance, RoG proposes to use an ensemble version of generative classifiers. To this end, RoG computes the estimated clean means and covariances for hidden-layer features from several specified layers. Then the final clean class posterior is induced as the weighted sum of posterior distributions for these layers, namely

$$P(Y=\mathbf{e}_c|X=\mathbf{x}) = \sum_{l\in\mathcal{L}} \alpha_l P(Y=\mathbf{e}_c|\phi_l(\mathbf{x})), \tag{12}$$

where α_l is the ensemble weight for the l-th layer and $\mathcal{L}$ is the set of selected layers. Here the weights satisfy $\sum_{l\in\mathcal{L}} \alpha_l = 1$ and $\alpha_l > 0, \forall l \in \mathcal{L}$. To learn these weights, RoG optimizes the Negative Log-likelihood Loss (NLL) over a noisy validation set $\widetilde{S}_{\text{val}}$. Actually, some other loss functions such as logistic loss and mean squared loss, can also be adopted here, and the main difference lies in the optimization process, where they will have different gradients for iterations.

However, RoG has certain limitations in real-world applications as mentioned in Section 1. Moreover, it is empirical to decide how many examples to be selected for each class, so it is still challenging to obtain accurate estimations in some scenarios with complicated noise.

4 Our Proposed Method

In this section, we provide a new method to obtain unbiased estimators of noise-free statistics. Different from MCLDCE [9] which estimates a single global statistic over the entire training set, our method unbiasedly estimates the statistics via a class-wise way.

The framework of our method is presented in Fig. 2. As shown in this figure, our target is to establish the relationship between the first-/second-order statistics of noisy data and clean data in each class. Subsequently, we will utilize this relationship to derive unbiased estimators for the means and covariances of clean sample. After that, we leverage these statistics to induce a generative classifier by using Eqs. (11) and (12) in RoG [25]. Therefore, we find that in this process, the critical step is to accurately estimate the mean $\{\widehat{\boldsymbol{\mu}}_c\}_{c=1}^C$ and covariance $\widehat{\boldsymbol{\Sigma}}$ in Eqs. (11) and (12). To fulfill such estimation, here we need the following assumptions.

Assumption 1. *(**Sufficiently Scattered Assumption** [29]). The clean class posterior $\mathbf{P}(Y|X) = [P(Y=\mathbf{e}_1|X), P(Y=\mathbf{e}_2|X), \cdots, P(Y=\mathbf{e}_C|X)]^\top \in [0,1]^C$ is said to be sufficiently scattered if there exists a set $\mathcal{H} = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_m\}$ such that the matrix $\mathbf{H} = [\mathbf{P}(Y|X=\mathbf{x}_1), \cdots, \mathbf{P}(Y|X=\mathbf{x}_m)]$ satisfies: (1) $\mathcal{Q} \subseteq cone\{\mathbf{H}\}$, where $\mathcal{Q} = \{\mathbf{v} \in \mathbb{R}^C | \mathbf{v}^\top \mathbf{1} \geq \sqrt{C-1}\|\mathbf{v}\|_2\}$ and $cone\{\mathbf{H}\}$ denotes the convex cone combined by the columns of $\mathbf{H}$; and (2) $cone\{\mathbf{H}\} \not\subseteq cone\{\mathbf{U}\}$ for any unitary matrix $\mathbf{U} \in \mathbb{R}^{C\times C}$ that is not a permutation matrix.*

Assumption 2. *(**Nonsingular** $\mathbf{T}$). The noise transition matrix is nonsingular, i.e., $Rank(\mathbf{T}) = C$.*

Assumption 1 requires the clean class posteriors to be sufficiently scattered in the probability simplex so that the ground-truth transition matrix can be identified. Assumption 2 is a widely adopted constraint in the literature [42], [59], which ensures the invertibility of the transition matrix. To verify these assumptions, we provide detailed experiments and analyses in the **supplementary material**. Additionally, we need to assume that the DNN representations still exhibit *clustering properties* (i.e., distinguishable for different classes) when the DNN is trained with noisy labels. This is because that the lower-level features are usually not significantly influenced by the higher-level supervision. Actually, this assumption has been demonstrated by some recent studies [25], [26], [55], [59], where it is identified that even though label noise may mislead the final classification results, it can still induce good feature representations.

Estimating per-class sample mean and covariance is a challenging task as the underlying clean label for each example is typically unknown. However, the noisy statistics for each class can be easily estimated, forming the

foundation of our approach. We denote the noisy sample means for all classes by $\widetilde{\boldsymbol{\mu}}_i = \mathbb{E}_{X|\widetilde{Y}=\mathbf{e}_i}[X], \forall i \in [\![C]\!]$, which can be readily estimated empirically based on $\widetilde{S}$. Additionally, $\boldsymbol{\sigma}_i = \mathbb{E}_{X|Y=\mathbf{e}_i}[XX^\top]$ stands for the second-order statistics of the i-th class. The corresponding contaminated counterpart is $\widetilde{\boldsymbol{\sigma}}_i$. The covariance matrix w.r.t. the ground-truth labels for the i-th class (defined in Eq. (9)) can be constructed based on the first- and second-order statistics, namely $\boldsymbol{\Sigma}_i = \boldsymbol{\sigma}_i - \boldsymbol{\mu}_i\boldsymbol{\mu}_i^\top$.

Subsequently, our task boils down to constructing unbiased estimators of $\{\boldsymbol{\mu}_i\}_{i=1}^C$ and $\{\boldsymbol{\Sigma}_i\}_{i=1}^C$ via leveraging $\{\widetilde{\boldsymbol{\mu}}_i\}_{i=1}^C$ and $\{\widetilde{\boldsymbol{\sigma}}_i\}_{i=1}^C$. Thanks to the global centroid estimation method briefly revisited in Section 3, we can derive the relationship between $\boldsymbol{\mu}_i$, $\boldsymbol{\Sigma}_i$ and $\widetilde{\boldsymbol{\mu}}_i$, $\widetilde{\boldsymbol{\sigma}}_i$. Note that for [9] and [15], the aim of global centroid estimation is to construct an unbiased ℓ_2 loss. By contrast, we further develop the centroid estimation method for per-class statistic estimation to consider local properties within each class.

First of all, the relationship between the clean global centroid and noisy global centroid has already been provided in Eq. (7). Next, we will utilize this equation to derive the relationship between clean and noisy sample means. We define $\mathbf{U} = [\boldsymbol{\mu}_1, \cdots, \boldsymbol{\mu}_C]$, which contains the per-class sample mean. The noisy counterpart is defined as $\widetilde{\mathbf{U}} = [\widetilde{\boldsymbol{\mu}}_1, \cdots, \widetilde{\boldsymbol{\mu}}_C]$. We factorize the clean global centroid $\mathbb{E}_{(X,Y)\sim\mathcal{D}}[XY^\top]$ as follows:

$$\begin{aligned}\mathbb{E}_{(X,Y)\sim\mathcal{D}}[XY^\top] &= \mathbb{E}_Y\mathbb{E}_{X|Y}[XY^\top] \\ &= [\pi_1\mathbb{E}_{X|Y=\boldsymbol{e}_1}[X], \cdots, \pi_C\mathbb{E}_{X|Y=\boldsymbol{e}_C}[X]] \\ &= [\pi_1\boldsymbol{\mu}_1, \pi_2\boldsymbol{\mu}_2, \cdots, \pi_C\boldsymbol{\mu}_C] \\ &= \mathbf{U}\boldsymbol{\Lambda},\end{aligned} \tag{13}$$

where $\boldsymbol{\Lambda} = diag([\pi_1, \cdots, \pi_C])$ and $diag(\cdot)$ outputs a diagonal matrix of which the diagonal elements are filled by the elements of the input vector. Similarly, we have $\mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}[X\widetilde{Y}] = \widetilde{\mathbf{U}}\widetilde{\boldsymbol{\Lambda}}$, where $\widetilde{\boldsymbol{\Lambda}} = diag([\widetilde{\pi}_1, \cdots, \widetilde{\pi}_C])$ is the noisy version of $\boldsymbol{\Lambda}$. As a sequel, by leveraging Eq. (7), we obtain

$$\mathbf{U} = \widetilde{\mathbf{U}}\widetilde{\boldsymbol{\Lambda}}\mathbf{M}^{-1}\boldsymbol{\Lambda}^{-1}. \tag{14}$$

This implies that for any $i = 1, 2, \cdots, C$, we have

$$\boldsymbol{\mu}_i = \sum_{j=1}^{C} \frac{\widetilde{\pi}_j}{\pi_i}\mathbf{M}_{ji}^{-1}\widetilde{\boldsymbol{\mu}}_j, \tag{15}$$

which establishes the relationship between the clean sample mean and noisy sample mean (or the first-order statistics). Similarly, we can derive the relationship between the clean and noisy second-order statistics, namely

$$\boldsymbol{\sigma}_i = \sum_{j=1}^{C} \frac{\widetilde{\pi}_j}{\pi_i}\mathbf{M}_{ji}^{-1}\widetilde{\boldsymbol{\sigma}}_j, \quad \forall i \in [\![C]\!]. \tag{16}$$

From Eq. (15) and (16), we can find that we need to empirically estimate each term (i.e., π_i, $\widetilde{\pi}_i$, $\mathbf{M}$, $\widetilde{\boldsymbol{\mu}}_i$, and $\widetilde{\boldsymbol{\sigma}}_i$) in the right-hand side of both equations so as to obtain the unbiased estimators of $\boldsymbol{\mu}_i$ and $\boldsymbol{\sigma}_i$. To this end, first of all, we need to estimate the transition matrix $\mathbf{T}$ by solving Eq. (1), which is denoted by $\widehat{\mathbf{T}}$. Based on Assumption 1, $\widehat{\mathbf{T}}$ will converge to the ground-truth $\mathbf{T}$ given the sufficient noisy data (Theorem 1 in [29]). Subsequently, we need to empirically estimate clean class priors $\boldsymbol{\pi} = [\pi_1, \pi_2, \cdots, \pi_C]^\top$ so as to estimate the matrix $\mathbf{M}$. According to the common practice in previous works [9], [15], estimating $\boldsymbol{\pi}$ is equivalent to solving the following linear system of equations:

$$\begin{cases} \widetilde{\pi}_1 = T_{11}\pi_1 + T_{21}\pi_2 + \cdots + T_{C1}\pi_C \\ \widetilde{\pi}_2 = T_{12}\pi_1 + T_{22}\pi_2 + \cdots + T_{C2}\pi_C \\ \qquad\qquad \vdots \\ \widetilde{\pi}_C = T_{1C}\pi_1 + T_{2C}\pi_2 + \cdots + T_{CC}\pi_C \end{cases}, \tag{17}$$

where $\widetilde{\pi}_i = P(\widetilde{Y} = \mathbf{e}_i)$ is the noisy class prior for the i-th class. Here $\widetilde{\pi}_i$ can be empirically estimated by

$$\widehat{\widetilde{\pi}}_i = \frac{\sum_{j=1}^{n}\mathbf{1}\{\widetilde{\mathbf{y}}_j = \mathbf{e}_i\}}{n}, \quad \forall i \in [\![C]\!]. \tag{18}$$

Since $\mathbf{T}$ is assumed to be nonsingular in Assumption 2, we can obtain a unique solution by solving Eq. (17), which is denoted by $\widehat{\boldsymbol{\pi}} = [\widehat{\pi}_1, \widehat{\pi}_2, \cdots, \widehat{\pi}_C] \in (0, 1)^C$. Based on this, the empirical estimation of matrix $\mathbf{M}$ can be calculated as

$$\widehat{\mathbf{M}} = \sum_{i=1}^{C}\sum_{j=1}^{C}\widehat{\pi}_i\widehat{T}_{ij}\mathbf{K}_{i\to j}^\top. \tag{19}$$

For the contaminated statistics $\widetilde{\boldsymbol{\mu}}_i$ and $\widetilde{\boldsymbol{\sigma}}_i$, they can be directly estimated based on the noisy dataset $\widetilde{S}$. Specifically, we use a DNN to extract the features of the given inputs. Let $\phi_l(\mathbf{x}) \in \mathbb{R}^{d_l}$ be the output of the l-th layer of a DNN given input $\mathbf{x}$, where d_l is the feature dimensionality. For brevity, we omit the superscript "(l)" in the following equations. For

Algorithm 1 **P**er-**C**lass **S**tatistics **E**stimation (PCSE) for multi-class label noise learning.

Input: Noisy training set $\widetilde{S} = \{(\mathbf{x}_i, \widetilde{\mathbf{y}}_i)\}_{i=1}^n$; noisy validation set $\widetilde{S}_{\text{val}} = \{(\mathbf{x}_i^{\text{val}}, \widetilde{\mathbf{y}}_i^{\text{val}})\}_{i=1}^{n_{\text{val}}}$; neural network $f_{\boldsymbol{\theta}}$; and the set of layers $\mathcal{L}$.
Initialize: Set the learnable weight $\alpha_l = 1/|\mathcal{L}|$ for $\forall l \in \mathcal{L}$; and pre-train neural network $f_{\boldsymbol{\theta}}$ with $\widetilde{S}$ using any existing method (e.g. Co-teaching [18], JoCoR [44]);
Obtain the transition matrix $\widehat{\mathbf{T}}$ by solving Eq. (1);
Compute noisy class priors $\{\widehat{\widetilde{\pi}}_i\}_{i=1}^C$ via Eq. (18);
Compute clean class priors $\{\widehat{\pi}_i\}_{i=1}^C$ by solving Eq. (17);
Compute $\widehat{\mathbf{M}}$ via Eq. (19);
For all l in $\mathcal{L}$ **do**
  Extract features $\{\phi_l(\mathbf{x}_i)\}_{i=1}^n$ for examples in $\widetilde{S}$;
  For $i = 1$ to C **do**
    Compute the empirical noisy first-order statistic $\widehat{\widetilde{\boldsymbol{\mu}}}_i$ and second-order statistic $\widehat{\widetilde{\boldsymbol{\sigma}}}_i$ via Eq. (20);
    Compute the empirical clean sample mean $\widehat{\boldsymbol{\mu}}_i$ and clean covariance $\widehat{\boldsymbol{\Sigma}}_i$ via Eqs. (21), (22) and (23);
  End For
  Compute the class posterior $P(Y = \widetilde{\mathbf{y}}_i^{\text{val}}|\phi_\ell(\mathbf{x}_i^{\text{val}}))$ for all $(\mathbf{x}_i^{\text{val}}, \widetilde{\mathbf{y}}_i^{\text{val}})$ in $\widetilde{S}_{\text{val}}$ via Eq. (11);
End For
Learn the weights $\{\alpha_l\}_{l\in\mathcal{L}}$ in Eq. (12) by optimizing the Negative Log-likelihood Loss over $\widetilde{S}_{\text{val}}$.
Output: The optimal parameters $\boldsymbol{\theta}^*$ for the pre-trained neural network and the optimal ensemble weights $\{\alpha_l^*\}_{l\in\mathcal{L}}$.

the i-th class, we denote the empirical estimations of $\widetilde{\boldsymbol{\mu}}_i$ and $\widetilde{\boldsymbol{\sigma}}_i$ by $\widehat{\widetilde{\boldsymbol{\mu}}}_i$ and $\widehat{\widetilde{\boldsymbol{\sigma}}}_i$, respectively, which are given by

$$\begin{aligned}\widehat{\widetilde{\boldsymbol{\mu}}}_i &= \frac{1}{\widetilde{n}_i}\sum_{j\in[\![n]\!],\widetilde{\mathbf{y}}_j=\mathbf{e}_i}\phi_l(\mathbf{x}_j),\\ \widehat{\widetilde{\boldsymbol{\sigma}}}_i &= \frac{1}{\widetilde{n}_i}\sum_{j\in[\![n]\!],\widetilde{\mathbf{y}}_j=\mathbf{e}_i}\phi_l(\mathbf{x}_j)\phi_l(\mathbf{x}_j)^\top,\end{aligned} \tag{20}$$

where $\widetilde{n}_i$ is the number of examples for the i-th class in $\widetilde{S}$. Subsequently, these empirical estimations can be used to calculate the unbiased estimators of $\boldsymbol{\mu}_i$ and $\boldsymbol{\sigma}_i$, which are

$$\widehat{\boldsymbol{\mu}}_i = \sum_{j=1}^{C}\frac{\widehat{\widetilde{\pi}}_j}{\widehat{\widetilde{\pi}}_i}\widehat{\mathbf{M}}_{ji}^{-1}\widehat{\widetilde{\boldsymbol{\mu}}}_j, \tag{21}$$

and

$$\widehat{\boldsymbol{\sigma}}_i = \sum_{j=1}^{C}\frac{\widehat{\widetilde{\pi}}_j}{\widehat{\widetilde{\pi}}_i}\widehat{\mathbf{M}}_{ji}^{-1}\widehat{\widetilde{\boldsymbol{\sigma}}}_j, \tag{22}$$

respectively. The unbiased estimator of class-wise covariance can also be derived, which is

$$\widehat{\boldsymbol{\Sigma}}_i = \widehat{\boldsymbol{\sigma}}_i - \widehat{\boldsymbol{\mu}}_i\widehat{\boldsymbol{\mu}}_i^\top, \quad \forall i \in [\![C]\!]. \tag{23}$$

Then, the overall covariance under clean distribution can be estimated as $\widehat{\boldsymbol{\Sigma}} = \sum_i \widehat{\pi}_i\widehat{\boldsymbol{\Sigma}}_i$. By substituting these estimated noise-free statistics into Eq. (11) and (12), the clean class posterior can be inferred. The main steps of our PCSE are summarized in Algorithm 1.

Remark 1. Our PCSE is different from MC-LDCE [9] and CWD [15] in that PCSE focuses more on the local information of dataset while MC-LDCE and CWD build unbiased loss functions based on the estimation of dataset centroid. Besides, our theoretical results differ from those of CWD in that we focus on the estimation error bounds of per-class statistics while CWD aims to provide guarantees on the generalization error bound. Such estimation error bounds successfully ensure the practical effectiveness of our proposed estimators by revealing the critical factors that substantially affect estimation performance.

Remark 2. It is worth noting that our PCSE can not only be applied to LNL with a single ground-truth label, it can potentially be adapted to the problem of learning with class-conditional multi-label noise [49] as well. Since such an extension is beyond the scope of this paper, we provide a brief discussion in the **supplementary material**.

5 Theoretical Analyses

In this section, we provide the error bounds of our PCSE in estimating the per-class noise-free statistics (Section 5.1). Moreover, we establish the relationship between our proposed PCSE and NESC [12] for binary classification scenarios (Section 5.2). Due to space limitations, the detailed proofs of all the theorems in this section are deferred to the **supplementary material**.

5.1 Estimation Error Bound

Our method aims to estimate the mean and covariance values of clean data. Such two estimations can be simply decomposed into the first- and second-order statistics[1]. Therefore, here we directly provide the estimation error bounds of our method on clean first- and second-order statistics by the following theorem.

Theorem 1. ***(Estimation error bounds under clean*** $\mathbf{T}$***)*** *Let $\psi_1(\mathbf{x}) = \mathbf{x}$ and $\psi_2(\mathbf{x}) = vec(\mathbf{x}\mathbf{x}^\top)^2$ correspond to the first- and second-order statistics, respectively. For $s \in \{1,2\}$, assume that we have a bounded space $\mathcal{X}^{(s)} = \{\psi_s(\mathbf{x}) \,|\, \|\psi_s(\mathbf{x})\|_2 \le \overline{X}^{(s)}\} \subseteq \mathbb{R}^{d^s}$. We denote that $\mathbf{U}^{(s)} = [\boldsymbol{u}_1^{(s)}, \cdots, \boldsymbol{u}_C^{(s)}]$, where $\boldsymbol{u}_j^{(s)} = \mathbb{E}_{X|Y=\mathbf{e}_j}[\psi_s(X)]$ for $j \in [\![C]\!]$, and $\widehat{\mathbf{U}}^{(s)} = [\widehat{\boldsymbol{u}}_1^{(s)}, \cdots, \widehat{\boldsymbol{u}}_C^{(s)}]$ is the proposed estimator of $\mathbf{U}^{(s)}$. For $s \in \{1,2\}$ and any $\delta > 0$, when the number of training examples $\widetilde{n} > 2C^2\|\mathbf{T}^{-1}\|_1^2 \log\frac{8C}{\delta}$, with probability at least $1-\delta$, we have*

$$\begin{aligned}\left\|\mathbf{U}^{(s)} - \widehat{\mathbf{U}}^{(s)}\right\|_2 \le{}& \gamma^{(s)}\sqrt{\frac{2d^sC}{\min_k \widetilde{n}_k}\log\frac{8d^sC}{\delta}}\\ &+ (\zeta^{(s)}\|\mathbf{T}^{-1}\|_1 + \beta^{(s)})\cdot\sqrt{\frac{1}{2\widetilde{n}}\log\frac{8C}{\delta}},\end{aligned} \tag{24}$$

where $\zeta^{(s)} = \frac{\overline{X}^{(s)}\sqrt{C}\xi_M cond_2(\widehat{\mathbf{M}})}{\min_k \widehat{\widetilde{\pi}}_k \min_k \pi_k} + \frac{2\overline{X}^{(s)}C\sqrt{C}\max_k \widehat{\widetilde{\pi}}_k}{\min_k \pi_k}$, $\beta^{(s)} = \frac{\overline{X}^{(s)}\sqrt{C}\xi_M cond_2(\mathbf{M})}{\min_k \pi_k}$, $\gamma^{(s)} = \frac{\overline{X}^{(s)}\cdot\xi_M cond_2(\mathbf{M})\max_k \widetilde{\pi}_k}{\min_k \pi_k}$, ξ_M is a positive constant, and $cond_2(\mathbf{M})$, $cond_2(\widehat{\mathbf{M}})$ are the condition numbers of $\mathbf{M}$ and $\widehat{\mathbf{M}}$, respectively. Additionally, the number of examples for the k-th class is $\widetilde{n}_k$.

Note that the above theorem is derived based on the clean transition matrix $\mathbf{T}$. However, this matrix should be estimated and may not be accurate practically, so we further derive an estimation error bound based on the estimated transition matrix $\widehat{\mathbf{T}}$ as below:

Theorem 2. ***(Estimation error bounds under noisy*** $\mathbf{T}$***)*** *Let $\widehat{\mathbf{T}}$ be the estimated transition matrix. Based on the assumptions in Theorem 1, if we further assume that $\mathbf{I}+(\mathbf{T}-\widehat{\mathbf{T}})\widehat{\mathbf{T}}^{-1}$ is invertible, and the norm of its inverse matrix is upper-bounded, then for $s \in \{1,2\}$ and any $\delta > 0$, when $\widetilde{n} > 2C^2\|\mathbf{T}^{-1}\|_1^2\log\frac{8C}{\delta}$, with probability at least $1-\delta$, we have*

$$\begin{aligned}\left\|\mathbf{U}^{(s)} - \widehat{\mathbf{U}}^{(s)}\right\|_2 \le{}& \gamma^{(s)}\sqrt{\frac{2d^sC}{\min_k \widetilde{n}_k}\log\frac{8d^sC}{\delta}}\\ &+ (\zeta^{(s)}\|\widehat{\mathbf{T}}^{-1}\|_1 + \beta^{(s)})\sqrt{\frac{1}{2\widetilde{n}}\log\frac{8C}{\delta}} + \mathcal{O}\left(\frac{\|\mathbf{T}-\widehat{\mathbf{T}}\|_1}{\sqrt{\widetilde{n}}}\right),\end{aligned} \tag{25}$$

where each symbol has the same definition as in Theorem 1.

Remark 3. From the above theorems, we can observe that when $\min_k \widetilde{n}_k \to \infty$, we have $\widehat{\mathbf{U}}^{(s)} \to \mathbf{U}^{(s)}$ in probability. Based on Theorem 1 and simple calculation, we find that the estimation error bounds of mean and covariance have the order of $\mathcal{O}(\sqrt{dC\log(dC)}/\sqrt{\min_k \widetilde{n}_k} + C\sqrt{C\log C}/\sqrt{\widetilde{n}})$ and $\mathcal{O}((d^2\sqrt{C\log(d^2C)} + d\sqrt{dC\log(dC)})/\sqrt{\min_k \widetilde{n}_k} + dC\sqrt{C\log C}/\sqrt{\widetilde{n}})$, respectively, which means the estimation

1. For $\forall i \in [\![C]\!]$, we have $\boldsymbol{\mu}_i = \mathbf{U}_i^{(1)}$, and $\boldsymbol{\Sigma}_i = mat(\mathbf{U}_i^{(2)}) - \mathbf{U}_i^{(1)}\mathbf{U}_i^{(1)\top}$, where $\mathbf{U}^{(1)}$ and $\mathbf{U}^{(2)}$ defined in Theorem 1 correspond to the matrices consisting of the first- and second-order statistics, respectively.

2. Suppose that the row-block matrix $\mathbf{B}_i \in \mathbb{R}^{d^2\times d}$ consists of d blocks with size $d \times d$, with an identity matrix only in the i-th block and the others are all zeros. Then for any matrix $\mathbf{Q} \in \mathbb{R}^{d\times d}$, the $vec(\cdot)$ operator is defined as $vec(\mathbf{Q}) = \sum_{i=1}^{d}\mathbf{B}_i\mathbf{Q}\mathbf{e}_i$, and for any vector $\mathbf{q} \in \mathbb{R}^{d^2}$, the $mat(\cdot)$ operator is defined as $mat(\mathbf{q}) = \sum_{i=1}^{d}\mathbf{B}_i^\top\mathbf{q}\mathbf{e}_i^\top$.

error of our PCSE decreases quickly with the increase of the numbers of noisy examples (i.e., $\widetilde{n}$ and $\widetilde{n}_k$). Additionally, from Theorem 2, we can find that despite the bias existing in the estimated transition matrix, the estimated value $\widehat{\mathbf{U}}^{(s)}$ converges to the ground-truth statistic $\mathbf{U}^{(s)}$. Note that the condition $\widetilde{n} > 2C^2\|\mathbf{T}^{-1}\|_1^2 \log(8C/\delta)$ is required in both theorems. Actually, this condition can be easily achieved in our implementations. For example, if we set $C = 10$, noise rate $\epsilon = 20\%$, and confidence $1-\delta = 95\%$, then the condition becomes $\widetilde{n} > 3383 = 2C^2\|\mathbf{T}^{-1}\|_1^2 \log(8C/\delta)$.

Remark 4. In Theorem 1, the factor $\|\mathbf{T}^{-1}\|_1$ can be interpreted as a constant that captures the overall amount of label noise. Lower levels of noise are associated with smaller estimation errors. To demonstrate this point, we let $\mathbf{I}$ and $\mathbf{N}$ be the identity matrix and the matrix with all elements equaling to $\frac{1}{C}$, respectively, where C is the number of classes. Let $\tau \in [0,1]$, we define $\mathbf{T} = (1-\tau)\mathbf{I} + \tau\mathbf{N}$. Therefore, $\tau = 0$ represents the noise-free case, and $\tau = 1$ is the noise-only case. It is easy to verify that $\mathbf{T}^{-1} = (1-\tau)^{-1}(\mathbf{I} - \tau\mathbf{N})$ and $\|\mathbf{T}^{-1}\|_1 = (1-\tau)^{-1}(1 + (1-\frac{2}{C})\tau)$. Since $\tau = \frac{C}{C-1}\epsilon$ for symmetric label noise, τ can also be perceived as the noise level. Therefore, as τ decreases, $\|\mathbf{T}^{-1}\|_1$ also decreases, which leads to the lower estimation error bound.

Remark 5. Comparing Theorem 2 with Theorem 1, we can find that with biased or noisy $\mathbf{T}$, the error upper bound becomes losser than the one with clean $\mathbf{T}$. As we have discussed in Theorem 2, the bound factor caused by the biased $\mathbf{T}$ is $\mathcal{O}\left(\|\mathbf{T}-\widehat{\mathbf{T}}\|_1/\sqrt{\widetilde{n}}\right)$, which will gradually vanish with the increase of the training sample size $\widetilde{n}$.

Based on the above estimation error bounds, the factors that affect the estimation error can be identified, which are

- **The number of categories**. Decreasing the number of categories C will lead to lower estimation error bounds.
- **The degree of class imbalance**. The estimation error decreases when the training data is balanced. Since the balanced training set has a larger $\min_k \pi_k$ than the imbalanced one, we can obtain smaller $\zeta^{(s)}$, $\beta^{(s)}$, and $\gamma^{(s)}$ for $s \in \{1,2\}$, which leads to lower estimation error bounds.
- **The noise level**. As explained above, the estimation error decreases when the level of label noise $\|\mathbf{T}^{-1}\|_1$ is low, which is consistent with the intuition that cleaner data naturally leads to better convergence results.
- **The dimension of feature**. Decreasing dimension of feature d will lead to smaller estimation error bounds since low-dimensional data is simpler and is more controllable than high-dimensional data.

In summary, the above two theorems clearly indicate that our estimations of per-class statistics have guaranteed convergence so that they can recover the statistics under clean distribution. Such convergence is critical for our generative modeling in Eqs. (11) and (12).

5.2 Relationship between NESC [12] and our PCSE

In addition to identifying the factors that affect estimation errors, we also discover the connection between our PCSE and NESC [12], which only provides unbiased estimators of per-class statistics under binary classification.

Now we consider the binary classification problem and assume that the label space is $\mathcal{Y}_{\text{bin}} = \{-1,+1\}$. The noisy and clean positive class priors are denoted by $\widetilde{\pi} = P(\widetilde{Y} = +1)$ and $\pi = P(Y = +1)$, respectively. We denote the label flip probabilities of the positive and negative classes by $\eta_P = P(\widetilde{Y} = -1|Y = +1)$ and $\eta_N = P(\widetilde{Y} = +1|Y = -1)$, respectively. Let $\boldsymbol{\mu}_N = \mathbb{E}_{X|Y=-1}[X]$ and $\boldsymbol{\mu}_P = \mathbb{E}_{X|Y=+1}[X]$ be the clean sample means of the negative class and the positive class, respectively. The contaminated counterparts are $\widetilde{\boldsymbol{\mu}}_N = \mathbb{E}_{X|\widetilde{Y}=-1}[X]$ and $\widetilde{\boldsymbol{\mu}}_P = \mathbb{E}_{X|\widetilde{Y}=+1}[X]$, respectively. Moreover, we denote $\boldsymbol{\sigma}_N = \mathbb{E}_{X|Y=-1}[vec(XX^\top)]$ and $\boldsymbol{\sigma}_P = \mathbb{E}_{X|Y=+1}[vec(XX^\top)]$ as the vectorized clean second-order statistics, where $vec(\cdot)$ is the vectorization of the input matrix. The corresponding contaminated counterparts are $\widetilde{\boldsymbol{\sigma}}_N = \mathbb{E}_{X|\widetilde{Y}=-1}[vec(XX^\top)]$ and $\widetilde{\boldsymbol{\sigma}}_P = \mathbb{E}_{X|\widetilde{Y}=+1}[vec(XX^\top)]$, respectively. The connection between our PCSE and NESC is established in the following theorem.

Theorem 3. ***(Connection between PCSE and NESC [12])*** *For binary classification, if the label noise is symmetric, i.e., $\eta_P = \eta_N = \eta$, then the estimators of PCSE and NESC for per-class first- and second-order statistics are the same, which are*

$$\begin{cases} [\boldsymbol{\mu}_P, \boldsymbol{\mu}_N] = [\widetilde{\boldsymbol{\mu}}_P, \widetilde{\boldsymbol{\mu}}_N]\mathbf{S} \\ [\boldsymbol{\sigma}_P, \boldsymbol{\sigma}_N] = [\widetilde{\boldsymbol{\sigma}}_P, \widetilde{\boldsymbol{\sigma}}_N]\mathbf{S} \end{cases}, \tag{26}$$

where $\mathbf{S} = \begin{bmatrix} \frac{(1-\widetilde{\pi})\cdot(1-\eta)}{1-\widetilde{\pi}-\eta} & \frac{-\eta(1-\widetilde{\pi})}{\widetilde{\pi}-\eta} \\ \frac{-\widetilde{\pi}\cdot\eta}{1-\widetilde{\pi}-\eta} & \frac{\widetilde{\pi}\cdot(1-\eta)}{\widetilde{\pi}-\eta} \end{bmatrix}$ *is the coefficient matrix.*

This theorem shows that NESC and our PCSE have the same estimators of per-class sample mean and covariance under symmetric label noise. However, for asymmetric label noise, namely $\eta_P \neq \eta_N$, the estimators of our PCSE and NESC are different. In the next section, we will demonstrate that in the presence of asymmetric label noise, the estimation error of PCSE is usually smaller than that of NESC. Besides, another advantage of our PCSE over NESC is that NESC is only applicable to binary classification, while our PCSE can also handle multi-class classification.

6 Experimental Results

In this section, we show experimental results on various datasets to validate the effectiveness of our proposed method. In detail, we first conduct experiments to reveal the superior performance of our PCSE on statistic estimation over other methods, and then verify that our PCSE can boost the classification performance of many DNNs pre-trained on the noisy set. Afterwards, we compare our PCSE with existing state-of-the-art LNL methods on various benchmark and real-world datasets.

6.1 Algorithm Validation

In this section, we conduct experimental investigations to comprehensively evaluate the performance of our PCSE, including the estimation error analysis of the proposed statistic estimators and the classification performance evaluation with various pre-training methods.

In the following analyses, all experiments are conducted on *CIFAR-10* and *CIFAR-100* [24]. The *CIFAR-10* dataset contains 60,000 color images across 10 classes, with 6,000 images per class. There are 5,000 training images and 1,000 test images per class. The *CIFAR-100* dataset consists of 60,000 color images from 100 classes, with 600 images per class. There are 500 training images and 100 test images

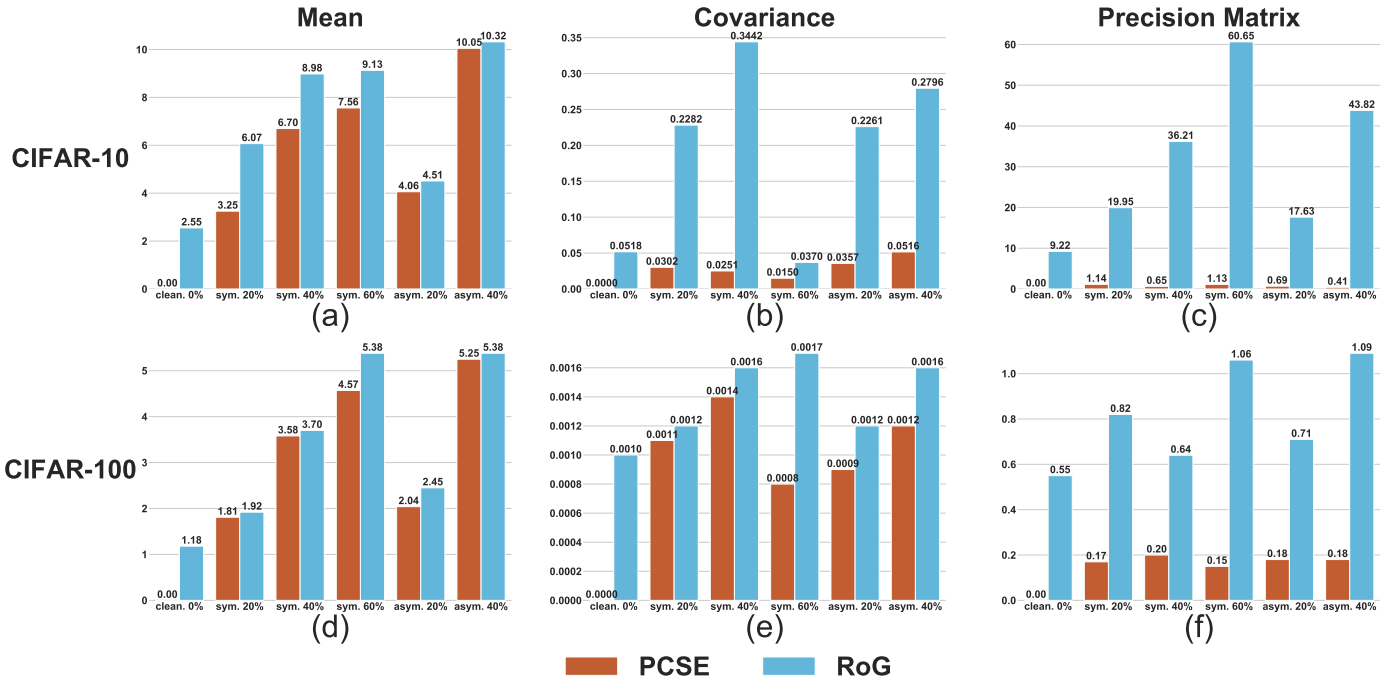


Fig. 3. Estimation errors of RoG and our PCSE on per-class mean, covariance and precision matrix. The average error over 3 independent trials is reported. The first and second rows show the estimation errors of various methods on *CIFAR-10* and *CIFAR-100*, respectively, and the three columns present the estimation errors on sample mean, covariance and precision matrix, respectively. We use "sym. ϵ" to denote the symmetric label noise with noise rate ϵ, "asym." to denote the pairflip label noise and "clean. 0%" to denote the noise-free case. This figure clearly indicates that our PCSE can obtain more precise estimations of per-class statistics than RoG.

TABLE 2
Comparison of RoG and PCSE when different pre-training methods are adopted. The best two records on each dataset are highlighted in red and blue, respectively. The "√"("×") denotes that our PCSE is significantly better (worse) than the others. The average performance improvement of PCSE over RoG and the average performance improvement of both approaches over their pre-training method are also presented.

Dataset	Method	clean. 0%	sym. 20%	sym. 40%	sym. 60%	asym. 20%	asym. 40%	avg. improvement over RoG (%)	avg. improvement over pre-training method (%)
CIFAR-10	CrossEntropy	94.07 ± 0.13	84.05 ± 0.32 √	67.34 ± 0.58 √	41.14 ± 0.86 √	81.41 ± 0.36 √	59.07 ± 0.79 √	-	-
	CrossEntropy+RoG [25]	93.99 ± 0.21	87.72 ± 0.12	81.31 ± 0.30	66.50 ± 1.43	89.64 ± 0.47	70.98 ± 0.54 √	-	+ 10.51
	CrossEntropy+PCSE	94.12 ± 0.32	87.89 ± 0.21	82.04 ± 0.31	68.85 ± 1.06	90.02 ± 0.21	78.80 ± 0.74	+ 1.93	+ 12.44
	Co-teaching [18]	94.04 ± 0.36	91.17 ± 0.06 √	85.88 ± 0.21 √	70.25 ± 0.99 √	91.49 ± 0.12 √	70.36 ± 0.38 √	-	-
	Co-teaching+RoG [25]	94.01 ± 0.16	90.07 ± 0.35 √	85.41 ± 0.44 √	78.47 ± 0.91	91.08 ± 0.41 √	78.92 ± 0.96 √	-	+ 2.44
	Co-teaching+PCSE	94.12 ± 0.17	92.29 ± 0.24	89.86 ± 0.14	80.29 ± 0.93	92.11 ± 0.11	84.78 ± 0.33	+ 2.60	+ 5.04
	JoCoR [44]	94.26 ± 0.32	91.02 ± 0.26 √	89.62 ± 0.59	69.21 ± 1.12 √	89.23 ± 0.26 √	81.83 ± 0.65	-	-
	JoCoR+RoG [25]	93.73 ± 0.43	91.32 ± 0.34	88.34 ± 0.45 √	75.77 ± 0.89 √	89.65 ± 0.23 √	75.12 ± 0.87 √	-	− 0.21
	JoCoR+PCSE	94.36 ± 0.39	91.92 ± 0.35	88.97 ± 0.42	79.21 ± 1.54	91.52 ± 0.06	82.43 ± 0.14	+ 2.41	+ 2.21
	ASL [58]	90.46 ± 0.21	89.72 ± 0.20	85.02 ± 0.25 √	76.70 ± 1.15	88.76 ± 1.24	75.43 ± 1.02	-	-
	ASL+RoG [25]	90.36 ± 0.19	89.98 ± 0.04	86.43 ± 0.32	74.29 ± 2.05	88.65 ± 1.02	66.91 ± 1.21 √	-	− 1.58
	ASL+PCSE	90.49 ± 0.14	90.17 ± 0.13	86.92 ± 0.20	76.00 ± 1.12	89.65 ± 0.97	75.80 ± 0.87	+ 2.07	+ 0.49
	ROBOT [52]	94.16 ± 0.16	91.43 ± 0.14 √	89.03 ± 0.21	81.44 ± 0.75 √	91.23 ± 0.22	88.81 ± 0.35	-	-
	ROBOT+RoG [52]	94.00 ± 0.02	91.46 ± 0.26	88.96 ± 1.78	80.71 ± 0.61	90.94 ± 0.60	85.44 ± 0.69	-	− 0.77
	ROBOT+PCSE	94.12 ± 0.12	91.86 ± 0.14	89.40 ± 1.34	81.70 ± 0.40	91.56 ± 0.03	89.03 ± 3.57	+ 1.03	+ 0.26
CIFAR-100	CrossEntropy	72.62 ± 0.23	58.04 ± 0.11 √	41.40 ± 0.95 √	21.75 ± 1.00 √	60.02 ± 0.33 √	42.22 ± 0.68 √	-	-
	CrossEntropy+RoG [25]	70.69 ± 0.22 √	60.98 ± 0.20 √	52.96 ± 0.86	37.59 ± 1.43	65.21 ± 0.54	56.28 ± 0.75	-	+ 7.94
	CrossEntropy+PCSE	72.71 ± 0.34	61.55 ± 0.19	53.91 ± 1.02	39.13 ± 1.05	65.47 ± 0.52	56.65 ± 0.43	+ 0.95	+ 8.89
	Co-teaching [18]	72.82 ± 0.18 ×	64.18 ± 0.33 √	58.32 ± 1.05 √	44.01 ± 0.51 √	64.87 ± 0.62	51.60 ± 0.56	-	-
	Co-teaching+RoG [25]	69.57 ± 0.41 √	63.12 ± 0.61 √	58.24 ± 1.24 √	44.13 ± 0.80 √	63.68 ± 0.58	52.50 ± 0.41	-	− 0.76
	Co-teaching+PCSE	72.33 ± 0.44	66.90 ± 0.55	59.49 ± 1.01	46.95 ± 0.92	65.33 ± 0.49	52.94 ± 0.65	+ 2.12	+ 1.36
	JoCoR [44]	71.00 ± 0.08 ×	63.19 ± 0.21 √	56.51 ± 0.40 √	44.47 ± 3.68	64.69 ± 0.85	50.88 ± 0.77	-	-
	JoCoR+RoG [25]	68.08 ± 0.13 √	64.07 ± 0.31 √	57.79 ± 0.52	47.28 ± 3.15	64.07 ± 1.02	54.08 ± 0.92	-	+ 0.77
	JoCoR+PCSE	69.94 ± 0.21	65.64 ± 0.28	58.77 ± 0.49	49.10 ± 3.21	65.54 ± 0.54	54.64 ± 1.32	+ 1.38	+ 2.15
	ASL [58]	72.19 ± 0.05	68.65 ± 0.43	62.16 ± 0.29 √	51.31 ± 0.54 √	63.43 ± 0.46	42.50 ± 0.35	-	-
	ASL+RoG [25]	70.68 ± 0.35 √	67.83 ± 0.40	61.86 ± 0.31	52.96 ± 0.40	61.01 ± 0.58 √	43.72 ± 0.95	-	− 0.36
	ASL+PCSE	72.28 ± 0.23	68.72 ± 0.35	63.00 ± 0.15	53.24 ± 0.25	63.61 ± 0.42	43.03 ± 0.43	+ 0.97	+ 0.61
	ROBOT [52]	76.25 ± 0.08	72.29 ± 0.14	67.22 ± 0.46	59.51 ± 0.05 √	70.94 ± 0.50	57.06 ± 0.43 √	-	-
	ROBOT+RoG [52]	75.42 ± 0.06 √	71.57 ± 0.18 √	67.42 ± 0.33	59.20 ± 0.72	71.69 ± 0.40	60.31 ± 0.44 √	-	− 0.39
	ROBOT+PCSE	76.32 ± 0.07	72.46 ± 0.04	67.74 ± 0.14	59.95 ± 0.14	71.91 ± 0.33	64.59 ± 1.62	+ 1.23	+ 1.62

for each class. Both fine-grained and coarse-grained labels are provided for each example in *CIFAR-100*. Here we adopt the fine-grained labels throughout all experiments. Since the original labels in *CIFAR-10* and *CIFAR-100* are noise-free, we consider injecting synthetic label noise to simulate different noise settings. Here we follow [9], [15] and investigate two types of label noise, namely: 1) symmetric label noise with noise rate $\epsilon \in \{20\%, 40\%, 60\%\}$ (denoted as "sym. ϵ" hereinafter), which means that for each example, we uniformly select a label different from its ground-truth label with probability $\epsilon/(C-1)$; and 2) pairflip label noise with noise rate $\epsilon \in \{20\%, 40\%\}$ (denoted as "asym. ϵ" hereinafter),

which means that the label of each class is flipped into the next class circularly with probability ϵ. We also consider the noise-free case, which is denoted by "clean. 0%".

6.1.1 Estimation Error Analysis

Since the statistic estimation is crucial in our proposed method, we start by investigating the estimation errors of per-class statistics (i.e., mean and covariance matrix) generated by RoG [25] and our PCSE on *CIFAR-10* and *CIFAR-100* datasets, because RoG is also based on statistic estimation as mentioned in Section 3.2. Moreover, as we use the inverse of covariance matrix (a.k.a. precision matrix) for estimating the clean class posterior (see Eq. (11)), we also investigate the estimation error of our method on precision matrix. To investigate the estimation errors of these statistics, we utilize the following metrics for the mean $\boldsymbol{\mu}$, covariance $\boldsymbol{\Sigma}$, and precision matrix $\boldsymbol{\Sigma}^{-1}$, respectively, which are:

$$\begin{aligned}
\text{Error}_{\boldsymbol{\mu}} &= \tfrac{1}{C}\textstyle\sum_{l\in\mathcal{L}}\sum_{c}\left\|\boldsymbol{\mu}_c^{(l)}-\widehat{\boldsymbol{\mu}}_c^{(l)}\right\|_2 \\
\text{Error}_{\boldsymbol{\Sigma}} &= \tfrac{1}{d^2}\textstyle\sum_{l\in\mathcal{L}}\sum_{i,j}\left|\boldsymbol{\Sigma}_{ij}^{(l)}-\widehat{\boldsymbol{\Sigma}}_{ij}^{(l)}\right| \\
\text{Error}_{\boldsymbol{\Sigma}^{-1}} &= \tfrac{1}{d^2}\textstyle\sum_{l\in\mathcal{L}}\sum_{i,j}\left|(\boldsymbol{\Sigma}^{(l)})_{ij}^{-1}-(\widehat{\boldsymbol{\Sigma}}^{(l)})_{ij}^{-1}\right|
\end{aligned}, \quad (27)$$

where $\widehat{\boldsymbol{\mu}}_c^{(l)}$ and $\widehat{\boldsymbol{\Sigma}}^{(l)}$ are the estimated noise-free statistics while $\boldsymbol{\mu}_c^{(l)}$ and $\boldsymbol{\Sigma}^{(l)}$ are the corresponding ground-truth values calculated with clean labels.

The estimation errors of per-class statistics for RoG and our proposed PCSE on *CIFAR-10* and *CIFAR-100* are shown in Fig. 3. As depicted in this figure, our PCSE consistently obtains lower estimation errors on *CIFAR-10* than RoG. For example, PCSE achieves a striking reduction of 50% to 90% in the estimation errors of covariance matrices when compared with RoG (see Fig. 3(b)). A remarkable reduction in the estimation errors of the precision matrices can also be observed in Fig. 3(c), where the estimation errors of our PCSE are 50 to 100 times smaller than those of RoG. On *CIFAR-100*, our PCSE achieves a similar result to that on *CIFAR-10*, with slightly improved estimations of sample means over RoG. However, PCSE significantly outperforms RoG in estimating the precision matrix. To summarize, the results on the estimation errors clearly indicate that our PCSE can obtain more accurate estimations of mean, covariance, and precision matrix than RoG. The comparison of estimation errors on binary classification datasets is deferred to the **supplementary material**.

6.1.2 Experiments with Various Pre-training Methods

In this subsection, we show that PCSE can boost the classification performance of a pre-trained DNN regardless of whether it is a label-noise-robust method. To this end, we consider different pre-training methods, including CrossEntropy (which directly minimizes the conventional cross-entropy loss), Co-teaching [18], JoCoR [44], ASL [58], and ROBOT [52]. Here, CrossEntropy is a non-robust method while others are specifically designed robust methods for handling label noise. In detail, Co-teaching is a powerful sample selection based method, ASL is the state-of-the-art LNL method based on robust loss function design, and ROBOT is the state-of-the-art method for the estimation of transition matrices. Besides, JoCoR is a hybrid method that combines sample selection with consistency regularization. For all these methods, the optimal hyperparameters recommended in their original papers are adopted in all of our experiments. For example, in Co-teaching, the selection ratio $R(T)$ is set to $1-\epsilon\cdot\min\{T/10,1\}$ for a given noise rate ϵ at the T-th epoch. Here we use "CrossEntropy+RoG" and "CrossEntropy+PCSE" to denote that the pre-training method is CrossEntropy and the post-processing methods are RoG and PCSE, respectively. The same naming rule holds for other pre-training methods.

The experimental results on *CIFAR-10* and *CIFAR-100* with different pre-training methods (i.e., CrossEntropy, Co-teaching [18], JoCoR [44]) are shown in Table 2, where the average performance improvement of PCSE and RoG over their corresponding pre-training method is also presented. In this table, we find that PCSE greatly boosts the classification performance of pre-trained DNNs when the pre-training method is CrossEntropy (i.e., non-robust pre-training method). Specifically, on *CIFAR-10* dataset, PCSE improves the classification accuracy of a DNN pre-trained with vanilla Cross-Entropy loss by 12.57%. The corresponding improvement on *CIFAR-100* dataset is 8.89%. Additionally, we identify that RoG underperforms the pre-training methods in some cases while our PCSE consistently outperforms the adopted pre-training methods. Besides, the proposed method surpasses RoG in all scenarios (see the penultimate column of Table 2). We conjecture that it is attributed to the significant improvement of PCSE over RoG in estimating some key statistics (see Fig. 3). In Table 2, we can observe that in noise-free cases (i.e., clean. 0%), the post-processing sometimes leads to degraded performance. To explain this phenomenon, we use $\phi_{\mathbf{v}}(\cdot)=\phi_L\circ\phi_{L-1}\cdots\circ\phi_1(\cdot)\in\mathbb{R}^d$ and $f_{\mathbf{W}}(\phi_{\mathbf{v}}(\cdot))=\left[\frac{\exp(\mathbf{w}_1^\top\phi_{\mathbf{v}}(\cdot))}{\sum_j\exp(\mathbf{w}_j^\top\phi_{\mathbf{v}}(\cdot))},\frac{\exp(\mathbf{w}_2^\top\phi_{\mathbf{v}}(\cdot))}{\sum_j\exp(\mathbf{w}_j^\top\phi_{\mathbf{v}}(\cdot))},\cdots,\frac{\exp(\mathbf{w}_C^\top\phi_{\mathbf{v}}(\cdot))}{\sum_j\exp(\mathbf{w}_j^\top\phi_{\mathbf{v}}(\cdot))}\right]\in\mathbb{R}^C$ to denote the feature extractor and the classifier, parameterized by $\mathbf{v}$ and $\mathbf{W}=[\mathbf{w}_1,\mathbf{w}_2,\cdots,\mathbf{w}_C]$, respectively. In the noise-free case, since $\widetilde{\mathbf{y}}=\mathbf{y}$, the pre-training actually solves the following problem:

$$(\mathbf{W}^*,\mathbf{v}^*)\in\arg\min_{\mathbf{W},\mathbf{v}} g(\mathbf{W},\mathbf{v})=\mathbb{E}_{(\mathbf{x},\mathbf{y})\sim S}\left[\mathcal{L}(f_{\mathbf{W}}(\phi_{\mathbf{v}}(\mathbf{x})),\mathbf{y})\right], \quad (28)$$

where $\mathbf{W}^*$ and $\mathbf{v}^*$ are the optimal parameters, and $\mathcal{L}$ is a loss function. In post-processing, we fix $\mathbf{v}^*$ and replace $\mathbf{W}^*$ with $\mathbf{W}'\in\mathcal{C}_{\mathbf{W}}=\{\mathbf{W}\,|\,\mathbf{w}_i=\mathbf{A}\mathbf{u}_i,\mathbf{A}\succ 0,\mathbf{u}_i\in\mathbb{R}^d\}$, which imposes more constraints on $\mathbf{W}$. Suppose that $\mathbf{W}^{**}\in\arg\min_{\mathbf{W}\in\mathcal{C}_{\mathbf{W}}} g(\mathbf{W},\mathbf{v}^*)$, and we have $g(\mathbf{W}^*,\mathbf{v}^*)<g(\mathbf{W}^{**},\mathbf{v}^*)$. Since $\mathbf{W}'$ is an element in $\mathcal{C}_{\mathbf{W}}$, we have $g(\mathbf{W}^{**},\mathbf{v}^*)\le g(\mathbf{W}',\mathbf{v}^*)$, and thereby $g(\mathbf{W}^*,\mathbf{v}^*)<g(\mathbf{W}',\mathbf{v}^*)$. Since $g(\cdot,\cdot)$ is a surrogate loss for the error rate, this reveals that the post-processing can lead to the degraded performance in the noise-free case. Notably, such phenomenon also exists in [25] as well.

6.2 Comparison with Other Existing Methods

In this section, we conduct experimental investigations to demonstrate the superior performance of our proposed PCSE to state-of-the-art methods in dealing with noisy labels. Specifically, we conduct intensive experiments on a variety of datasets, including twelve UCI benchmark datasets [2] (Section 6.2.1), two synthetic noisy datasets (Section 6.2.2), and five real-world noisy datasets (Section 6.2.3).

Our experiments provide a comprehensive validation of the model generalizability of PCSE.

6.2.1 Experiments on Binary UCI Benchmark Datasets

In line with previous studies [15], [33], we start by conducting experiments on twelve benchmark datasets regarding binary classification from UCI machine learning repository [2], including *Breast cancer*, *Heart*, *Diabetes*, *German*, *Image*[3]. Experiments on additional seven UCI datasets (namely, *GammaTele*, *Banana*, *Ringnorm*, *Splice*, *Thyroid*, *Twonorm*, and *Waveform*) can be found in the **supplementary material**. A brief introduction of the datasets is presented in Table 3, which contains some essential configurations such as the number of examples n, the feature dimensionality d, the number of positive examples n_+, and the number of negative examples n_-. The features for each dataset have been normalized and standardized.

Subsequently, we compare our PCSE with other existing LNL methods in terms of classification accuracy on the popular UCI benchmark datasets. In our experiments, we use five-fold cross-validation for model selection. Note that training and cross-validation are conducted on the noisy training set in our settings. To stably simulate a given noise rate, we repeat each experiment three times with different random seeds. The average test accuracy and the standard deviation over the trials are recorded. Besides, the paired t-test with a significance level of 0.1 is employed to examine whether our algorithm is significantly better or worse than the compared methods. Following the previous work [15], [33], we choose three pairs of label flip rates, namely $(\eta_P, \eta_N) = (0.2, 0.2)$, $(\eta_P, \eta_N) = (0.4, 0.4)$, and $(\eta_P, \eta_N) = (0.3, 0.1)$, where the first two cases correspond to symmetric label noise while the last one represents asymmetric label noise. Additionally, we test whether our method can handle the noise-free case by considering $(\eta_P, \eta_N) = (0.0, 0.0)$.

The compared methods are CrossEntropy introduced in Section 6.1.2, GCE [57], $\mathcal{L}_{\text{DMI}}$ [50], Co-teaching [18], CEGE [16], CWD [15], ULE [33], RoG [25], MC-LDCE [9], ASL [58], ROBOT [52], NESC [12], and our PCSE. Here, Co-teaching is a representative sample selection based method; GCE, $\mathcal{L}_{\text{DMI}}$, ULE, and ASL are typical methods based on robust loss functions; CEGE, ROBOT, CWD, and MC-LDCE are competitive statistic estimation based methods, where the latter two are highly related to our PCSE in that they are both proposed to estimate the centroid of the dataset. Besides, RoG and NESC are both proposed to estimate class-wise mean and covariance, and the former is specifically designed as a post-processing approach. Therefore, by including the above methods, we can comprehensively compare our method with various different types of existing methods. The experimental settings for the compared methods are deferred to the **supplementary material**. It is worth noting that the main differences between RoG, NESC, and PCSE lie in the estimators of the sample mean and covariance, so the comparison of their accuracies can directly reflect the estimation quality.

The classification results of all compared methods on five adopted UCI benchmark datasets are recorded in Table 4, where our PCSE ranks among the top two in most cases.

3. These datasets are available at http://theoval.cmp.uea.ac.uk/matlab which have already been preprocessed.

TABLE 3
Properties of five adopted UCI Benchmark datasets.

Datasets	n	d	n_+	n_-
Breast cancer	263	9	77	186
Heart	270	13	120	150
Diabetes	768	8	268	500
German	1000	20	300	700
Image	2086	18	1188	898

Under symmetric label noise, NESC has the same unbiased estimators as our PCSE, so the classification results are the same for the two methods when $\eta_P = \eta_N$, which provides evidence for the correctness of Theorem 3. However, under asymmetric label noise, our proposed method shows significantly better results than NESC in four out of five datasets, which can be attributed to the reduction in estimation errors of per-class statistics. For the average accuracy over all five datasets under different label flip rates, our PCSE achieves a record of 77.3%, which leads the second and third best methods by a margin of 0.6% and 2.7%, respectively. To summarize, the results in Table 4 clearly verify the robustness and discriminativeness of our PCSE over other baseline methods in dealing with label noise.

Subsequently, the commonly used Friedman test [7] is employed as the statistical test to analyze the relative performance among the compared approaches. Here, the test is applied individually for each of the four noise rates, namely $(\eta_P, \eta_N) = (0.0, 0.0)$, $(0.2, 0.2)$, $(0.3, 0.1)$, and $(0.4, 0.4)$. The Friedman statistic F_F for the four noise rates are 58.19, 70.29, 48.82, and 68.96, respectively, and the corresponding critical value (at 0.1 significance level) is 3.08 (with 13 algorithms and 12 datasets). Therefore, these algorithms are distinguishable in performance. Finally, the post-hoc Nemenyi test is adopted to illustrate the relative performance among approaches, and PCSE is regarded as the control method. The performance of our PCSE is significantly better than another approach if the corresponding average ranks differ by at least the Critical Difference (CD). Fig. 4 shows the CD diagrams, where the average rank of each algorithm is marked along the axis. In this figure, every compared algorithm whose average rank is within one CD to that of PCSE is interconnected to each other with a bold wavy line. It can be observed that PCSE achieves the best average rank among all cases, and the performance of PCSE is significantly better than RoG in all cases. In Fig. 4(c), PCSE surpasses most of the compared algorithms, including NESC. In summary, the CD diagrams suggest that our method achieves more precise estimations of statistics than RoG and NESC under binary classification scenarios.

6.2.2 Experiments on Synthetic Multi-Class Datasets

To further evaluate the efficacy of PCSE in multi-class classification, we conduct experiments on two standard benchmark datasets, namely *CIFAR-10* and *CIFAR-100* [24].

The baseline methods in this section include the previously used CrossEntropy, GCE, Co-teaching, $\mathcal{L}_{\text{DMI}}$, CWD, RoG, MC-LDCE, ASL, and ROBOT, where ROBOT is adopted as the pre-training method for our PCSE. Here CEGE, ULE, and NESC are not compared as they can only handle binary classification. The backbone network em-

TABLE 4
Comparison of various approaches on five UCI benchmark datasets. The best two records on each dataset are highlighted in red and blue, respectively. The "√"("×") denotes that our PCSE is significantly better (worse) than the corresponding compared method revealed by the paired t-test with significance level 0.1.

Dataset	(η_P, η_N)	CrossEntropy	GCE [57]	$\mathcal{L}_{\mathrm{DMI}}$ [50]	Co-teaching [18]	CEGE [16]	CWD [15]	ULE [33]	RoG [25]	MC-LDCE [9]	ASL [58]	ROBOT [52]	NESC [12]	PCSE
Breast cancer	(0.0, 0.0)	73.7 ± 0.7√	72.4 ± 0.8√	74.4 ± 1.2	73.2 ± 0.6√	73.5 ± 0.4√	73.6 ± 0.2	73.7 ± 0.2√	72.7 ± 0.4√	70.0 ± 1.2√	70.9 ± 1.5	70.7 ± 2.3	75.9 ± 1.1	75.9 ± 1.1
	(0.2, 0.2)	67.2 ± 0.4√	70.3 ± 1.1	70.4 ± 0.6√	61.1 ± 1.1√	69.8 ± 1.2	70.3 ± 0.6√	71.2 ± 0.8	66.5 ± 0.8√	67.3 ± 3.0	68.0 ± 3.7√	66.1 ± 5.0	71.8 ± 0.2	71.8 ± 0.2
	(0.3, 0.1)	70.0 ± 1.3√	71.5 ± 1.6	70.1 ± 2.4√	70.7 ± 0.0	70.4 ± 2.7√	68.7 ± 2.0√	71.5 ± 1.4	70.1 ± 2.3	70.3 ± 1.3√	71.1 ± 0.5	69.3 ± 1.8	69.7 ± 1.5√	72.3 ± 1.8
	(0.4, 0.4)	56.0 ± 2.1√	66.7 ± 3.5	59.5 ± 5.1√	54.9 ± 1.0√	60.6 ± 4.8√	61.1 ± 5.7√	62.7 ± 4.2	59.7 ± 1.7√	56.6 ± 4.2√	58.0 ± 11.3	57.4 ± 4.3√	70.9 ± 2.9	70.9 ± 2.9
Heart	(0.0, 0.0)	83.7 ± 0.9√	83.6 ± 0.7√	83.8 ± 1.1	83.2 ± 0.5√	83.1 ± 1.0√	83.5 ± 1.8	83.2 ± 1.0√	84.8 ± 0.5√	80.4 ± 0.8√	82.5 ± 1.8√	81.9 ± 1.8√	87.3 ± 0.7	87.3 ± 0.7
	(0.2, 0.2)	74.3 ± 3.9√	78.0 ± 2.4	76.7 ± 1.8	68.2 ± 2.2√	75.5 ± 0.5	75.9 ± 2.2√	75.4 ± 2.4	74.8 ± 1.2	76.1 ± 1.5	77.4 ± 3.2	78.5 ± 3.7	80.0 ± 4.1	80.0 ± 4.1
	(0.3, 0.1)	73.1 ± 1.2√	78.4 ± 3.6	75.8 ± 1.6√	72.0 ± 3.2√	72.5 ± 2.5√	75.8 ± 1.1√	74.2 ± 1.6√	77.4 ± 0.0√	73.5 ± 2.3√	75.4 ± 5.1	75.1 ± 6.4	75.7 ± 5.8	81.7 ± 1.8
	(0.4, 0.4)	57.0 ± 2.3	62.0 ± 3.6	63.1 ± 7.5	54.9 ± 2.0	58.3 ± 4.1	62.7 ± 5.2√	59.2 ± 3.7	61.1 ± 2.1√	56.6 ± 12.3	60.3 ± 5.7√	62.7 ± 7.0	65.9 ± 5.6	65.9 ± 5.6
Diabetis	(0.0, 0.0)	76.3 ± 0.4√	76.4 ± 0.2√	77.0 ± 0.3√	75.9 ± 0.6√	76.6 ± 0.4	77.1 ± 0.2	76.7 ± 0.0√	76.4 ± 1.4	75.6 ± 0.6√	75.4 ± 0.2√	76.6 ± 0.7√	78.3 ± 0.6	78.3 ± 0.6
	(0.2, 0.2)	70.1 ± 0.2√	74.7 ± 0.7	74.1 ± 0.8	64.6 ± 1.8√	74.5 ± 1.5√	74.6 ± 0.5	73.6 ± 3.2	70.5 ± 2.9√	73.1 ± 2.0	70.5 ± 3.9√	73.8 ± 1.7√	75.8 ± 1.4	75.8 ± 1.4
	(0.3, 0.1)	69.7 ± 1.8√	74.0 ± 0.9√	74.1 ± 1.3√	69.8 ± 0.9√	74.3 ± 1.0√	74.5 ± 0.9√	73.9 ± 0.4√	69.9 ± 1.1√	71.5 ± 1.9	68.0 ± 1.7√	72.4 ± 3.3	73.6 ± 0.8√	75.3 ± 0.8
	(0.4, 0.4)	60.7 ± 2.3√	66.0 ± 0.2√	64.7 ± 1.4	54.7 ± 1.2√	65.6 ± 0.5√	65.7 ± 1.8	63.7 ± 1.5√	63.6 ± 2.9	62.2 ± 4.2	65.6 ± 0.8√	66.5 ± 0.4√	68.5 ± 0.6	68.5 ± 0.6
German	(0.0, 0.0)	75.3 ± 0.5√	75.9 ± 0.6√	75.3 ± 0.8	76.0 ± 0.2	76.0 ± 0.4√	75.1 ± 0.4√	75.5 ± 0.1√	73.7 ± 1.2√	75.7 ± 0.9	72.4 ± 0.3√	74.1 ± 0.5√	77.7 ± 0.9	77.7 ± 0.9
	(0.2, 0.2)	66.6 ± 3.1	70.8 ± 0.9√	71.2 ± 0.7√	63.4 ± 0.8√	72.3 ± 0.8	71.0 ± 0.8√	71.8 ± 0.8	69.7 ± 0.4√	71.4 ± 1.4	70.7 ± 0.5√	69.5 ± 1.4√	72.9 ± 0.6	72.9 ± 0.6
	(0.3, 0.1)	68.6 ± 2.9√	73.1 ± 0.5	71.6 ± 1.4√	72.2 ± 0.3	71.0 ± 2.0	71.7 ± 1.5√	72.1 ± 1.4√	70.2 ± 2.0	72.5 ± 0.5	70.3 ± 0.4√	68.7 ± 0.2√	71.9 ± 0.4√	73.5 ± 0.9
	(0.4, 0.4)	59.0 ± 3.0√	62.1 ± 1.0√	66.4 ± 0.8√	53.1 ± 1.8√	65.2 ± 2.1√	67.4 ± 3.4	65.3 ± 1.1√	64.2 ± 3.0	60.4 ± 2.5√	67.3 ± 1.4	63.1 ± 0.7√	69.3 ± 0.5	69.3 ± 0.5
Image	(0.0, 0.0)	94.2 ± 0.8	94.4 ± 0.7	94.3 ± 1.4	90.9 ± 0.8√	95.3 ± 0.2	95.3 ± 0.1	95.8 ± 0.4×	90.9 ± 0.8√	90.5 ± 0.6√	89.5 ± 0.8√	94.4 ± 0.4	94.2 ± 0.8	94.2 ± 0.8
	(0.2, 0.2)	87.0 ± 2.2	82.3 ± 13.4	87.0 ± 1.5	76.2 ± 0.5√	90.8 ± 0.1	86.1 ± 4.5	88.7 ± 2.9	84.0 ± 4.2	87.4 ± 1.4√	82.7 ± 2.0√	88.9 ± 1.3	90.5 ± 0.4	90.5 ± 0.4
	(0.3, 0.1)	82.9 ± 4.2	87.2 ± 3.6	87.0 ± 0.6√	74.5 ± 0.7√	87.9 ± 0.6√	87.4 ± 0.7√	86.1 ± 1.9	82.3 ± 3.1√	83.4 ± 1.6√	74.1 ± 1.7√	85.5 ± 3.1	88.3 ± 0.5√	89.8 ± 0.1
	(0.4, 0.4)	65.9 ± 3.2√	66.2 ± 16.4	72.8 ± 7.5	56.5 ± 0.5√	71.8 ± 2.3	74.0 ± 0.6	75.4 ± 0.6	66.0 ± 2.0	73.8 ± 3.2	66.6 ± 2.2	70.4 ± 6.4√	74.9 ± 5.2	74.9 ± 5.2
Average		**71.6**	**74.3**	**74.5**	**68.3**	**74.2**	**74.6**	**74.5**	**72.4**	**72.4**	**71.8**	**73.3**	**76.7**	**77.3**

TABLE 5
Comparison of various approaches on *CIFAR-10* and *CIFAR-100* datasets. The best two records on each dataset are highlighted in red and blue, respectively. The "√"("×") denotes that our PCSE is significantly better (worse) than the corresponding compared method revealed by the paired t-test with significance level 0.05.

Dataset	Noise setting	CrossEntropy	Co-teaching [18]	$\mathcal{L}_{\mathrm{DMI}}$ [50]	GCE [57]	CWD [15]	RoG [25]	MC-LDCE [9]	ASL [58]	ROBOT [52]	PCSE
CIFAR-10	clean. 0%	94.07 ± 0.13	94.04 ± 0.36	94.42 ± 0.07	90.28 ± 0.06√	94.08 ± 0.13	94.01 ± 0.16	93.99 ± 0.08	90.46 ± 0.21√	94.16 ± 0.16	94.12 ± 0.12
	sym. 20%	84.05 ± 0.32√	91.17 ± 0.06√	87.81 ± 0.20√	88.43 ± 0.04√	88.77 ± 0.56√	90.07 ± 0.35√	88.16 ± 0.38√	89.72 ± 0.20√	91.43 ± 0.14√	91.86 ± 0.14
	sym. 40%	67.34 ± 0.58√	85.88 ± 0.21√	83.59 ± 0.47√	84.94 ± 0.21√	84.04 ± 0.10√	85.41 ± 0.44√	84.35 ± 0.41√	85.14 ± 0.25√	89.03 ± 0.21√	89.40 ± 1.34
	sym. 60%	41.14 ± 0.86√	70.25 ± 0.99√	71.76 ± 0.54√	74.52 ± 0.31√	75.75 ± 0.82√	78.47 ± 0.91√	75.85 ± 0.50√	76.70 ± 1.15√	81.44 ± 0.75	81.70 ± 0.40
	asym. 20%	81.41 ± 0.36√	91.49 ± 0.12√	89.96 ± 0.28√	88.10 ± 0.33√	88.89 ± 0.16√	91.08 ± 0.41√	88.77 ± 0.44√	89.38 ± 1.24√	91.23 ± 0.22√	91.56 ± 0.03
	asym. 40%	59.07 ± 0.79√	70.36 ± 0.38√	83.36 ± 2.25	73.12 ± 1.12√	83.66 ± 0.54	78.92 ± 0.96√	81.35 ± 0.26√	75.43 ± 1.02√	88.81 ± 0.35	89.03 ± 3.57
CIFAR-100	clean. 0%	72.62 ± 0.23√	72.82 ± 0.18√	72.12 ± 0.06√	54.76 ± 0.92√	70.00 ± 0.27√	69.57 ± 0.41√	69.14 ± 0.04√	72.19 ± 0.05√	76.25 ± 0.08	76.32 ± 0.07
	sym. 20%	58.04 ± 0.11√	64.18 ± 0.33√	58.94 ± 0.66√	52.41 ± 0.68√	59.64 ± 2.10√	63.12 ± 0.61√	54.90 ± 0.28√	68.65 ± 0.43√	72.29 ± 0.14	72.46 ± 0.04
	sym. 40%	41.40 ± 0.95√	58.32 ± 1.05√	47.86 ± 0.82√	44.81 ± 1.04√	48.05 ± 1.14√	58.24 ± 1.24√	50.63 ± 0.20√	62.16 ± 0.29√	67.22 ± 0.46	67.74 ± 0.14
	sym. 60%	21.75 ± 1.00√	44.01 ± 0.51√	36.74 ± 0.21√	35.34 ± 1.54√	35.04 ± 1.48√	44.13 ± 0.80√	36.35 ± 1.05√	51.31 ± 0.54√	59.51 ± 0.05√	59.95 ± 0.14
	asym. 20%	60.02 ± 0.33√	64.87 ± 0.62√	59.77 ± 0.29√	50.40 ± 0.09√	58.00 ± 1.06√	63.68 ± 0.58√	60.22 ± 0.56√	63.43 ± 0.46√	70.94 ± 0.50	71.91 ± 0.33
	asym. 40%	42.22 ± 0.68√	51.60 ± 0.56√	42.72 ± 0.12√	38.91 ± 1.66√	39.39 ± 2.54√	52.50 ± 0.41√	40.02 ± 0.36√	42.50 ± 0.35√	57.06 ± 0.43√	64.59 ± 1.62

ployed by all compared methods is ResNet-18 [20]. For RoG and our PCSE, we adopt Co-teaching as the pre-training method, and the features used for model inference are from the penultimate layer of the network in Co-teaching.

The experimental results on *CIFAR-10* and *CIFAR-100* are shown in Table 5. For each trial of every approach, the average test accuracy over the last five epochs is recorded, and the mean value of the averaged test accuracy over three independent trials as well as the standard deviation are reported. We can clearly observe that our proposed PCSE method achieves better or comparable performance than the other baseline methods on both datasets in all noisy scenarios. For example, our PCSE outperforms Co-teaching by 3.98% under sym. 40% label noise on *CIFAR-10*. Similarly, on *CIFAR-100*, our PCSE surpasses RoG by 2.82% under sym. 60% label noise. Notably, our PCSE outperforms RoG across all types and levels of label noise.

6.2.3 Experiments on Real-World Multi-Class Datasets

To further evaluate the effectiveness of our PCSE in tackling real-world label noise, we conduct the experiments on five practical noisy datasets, namely

- *CIFAR-10N* [45]. This dataset contains the images from the original *CIFAR-10* dataset [24], and the labels are provided by crowd-sourced annotators. There are five noisy label sets in this dataset, namely "Random1", "Random2", "Random3", "Aggre", and "Worse".
- *CIFAR-100N* [45]. The collection process of this dataset is similar to that of *CIFAR-10N*. Each image in *CIFAR-100N* contains a coarse label and a fine label given by a human annotator, among which the fine labels for all images are adopted in our experiments.
- *Animal-10N* [40]. This dataset contains 55,000 images of 10 different animals, where 50,000 images are used for training, and the remaining 5,000 for testing. Since this dataset intentionally collects some pairs of visually confusing animals such as "cat" vs. "lynx" and "chimpanzee" vs. "orangutan", labeling errors would be naturally brought in during the annotation process.
- *Clothing-1M* [48]. This dataset contains 1 million clothing images belonging to 14 classes. 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.
- *WebVision* [28]. This dataset contains more than 2.4 million images crawled from the Flickr website and Google Images search. The same 1,000 concepts as the *ILSVRC12* dataset [8] are used for querying images. Due to the potential mismatch between the searched images and their corresponding concepts, label noise naturally exists.

For *CIFAR-10N* and *CIFAR-100N*, our adopted baseline methods and the corresponding experimental settings are the same as those on *CIFAR-10* and *CIFAR-100*. Detailed results are shown in Table 6, which reveals that our PCSE surpasses the baseline methods or achieves a similar performance. For example, PCSE surpasses CWD and MC-LDCE by 2.42% and 2.63%, respectively on *CIFAR-10N-*

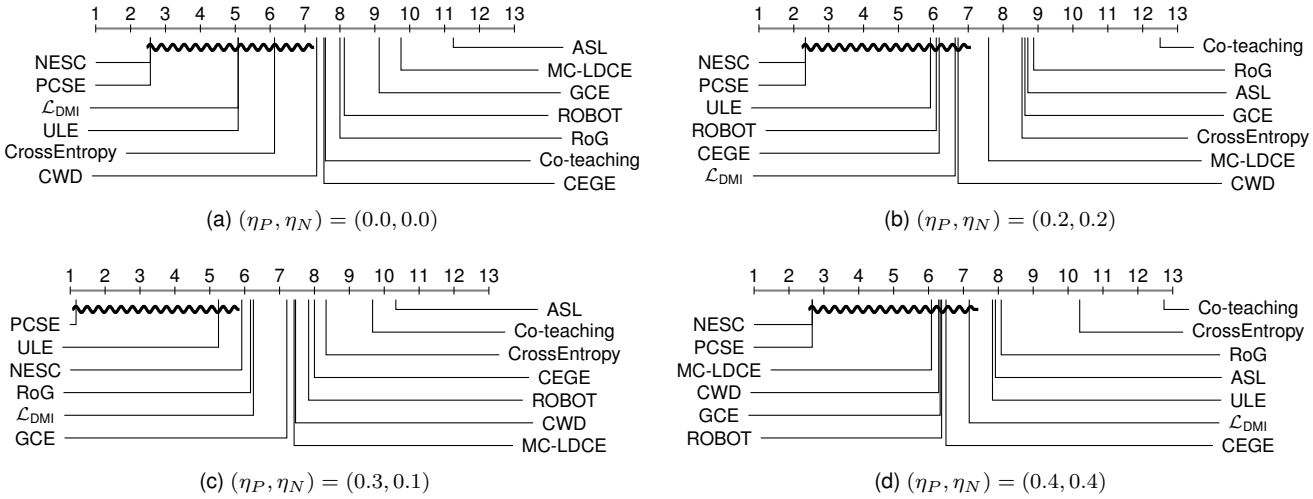


Fig. 4. Comparison of PCSE (the control algorithm) against twelve baseline methods with the Nemenyi test on the UCI benchmark datasets. Algorithms not connected with PCSE in the CD diagram are significantly inferior to PCSE (CD=4.8917 at 0.1 significance level).

TABLE 6
Comparison of mean test accuracy (%) of various methods on *CIFAR-10N* and *CIFAR-100N* datasets. The best two records on each dataset are highlighted in red and blue, respectively.

Dataset	CrossEntropy	GCE [57]	Co-teaching [18]	$\mathcal{L}_{\text{DMI}}$ [50]	CWD [15]	RoG [25]	MC-LDCE [9]	ASL [58]	ROBOT [52]	PCSE
CIFAR-10N-Aggre	87.77 ± 0.38	87.85 ± 0.70	91.20 ± 0.13	89.43 ± 0.11	89.60 ± 0.04	91.35 ± 0.07	89.39 ± 0.05	90.01 ± 0.37	91.35 ± 0.03	92.02 ± 0.13
CIFAR-10N-Random1	85.02 ± 0.65	87.61 ± 0.28	90.33 ± 0.13	87.27 ± 0.33	87.89 ± 0.33	90.48 ± 0.25	87.56 ± 0.53	88.68 ± 0.28	90.46 ± 0.18	91.19 ± 0.21
CIFAR-10N-Random2	86.46 ± 1.79	87.70 ± 0.56	90.30 ± 0.17	86.96 ± 0.21	87.56 ± 0.17	90.76 ± 0.18	87.51 ± 0.48	88.12 ± 0.21	90.37 ± 0.15	91.21 ± 0.14
CIFAR-10N-Random3	85.16 ± 0.61	87.58 ± 0.29	90.15 ± 0.18	87.11 ± 0.39	87.50 ± 0.19	90.37 ± 0.23	87.19 ± 0.33	88.81 ± 0.71	90.31 ± 0.21	91.13 ± 0.04
CIFAR-10N-Worse	77.69 ± 1.55	80.66 ± 0.35	83.83 ± 0.13	80.36 ± 0.19	80.43 ± 0.42	84.99 ± 0.11	79.84 ± 0.26	79.23 ± 0.70	84.05 ± 0.33	85.81 ± 0.20
CIFAR-100N	55.50 ± 0.66	56.73 ± 0.30	58.73 ± 0.26	50.54 ± 0.41	51.31 ± 1.46	58.49 ± 0.26	52.39 ± 0.73	58.17 ± 0.73	61.25± 0.26	59.75 ± 0.47

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

Dataset	Backbone	CrossEntropy	GCE [57]	Co-teaching [18]	$\mathcal{L}_{\text{DMI}}$ [50]	CWD [15]	RoG [25]	MC-LDCE [9]	ASL [58]	ROBOT [52]	PCSE
Animal-10N	VGG-19	79.42	76.21	83.06	80.62	82.52	83.28	81.20	77.70	83.52	83.82
	ResNet-18	81.71	81.17	84.86	82.46	83.48	85.04	84.30	82.56	84.68	85.48
Clothing-1M	ResNet-50	68.94	69.19	71.04	70.22	70.41	70.98	69.87	70.73	71.06	71.37

Aggre, which implies that estimating per-class statistics is more effective than estimating a single global centroid in handling label noise.

For *Animal-10N*, due to the prevalent use of ResNet-18 [20] and VGG-19 [39] as backbone networks in previous research [13], [40], we conducted experiments by using both types of networks to exclude the influence of different network architectures to the final performance. For all experiments, the batch size is set to 128, the learning rate is set to 0.001, and the Adam optimizer [23] with default parameters is used for model training. For *Clothing-1M*, ResNet-50 [20] is adopted as backbone network, as commonly used in [15] and [27]. For fairness of comparison, here we do not include the clean validation set in *Clothing-1M* during training. Other experimental settings are the same as those in [27]. The experimental results are presented in Table 7, which reveals that on *Animal-10N* and *Clothing-1M*, our PCSE outperforms all the compared methods when different types of backbone networks are employed.

For *WebVision*, ResNet-50 [20] is adopted as backbone network, and the other experimental settings are the same as [58]. In line with prior studies [27], [58], examples from the first 50 classes of the google image subset are used for training, and both validation sets from *WebVision* and

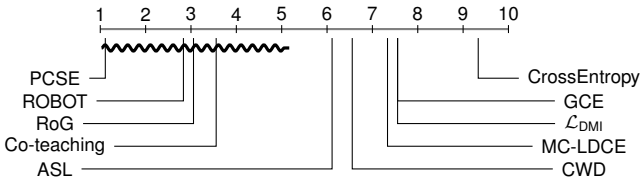


Fig. 5. Comparison of PCSE (the control algorithm) against seven baseline methods with the Nemenyi test on the real-world datasets. Algorithms not connected with PCSE in the CD diagram are significantly inferior to PCSE (CD = 4.1672 at 0.1 significance level).

ILSVRC12 are adopted for evaluation. We pre-train the network with Co-teaching for 100 epochs. The initial learning rate is set as 0.001 and is reduced by a factor of 10 after 50 epochs. The experimental results are provided in Table 8, which suggest that our PCSE can efficiently handle large-scale noisy datasets.

Furthermore, the Friedman test [7] and the Nemenyi test are also adopted for performance comparison. The Friedman statistic F_F and the corresponding critical value (at 0.1 significance level) are 63.00 and 2.92, respectively (with 10 algorithms and 9 datasets). Therefore, These methods are distinguishable in performance. Subsequently, the post-hoc Nemenyi test is conducted, and PCSE is regarded as the

TABLE 8
Comparison of various approaches on *WebVision*. Both validation sets from *WebVision* and *ILSVRC12* are adopted for evaluation, and test accuracies (%) are reported. The best two records on each dataset are highlighted in red and blue, respectively.

Validation set	CrossEntropy	GCE [57]	Co-teaching [18]	$\mathcal{L}_{\text{DMI}}$ [50]	CWD [15]	RoG [25]	MC-LDCE [9]	ASL [58]	ROBOT [52]	PCSE
WebVision	66.76	61.36	69.60	69.72	67.72	67.68	67.80	66.68	68.24	70.48
ILSVRC12	62.64	59.96	65.28	65.52	64.52	64.24	65.88	64.12	65.20	67.72

TABLE 9
Comparisons of running time (h). Experiments on *CIFAR-10N*, *CIFAR-100N*, and *Animal-10N* were conducted on a single Tesla V100 GPU, while experiments on *Clothing-1M* and *WebVision* were conducted on four Tesla V100 GPUs. The running time on *CIFAR-10N* is the averaged value over five label sets.

Dataset	CrossEntropy	GCE [57]	Co-teaching [18]	$\mathcal{L}_{\text{DMI}}$ [50]	CWD [15]	RoG [25]	MC-LDCE [9]	ASL [58]	ROBOT [52]	PCSE / Post-Process
CIFAR-10N	0.89	1.67	0.81	0.80	0.87	1.75	1.05	0.90	1.69	1.72 / 0.05
CIFAR-100N	0.90	1.68	0.82	0.85	0.91	1.73	1.35	0.91	1.66	1.71 / 0.03
Animal-10N	2.76	4.96	2.74	2.82	2.83	5.04	3.39	2.61	3.44	5.03 / 0.07
Clothing-1M	4.32	8.80	4.44	4.12	4.14	9.17	5.56	4.80	21.08	9.12 / 0.32
WebVision	4.14	7.75	4.15	4.20	4.29	8.33	4.45	3.47	14.89	8.24 / 0.49

control method. Fig. 5 shows the CD diagram. As illustrated in this figure, PCSE achieves the best average rank, and the performance of PCSE is significantly different from CWD and MC-LDCE, suggesting that local statistics within each class are more beneficial than a single global statistic in handling label noise.

In a word, the classification results on real-world noisy datasets clearly verify that PCSE is also effective in handling multi-class classification tasks with real-world label noise.

6.3 Comparisons of Running Time

In this section, we provide detailed comparisons of the running times of various methods. The experiments were conducted on the five real-world datasets. Besides, the experimental settings are the same as those in Section 6.2.2. We list the running time comparisons in Table 9. From this table, we can observe that the time consumed by our post-processing (after pre-training) is just few minutes on *CIFAR-10N* and *CIFAR-100N*. Moreover, on *Clothing-1M*, the post-processing of PCSE only consumes an additional 4% of the training time, which is negligible in practical applications.

7 Conclusion and Future Work

In this paper, we proposed a new method termed "Per-Class statistic estimation" (PCSE) to deal with multi-class label noise learning. Specifically, PCSE establishes the quantitative relationship between the per-class noisy first- and second-order statistics and the corresponding clean ones. The estimated statistics are further utilized to build a generative classifier for clean label inference. The advantages of our PCSE are three-fold:

- **Generality.** Our PCSE can be considered as a general post-processing strategy that can boost the classification performance of many DNNs pre-trained on the noisy training set. Additionally, it can handle both binary and multi-class classification problems.
- **Reliability.** Our proposed estimators of per-class statistics have theoretically guaranteed convergence, and the precision of the estimation on some key statistics has been empirically demonstrated.
- **Practicability.** Our proposed estimators do not rely on the error-prone clean sample selection process. Besides, our PCSE algorithm does not contain any tuning hyperparameters. Therefore, it can be easily implemented under various practical scenarios.

Due to the above reasons, our method has shown superior performance to various state-of-the-art LNL approaches on typical benchmark and real-world datasets.

In the future, we intend to study the problem of statistic estimation under instance-dependent label noise, where the label flipping process relies on not only the specific class but also the feature of instance.

References

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

[2] A. Asuncion and D. Newman, "UCI machine learning repository," 2007.

[3] Y. Bai, E. Yang, B. Han, Y. Yang, J. Li, Y. Mao, G. Niu, and T. Liu, "Understanding and improving early stopping for learning with noisy labels," *Advances in Neural Information Processing Systems*, vol. 34, pp. 24 392–24 403, 2021.

[4] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

[5] D. Cheng, Y. Ning, N. Wang, X. Gao, H. Yang, Y. Du, B. Han, and T. Liu, "Class-dependent label-noise learning with cycle-consistency regularization," in *Advances in Neural Information Processing Systems*, vol. 35, 2022, pp. 11 104–11 116.

[6] F. R. Cordeiro and G. Carneiro, "A survey on deep learning with noisy labels: How to train your model when you cannot trust on the annotations?" in *SIBGRAPI Conference on Graphics, Patterns and Images*. IEEE, 2020, pp. 9–16.

[7] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *Journal of Machine Learning Research*, vol. 7, no. 1, pp. 1–30, 2006.

[8] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 248–255.

[9] Y. Ding, T. Zhou, C. Zhang, Y. Luo, J. Tang, and C. Gong, "Multi-class label noise learning via loss decomposition and centroid estimation," in *SIAM International Conference on Data Mining*, 2022, pp. 253–261.

[10] 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 Conference on Artificial Intelligence*, 2020, pp. 2206–2212.

[11] W. Gao, L. Wang, Z.-H. Zhou *et al.*, "Risk minimization in the presence of label noise," in *AAAI Conference on Artificial Intelligence*, vol. 30, no. 1, 2016.

[12] W. Gao, T. Zhang, B.-B. Yang, and Z.-H. Zhou, "On the noise estimation statistics," *Artificial Intelligence*, vol. 293, p. 103451, 2021.

[13] A. Garg, C. Nguyen, R. Felix, T.-T. Do, and G. Carneiro, "Instance-dependent noisy label learning via graphical modelling," in *IEEE Winter Conference on Applications of Computer Vision*, 2023, pp. 2288–2298.

[14] A. Ghosh, H. Kumar, and P. S. Sastry, "Robust loss functions under label noise for deep neural networks," in *AAAI conference on artificial intelligence*, vol. 31, no. 1, 2017.

[15] C. Gong, Y. Ding, B. Han, G. Niu, J. Yang, J. You, D. Tao, and M. Sugiyama, "Class-wise denoising for robust learning under label noise," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 3, pp. 2835–2848, 2023.

[16] C. Gong, J. Yang, 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*, vol. 44, no. 6, pp. 2841–2855, 2020.

[17] 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.

[18] 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," *Advances in Neural Information Processing Systems*, vol. 31, pp. 8527–8537, 2018.

[19] T. Hastie and R. Tibshirani, "Discriminant analysis by Gaussian mixtures," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 58, no. 1, pp. 155–176, 1996.

[20] 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.

[21] R. A. Horn and C. R. Johnson, *Matrix Analysis*, 2nd ed. Cambridge University Press, 2013.

[22] 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*. PMLR, 2018, pp. 2304–2313.

[23] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *International Conference on Learning Representations*, 2014.

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

[25] 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*. PMLR, 2019, pp. 3763–3772.

[26] J. Li, M. Zhang, K. Xu, J. P. Dickerson, and J. Ba, "Noisy labels can induce good representations," *CoRR*, vol. abs/2012.12896, 2020.

[27] J. Li, R. Socher, and S. C. Hoi, "DivideMix: Learning with noisy labels as semi-supervised learning," in *International Conference on Learning Representations*, 2020.

[28] W. Li, L. Wang, W. Li, E. Agustsson, and L. Van Gool, "Webvision database: Visual learning and understanding from web data," *arXiv preprint arXiv:1708.02862*, 2017.

[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*. PMLR, 2021, pp. 6403–6413.

[30] 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, 2015.

[31] Y. Liu and H. Guo, "Peer loss functions: Learning from noisy labels without knowing noise rates," in *International Conference on Machine Learning*. PMLR, 2020, pp. 6226–6236.

[32] 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*. PMLR, 2020, pp. 6543–6553.

[33] N. Natarajan, I. S. Dhillon, P. K. Ravikumar, and A. Tewari, "Learning with noisy labels," *Advances in Neural Information Processing Systems*, vol. 26, pp. 1196–1204, 2013.

[34] K. Nishi, Y. Ding, A. Rich, and T. Hollerer, "Augmentation strategies for learning with noisy labels," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2021, pp. 8022–8031.

[35] C. G. Northcutt, A. Athalye, and J. Mueller, "Pervasive label errors in test sets destabilize machine learning benchmarks," in *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2021.

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

[37] 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.

[38] P. J. Rousseeuw and K. V. Driessen, "A fast algorithm for the minimum covariance determinant estimator," *Technometrics*, vol. 41, no. 3, pp. 212–223, 1999.

[39] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[40] H. Song, M. Kim, and J.-G. Lee, "SELFIE: Refurbishing unclean samples for robust deep learning," in *International Conference on Machine Learning*. PMLR, 2019, pp. 5907–5915.

[41] H. Song, M. Kim, D. Park, Y. Shin, and J.-G. Lee, "Learning from noisy labels with deep neural networks: A survey," *IEEE Transactions on Neural Networks and Learning Systems*, 2022.

[42] B. van Rooyen and R. C. Williamson, "A theory of learning with corrupted labels," *Journal of Machine Learning Research*, vol. 18, no. 228, pp. 1–50, 2018.

[43] 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.

[44] 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.

[45] J. Wei, Z. Zhu, H. Cheng, T. Liu, G. Niu, and Y. Liu, "Learning with noisy labels revisited: A study using real-world human annotations," in *International Conference on Learning Representations*, 2022.

[46] X. Xia, T. Liu, N. Wang, B. Han, C. Gong, G. Niu, and M. Sugiyama, "Are anchor points really indispensable in label-noise learning?" *Advances in Neural Information Processing Systems*, vol. 32, pp. 6835–6846, 2019.

[47] X. Xia, P. Lu, C. Gong, B. Han, J. Yu, and T. Liu, "Regularly truncated m-estimators for learning with noisy labels," *arXiv preprint arXiv:2309.00894*, 2023.

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

[49] M.-K. Xie and S.-J. Huang, "CCMN: A general framework for learning with class-conditional multi-label noise," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 1, pp. 154–166, 2022.

[50] Y. Xu, P. Cao, Y. Kong, and Y. Wang, "L_DMI: An information-theoretic noise-robust loss function," *arXiv preprint arXiv:1909.03388*, 2019.

[51] 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," *Advances in Neural Information Processing Systems*, vol. 33, pp. 7260–7271, 2020.

[52] L. Yong, R. Pi, W. ZHANG, X. Xia, J. Gao, X. Zhou, T. Liu, and B. Han, "A holistic view of label noise transition matrix in deep learning and beyond," in *International Conference on Learning Representations*, 2023.

[53] 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*. PMLR, 2019, pp. 7164–7173.

[54] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals, "Understanding deep learning (still) requires rethinking generalization," *Communications of the ACM*, vol. 64, no. 3, pp. 107–115, 2021.

[55] C. Zhang, L. Shen, J. Yang, and C. Gong, "Towards harnessing feature embedding for robust learning with noisy labels," *Machine Learning*, vol. 111, no. 9, pp. 3181–3201, 2022.

[56] Y. Zhang, G. Niu, and M. Sugiyama, "Learning noise transition matrix from only noisy labels via total variation regularization," in *International Conference on Machine Learning*. PMLR, 2021, pp. 12 501–12 512.

[57] 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.

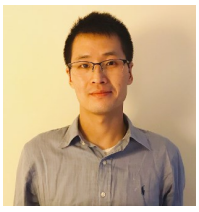
[58] X. Zhou, X. Liu, D. Zhai, J. Jiang, and X. Ji, "Asymmetric loss functions for noise-tolerant learning: Theory and applications," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 7, pp. 8094–8109, 2023.

[59] Z. Zhu, Y. Song, and Y. Liu, "Clusterability as an alternative to anchor points when learning with noisy labels," in *International Conference on Machine Learning*. PMLR, 2021, pp. 12 912–12 923.

Wenshui Luo received his bachelor degree from Nanjing University of Science and Technology (NJUST) in 2022. Currently, he is pursing master degree in NJUST under the supervision of Prof. Chen Gong. His research interests mainly lie in weakly-supervised learning.

Shuo Chen is currently a Research Scientist at RIKEN Center for Advanced Intelligence Project (RIKEN AIP). Before that, he was a Postdoctoral Researcher at RIKEN AIP from 2020 to 2023. He received his doctoral degree from Nanjing University of Science and Technology in 2020, and he was a CSC visiting student at the University of Pittsburgh from 2018 to 2019. His research interests mainly include machine learning and pattern recognition, in particular, contrastive learning and metric learning. He has published more than 40 technical papers at top-tier conferences such as ICML, NeurIPS, ICLR, CVPR, and prominent journals such as IEEE T-PAMI, IEEE T-IP, IEEE T-NNLS, etc. He has served as the Area Chair (AC) of NeurIPS, ICML, ICLR, CVPR, and ECCV. He won the "Excellent Achievement Award" of RIKEN, the "Excellent Doctoral Dissertation Award" of Chinese Institute of Electronics (CIE), and the "Excellent Doctoral Dissertation Nomination" of Chinese Association for Artificial Intelligence (CAAI).

Tongliang Liu (Senior Member, IEEE) is the Director of Sydney AI Centre at the University of Sydney. He is also a visiting professor at the University of Science and Technology of China, China; an affiliated professor with the Mohamed bin Zayed University of Artificial Intelligence, UAE; a visiting scientist with RIKEN AIP, Japan. He is broadly interested in the fields of trustworthy machine learning and its interdisciplinary applications, with a particular emphasis on learning with noisy labels, adversarial learning, causal representation learning, transfer learning, unsupervised learning, and statistical deep learning theory. He has authored and co-authored more than 200 research articles including ICML, NeurIPS, ICLR, CVPR, ICCV, ECCV, AAAI, IJCAI, T-PAMI, and JMLR. He is/was a senior meta reviewer for many conferences, such as NeurIPS, ICLR, AAAI, and IJCAI. He is a co-Editor-in-Chief for Neural Networks, an Associate Editor of IEEE T-PAMI, IEEE T-IP, TMLR, and ACM Computing Surveys, and is on the Editorial Boards of JMLR and MLJ. He is a recipient of CORE Award for Outstanding Research Contribution in 2024, the IEEE AI's 10 to Watch Award in 2022, the Future Fellowship Award from Australian Research Council (ARC) in 2022, the Top-40 Early Achievers by The Australian in 2020, and the Discovery Early Career Researcher Award (DECRA) from ARC in 2018.



Bo Han (Senior Member, IEEE) is currently an Assistant Professor in Machine Learning and a Director of Trustworthy Machine Learning and Reasoning Group at Hong Kong Baptist University, and a BAIHO Visiting Scientist at RIKEN Center for Advanced Intelligence Project (RIKEN AIP). He has served as Senior Area Chair of NeurIPS, and Area Chairs of NeurIPS, ICML and ICLR. He has also served as Associate Editors of IEEE T-PAMI, MLJ and JAIR, and Editorial Board Members of JMLR and MLJ. He received Outstanding Paper Award at NeurIPS, Most Influential Paper at NeurIPS, Notable Area Chair at NeurIPS, Outstanding Area Chair at ICLR, and Outstanding Associate Editor at IEEE T-NNLS. He received the RGC Early CAREER Scheme, NSFC General Program, IJCAI Early Career Spotlight, RIKEN BAIHO Award, Dean's Award for Outstanding Achievement, Microsoft Research StarTrack Program, and Faculty Research Awards from ByteDance, Baidu, Alibaba and Tencent.

Gang Niu (Senior Member, IEEE) is currently an indefinite-term senior 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, he was a senior software engineer at Baidu and then an assistant professor at the University of Tokyo. He joined RIKEN as a research scientist in 2018, and he was tenured in 2020 and promoted to senior research scientist in 2023. He published more than 100 journal articles and conference papers, including 39 ICML, 25 NeurIPS, and 14 ICLR (1 outstanding paper honorable mention) papers. He co-authored the book "Machine Learning from Weak Supervision: An Empirical Risk Minimization Approach" (the MIT Press). On the other hand, he served as a senior area chair/senior meta-reviewer 3 times and an area chair/meta-reviewer 25 times. He is also serving/served as an associate editor of IEEE T-PAMI, an action editor of TMLR, as well as an editorial board member and a guest editor of MLJ. Moreover, he served as a publication chair for ICML 2022, and co-organized 14 workshops, 1 competition, and 3 tutorials.

Masashi Sugiyama (Senior Member, IEEE) received his Ph.D. in Computer Science from Tokyo Institute of Technology, Japan, in 2001. After serving as an assistant and associate professor at the same institute, he became a professor at the University of Tokyo in 2014. Since 2016, he has also served as the director of the RIKEN Center for Advanced Intelligence Project. His research interests include theories and algorithms of machine learning. He was awarded the Japan Academy Medal in 2017 and the Commendation for Science and Technology by the Minister of Education, Culture, Sports, Science and Technology of Japan in 2022. He has also served as Associate Editor-in-Chief (AEIC) of IEEE T-PAMI.

Dacheng Tao (Fellow, IEEE) is currently a Distinguished University Professor in the College of Computing & Data Science at Nanyang Technological University. 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, with best paper awards, best student paper awards, and test-of-time awards. His publications have been cited over 131K times and he has an h-index 170+ in Google Scholar. He received the 2015 and 2020 Australian 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.

Chen Gong (Senior Member, IEEE) received his dual doctoral degree from Shanghai Jiao Tong University (SJTU) and University of Technology Sydney (UTS) in 2016 and 2017, respectively. Currently, he is a full professor in the School of Computer Science and Engineering, Nanjing University of Science and Technology. His research interests mainly include machine learning, data mining, and learning-based vision problems. He has published more than 130 technical papers at prominent journals and conferences such as JMLR, IEEE T-PAMI, IEEE T-NNLS, IEEE T-IP, IEEE T-CYB, IEEE T-CSVT, IEEE T-MM, IEEE T-ITS, ACM T-IST, ICML, NeurIPS, ICLR, CVPR, AAAI, IJCAI, ICDM, etc. He serves as the associate editor for IEEE T-CSVT and NePL, and also the Area Chair or Senior PC member of several top-tier conferences such as AAAI, IJCAI, ICLR, ECML-PKDD, AISTATS, ICDM, ACM MM, 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 Scientific Fund for Distinguished Young Scholars of Jiangsu Province. He was also selected as the "Global Top Chinese Young Scholars in AI" released by Baidu.

Estimating Per-Class Statistics for Label Noise Learning (Supplementary Material)

Wenshui Luo, Shuo Chen, Tongliang Liu, *Senior Member, IEEE,* Bo Han, *Senior Member, IEEE,* Gang Niu, *Senior Member, IEEE,* Masashi Sugiyama, *Senior Member, IEEE,* Dacheng Tao, *Fellow, IEEE,* Chen Gong, *Senior Member, IEEE*

CONTENTS

- *This research is supported by NSF of China (Nos: 62336003, 12371510, 62376235), NSF of Jiangsu Province (No: BZ2021013), NSF for Distinguished Young Scholar of Jiangsu Province (No: BK20220080), "111" Program (No: B13022), Guangdong Basic and Applied Basic Research Foundation (Nos: 2022A1515011652, 2024A1515012399), HKBU Faculty Niche Research Areas (No: RC-FNRA-IG/22-23/SCI/04), and HKBU CSD Departmental Incentive Grant. This work was done when Wenshui Luo was an intern at RIKEN.*
- *W. Luo and C. Gong are with the Key Laboratory of Intelligent Perception and Systems for High-Dimensional Information of Ministry of Education, 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, 210094, P.R. China.*
  E-mail: {randylo, chen.gong}@njust.edu.cn
- *S. Chen and G. Niu are with RIKEN Center for Advanced Intelligence Project, Tokyo, Japan.*
  E-mail: shuo.chen.ya@riken.jp; gang.niu.ml@gmail.com
- *T. Liu is with the School of Computer Science, Faculty of Engineering, the University of Sydney, Australia.*
  E-mail: tongliang.liu@sydney.edu.au
- *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
- *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
- *D. Tao is with Nanyang Technological University, Singapore.*
  E-mail: dacheng.tao@gmail.com
- *Corresponding authors: C. Gong and S. Chen.*

1 PROOF OF LEMMA 1

Proof. First of all, we derive the elements of $\mathbf{M}$. The (i,j)-th element of $\mathbf{M}$ is denoted as M_{ij}, which is obviously non-negative. For any $k \in [\![C]\!]$, we have

$$M_{kk} = \sum_{i,j=1}^{C} \mathbf{1}\{(i \neq k \wedge j \neq k \wedge i \neq j) \vee (i = j)\}\pi_i T_{ij}, \tag{1}$$

and for any $k, m \in [\![C]\!]$ with $k \neq m$, we have

$$M_{km} = \pi_k T_{km} + \pi_m T_{mk}. \tag{2}$$

Next, we consider the symmetric label noise. For any $k \in [\![C]\!]$, we have

$$\begin{aligned} M_{kk} - \sum_{m=1,m\neq k}^{C} M_{km} &= \sum_{i,j=1}^{C} \mathbf{1}\left\{(i \neq k \wedge j \neq k \wedge i \neq j) \vee (i = j)\right\} \cdot \pi_i T_{ij} - \sum_{m=1,m\neq k}^{C} (\pi_k T_{km} + \pi_m T_{mk}) \\ &= (1-\epsilon) + (1-\pi_k)\frac{C-2}{C-1}\epsilon - \frac{\epsilon}{C-1}[1 + (C-2)\pi_k] \\ &= \frac{1}{C-1} \cdot [(C-1) - \epsilon \cdot (2 + (2C-4)\pi_k)] \\ &\geq \frac{1}{C-1} \cdot \left[(C-1) - \epsilon \cdot \left(2 + (2C-4)\max_k \pi_k\right)\right] > 0, \end{aligned} \tag{3}$$

where the last inequality holds because we assume that $\epsilon < \min\left\{\frac{C-1}{\max_{i\in[\![C]\!]} \pi_i(2C-4)+2}, \frac{C-1}{C}\right\}$. As a result, $\mathbf{M}$ is a strictly diagonally dominant matrix [11]. Therefore, $\mathbf{M}$ is invertible under symmetric label noise.

For asymmetric label noise with uniform prior, when $\epsilon < \frac{1}{2}$, the transition matrix $\mathbf{T}$ is diagonally dominant. For any $k \in [\![C]\!]$, we have

$$\begin{aligned} M_{kk} - \sum_{m=1,m\neq k}^{C} M_{km} &= \sum_{i,j=1}^{C} \mathbf{1}\{(i \neq k \wedge j \neq k \wedge i \neq j) \vee (i = j)\} \cdot \pi_i T_{ij} - \sum_{m=1,m\neq k}^{C} (\pi_k T_{km} + \pi_m T_{mk}) \\ &= \sum_{i=1}^{C} \pi_i T_{ii} + \sum_{i\neq j, i\neq k, j\neq k} \pi_i T_{ij} - \sum_{m\neq k} (\pi_k T_{km} + \pi_m T_{mk}) \\ &= \frac{1}{C}\sum_{i=1}^{C} T_{ii} + \frac{1}{C}\sum_{i\neq j, i\neq k, j\neq k} T_{ij} - \frac{1}{C}\sum_{m\neq k}(T_{km} + T_{mk}). \end{aligned} \tag{4}$$

We also have that for any $i, k \in [\![C]\!]$ with $k \neq i$, $T_{ii} > \sum_{k\neq i} T_{ik} \geq T_{ik}$. Therefore, we have $\frac{1}{C}\sum_{i\neq k} T_{ii} > \frac{1}{C}\sum_{m\neq k} T_{mk}$, and $\frac{1}{C}T_{kk} > \frac{1}{C}\sum_{m\neq k} T_{km}$. Additionally, $\frac{1}{C}\sum_{i\neq j, i\neq k, j\neq k} T_{ij} > 0$, then for any $k \in [\![C]\!]$, we have $M_{kk} - \sum_{m\neq k} M_{km} > 0$. Consequently, $\mathbf{M}$ is strictly diagonally dominant, and thus it is invertible. Therefore, Lemma 1 is proved. □

2 VERIFICATION OF THE TWO ASSUMPTIONS

Sufficiently Scattered Assumption. Let the clean class posterior $\mathbf{P}(Y|X) = [P(Y = \mathbf{e}_1|X), P(Y = \mathbf{e}_2|X), \cdots, P(Y = \mathbf{e}_C|X)]^\top \in [0,1]^C$ and $\mathbf{H} = [\mathbf{P}(Y|X = \mathbf{x}_1), \cdots, \mathbf{P}(Y|X = \mathbf{x}_m)]$, where $\{\mathbf{x}_i\}_{i=1}^m$ is the set of all the instances. There are two aspects/conditions to be verified, namely

1) $\mathcal{Q} \subseteq$ *cone*$\{\mathbf{H}\}$, where $\mathcal{Q} = \{\mathbf{v} \in \mathbb{R}^C | \mathbf{v}^\top \mathbf{1} \geq \sqrt{C-1}\|\mathbf{v}\|_2\}$ and *cone*$\{\mathbf{H}\}$ is the convex cone combined by the column vectors of $\mathbf{H}$.
2) *cone*$\{\mathbf{H}\} \not\subseteq$ *cone*$\{\mathbf{U}\}$ for any unitary matrix $\mathbf{U} \in \mathbb{R}^{C\times C}$ that is not a permutation matrix.

For the convenience of presentation and discussion, we set the number of categories $C = 3$, and then $\mathbf{P}(Y|X) \in [0,1]^3$. To validate the existence of the vector set $\mathcal{H}$, we leverage training examples from three kinds of vehicles (namely, "airplane", "automobile", and "ship") in *CIFAR-10* as the training set for our setting. Since we only need to verify whether the clean posterior satisfies the above two conditions, the label noise is not injected to the training set. Here, naive training with cross-entropy loss is adopted for sake of its simplicity. Moreover, to avoid overfitting, we only use the training set to fit the posterior distribution, and then we use the test examples from the test set of *CIFAR-10* to verify the two conditions mentioned above.

Since the probability simplex formed by $\mathbf{P}(Y|X)$ in 3-dimensional space is a plane, we can restrict our study to this plane instead of the whole 3-dimensional space. To achieve this point, we project the simplex $\mathcal{S} = \{\mathbf{v} = [v_1, v_2, v_3]^\top \,|\, \mathbf{v}^\top \mathbf{1} = 1, v_i \geq 0, \forall i \in \{1,2,3\}\}$ to a 2-dimensional space. An illustration of the Sufficiently Scattered Assumption is provided in Fig. 1, where all the test examples constitute $\mathcal{H}$ in Fig. 1(b), and 15 randomly selected test examples constitute $\mathcal{H}$ in Fig. 1(c).

We justify the two conditions in this assumption by calculating their respective probabilities of occurrence, and the experimental results are provided in Table 1. For the first condition, we calculate the ratio of areas, namely $r_1 = |\mathcal{Q} \cap$

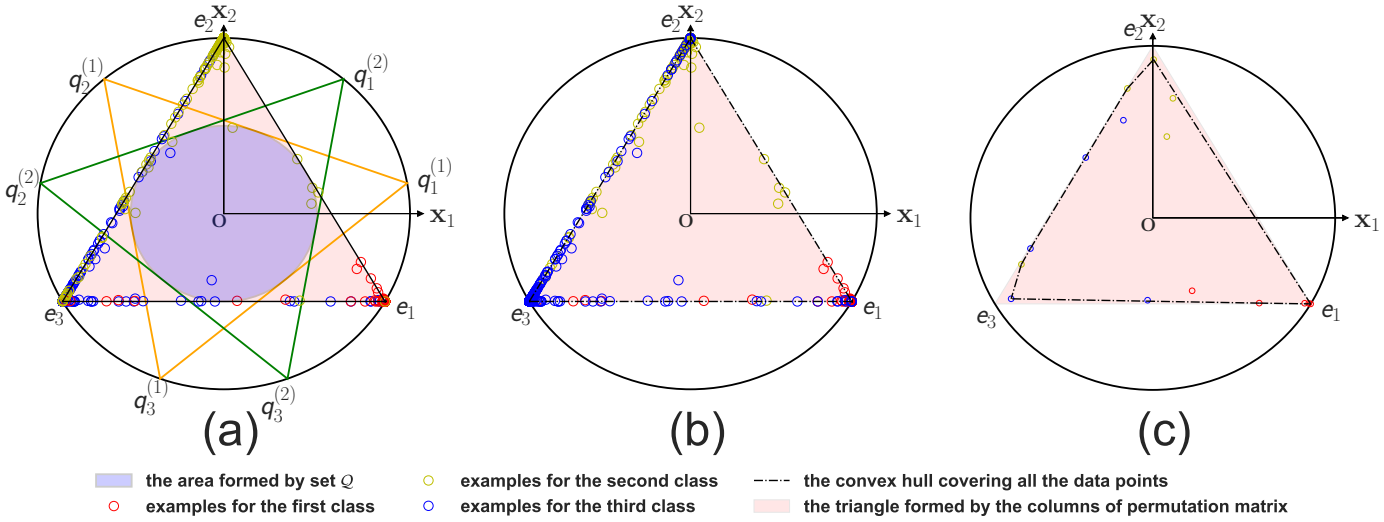


Fig. 1. Illustration of the Sufficiently Scattered Assumption by assuming that $C = 3$ and the viewers are facing the 3-dimensional simplex from the positive orthant. The data points belong to three kinds of vehicles in *CIFAR-10* [13]. In (a), a total of 3000 test examples are plotted (1000 examples for each class); the shaded area is formed by the columns of the permutation matrix; the green and orange triangles are formed by unitary matrices; and the purple circle corresponds to the space $\mathcal{Q}$. In (b), the convex hull formed by the data points is represented by the dashed line. In (c), a total of 15 examples are plotted, with 5 examples per class, and the anchor points are missing for the second and third classes.

cone$\{\mathbf{H}\}|/|\mathcal{Q}|$, to explore the proportion of elements in $\mathcal{Q}$ that is covered by *cone*$\{\mathbf{H}\}$. Through computation, we find that $r_1 > 99.8\%$ for the case in Fig. 1(b) and that $r_1 > 99.1\%$ for the case in Fig. 1(c). Therefore, the first condition holds almost surely. For the second condition, the discretization technique is utilized to enumerate the possible unitary matrices. Specifically, we set $\theta_i = -\frac{\pi}{6} + \frac{i}{M}[\frac{\pi}{2} - (-\frac{\pi}{6})]$ with $i \in \{0, 1, 2, \cdots, M-1\}$, where M is the total number of triangles/unitary matrices. With this approximation, the error is $\frac{2\pi}{3 \cdot M}$. We set $M = 1000$, and then the approximation error is about 0.002. Through computation, we identify that the only $\mathbf{Q}^{(i)}$ that satisfies this condition is $[\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3]$, which is exactly the permutation matrix. There are a total of 1,000 cases in our computation, and with approximation error $\frac{2\pi}{3000}$, the only unitary matrix that satisfies the second condition is the permutation matrix. Since we can obtain such results in both cases of Fig. 1(b) and Fig. 1(c), the second condition can be satisfied even when a small number of examples are investigated, i.e., m is quite small in the assumption.

Combining the justifications of the two conditions, we conclude that the *Sufficiently Scattered Assumption* can be easily satisfied in practice.

Invertibility of the noise transition matrix $\mathbf{T}$. This assumption is widely adopted in literature [5], [19], [25]. Next, we provide more evidences to show its rationality.

1) According to [1] (Lemma 1) and [18], the binary classification task (with label noise) is learnable only under the condition $\eta_P + \eta_N < 1$, where η_P and η_N are the label flip probabilities (defined in Section 5.2). This condition is equivalent to the invertibility of $\mathbf{T}$ under binary classification scenarios.
2) For any instance $\mathbf{x}$, we have $\mathbf{P}(\widetilde{Y} \,|\, X = \mathbf{x}) = \mathbf{T}^\top \mathbf{P}(Y \,|\, X = \mathbf{x})$. Suppose that $\mathbf{T}$ is not invertible, given any noisy posterior $\mathbf{P}(\widetilde{Y} \,|\, X = \mathbf{x})$, the clean posterior $\mathbf{P}(Y \,|\, X = \mathbf{x})$ is not unique (if it exists). Therefore, we cannot identify the unique clean posterior distribution even if the precise noisy posterior distribution is provided. Therefore, the invertibility is required to ensure the identifiability of both $\mathbf{T}$ and $\mathbf{P}(Y \,|\, X = \mathbf{x})$, as also assumed in [15] (Definition 4).
3) In practical datasets, it is reasonable to anticipate that the noise transition matrix is diagonally dominant. For instance, in early work [3] and recent work [2], $P(\widetilde{Y} = \mathbf{e}_i \,|\, Y = \mathbf{e}_i) > 0.5$ is assumed for $\forall i \in \{1, 2, \cdots, C\}$. We further take the human annotated noisy dataset *CIFAR-10N-worst* for example, which contains the worst-case annotations. In its (estimated) noise transition matrix (we can find it in the "Observations" tag at http://noisylabels.com/), the diagonal element is significantly larger than the off-diagonal elements in each row. To name a few, in the first and second rows, the diagonal elements (0.65 and 0.59, respectively) surpass the corresponding second highest off-diagonal elements (0.08 and 0.25), thereby establishing their dominance and ensuring the invertibility of the transition matrix $\mathbf{T}$.

In summary, the second assumption is a necessity and it can be easily satisfied in practice.

3 Extension of PCSE to Class-conditional Multi-label Noise

In this section, we briefly show that the proposed PCSE method can potentially be applied to the learning with multi-label noise problem. Before this, we formally define the problem of Class-Conditional Multi-label Noise (CCMN for short).

TABLE 1
Verification results of the Sufficiently Scattered Assumption.

Figure	Condition	Probability	Approximation error
Fig. 1(b) ($\lvert\mathcal{H}\rvert = 3000$)	$\mathcal{Q} \subseteq cone\{\mathbf{H}\}$	99.8%	0
	$cone\{\mathbf{H}\} \nsubseteq cone\{\mathbf{U}\}$	100%	$2\pi/3000$
Fig. 1(c) ($\lvert\mathcal{H}\rvert = 15$)	$\mathcal{Q} \subseteq cone\{\mathbf{H}\}$	99.1%	0
	$cone\{\mathbf{H}\} \nsubseteq cone\{\mathbf{U}\}$	100%	$2\pi/3000$

For any instance $\mathbf{x} \in \mathcal{X}$, we follow [20] and denote by $\mathcal{Y}' = \{-1, +1\}^C$ the label space. For any given label $\mathbf{y} \in \mathcal{Y}'$, we use y_j to denote the j-th element of $\mathbf{y}$. In our setting, $y_j = 1$ indicates the j-th label is a true label for the instance $\mathbf{x}$, while $y_j = -1$ indicates the opposite. Since we are tackling the multi-label classification problem, $\sum_{j=1}^{C} \mathbf{1}\{y_j = 1\} \geq 1$ holds for any instance $\mathbf{x}$. Let $S = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^n$ be the given training dataset drawn i.i.d. according to the true distribution $\mathcal{D}$. In the CCMN framework, the true label $\mathbf{y}$ can be flipped into the noisy one, namely $\widetilde{y}$. The flipping process follows a class-conditional noise model as: $P(\widetilde{y}_j = -1 | y_j = +1) = \rho_{+1}^j$, $P(\widetilde{y}_j = +1 | y_j = -1) = \rho_{-1}^j$, and $\forall j \in [\![C]\!], \rho_{+1}^j + \rho_{-1}^j < 1$. With such a contamination process, the noisy distribution is denoted by $\widetilde{\mathcal{D}}$, and the noisy dataset based on S is denoted by $\widetilde{S} = \{(\mathbf{x}_i, \widetilde{\mathbf{y}}_i)\}_{i=1}^n$. In the following, we use $\widetilde{y}_{ij}$ to denote the j-th element of $\widetilde{\mathbf{y}}_i$.

Now, we elaborate on the calibration method of the pre-trained classifier for the CCMN setting based on PCSE. From a high-level point of view, we decompose the CCMN problem into several independent binary classification problems (as previous research did [20]) and calibrate the classifier for each class by leveraging the PCSE method. Suppose that the pre-trained classifier for the j-th class is denoted by $f_j(\mathbf{x}) = \mathbf{w}_j^\top \phi(\mathbf{x}) + b_j$, where $\mathbf{w}_j \in \mathbb{R}^d$ and $\phi(\mathbf{x}) \in \mathbb{R}^d$ are the parameters and learned feature representation for $\mathbf{x}$, respectively. Besides, d denotes the feature dimensionality. Let

$$\mathbf{T}^j = \begin{bmatrix} 1 - \rho_{-1}^j & \rho_{-1}^j \\ \rho_{+1}^j & 1 - \rho_{+1}^j \end{bmatrix} \tag{5}$$

be the label transition matrix for the j-th class, whose (i,k)-th element is denoted by T_{ik}^j. Then, we can obtain the coefficient matrix $\mathbf{M}^j$ for the j-th class, which is given by

$$\mathbf{M}^j = \sum_{i \in \{0,1\}} \sum_{k \in \{0,1\}} \pi_i^j T_{ik}^j \mathbf{K}_{i \to k}, \tag{6}$$

where $\pi_0^j = P(y_j = -1)$ and $\pi_1^j = P(y_j = +1)$ are the class priors, $\mathbf{K}_{i \to k}$ is the 2-dimensional permutation matrix formed by switching the i-th and k-th rows of the 2-dimensional identity matrix. With such a coefficient matrix, we can readily obtain the estimators for the mean and covariance of the j-th class (by leveraging the similar derivation in PCSE). We denote such estimators by $\boldsymbol{\mu}_{-1}^j, \boldsymbol{\mu}_{+1}^j$ and $\boldsymbol{\Sigma}_{-1}^j, \boldsymbol{\Sigma}_{+1}^j$, respectively, where the subscripts $+1$ and -1 correspond to the positive and negative labels, respectively. Therefore, the overall covariance matrix for the j-th class is denoted by $\boldsymbol{\Sigma}^j = \pi_0^j \boldsymbol{\Sigma}_{-1}^j + \pi_1^j \boldsymbol{\Sigma}_{+1}^j$. Subsequently, the generative classifier based on $\phi(\mathbf{x})$ is given by $f_j'(\mathbf{x}) = \mathbf{w}_j'^\top \phi(\mathbf{x}) + b_j'$, where

$$\begin{cases} \mathbf{w}_j' = \boldsymbol{\Sigma}^j (\boldsymbol{\mu}_{+1}^j - \boldsymbol{\mu}_{-1}^j) \\ b_j' = -\frac{1}{2} \boldsymbol{\mu}_{+1}^{j} (\boldsymbol{\Sigma}^j)^{-1} \boldsymbol{\mu}_{+1}^j + \frac{1}{2} \boldsymbol{\mu}_{-1}^{j} (\boldsymbol{\Sigma}^j)^{-1} \boldsymbol{\mu}_{-1}^j + \log \frac{\pi_1^j}{\pi_0^j} \end{cases} . \tag{7}$$

Based on this calibrated classifier, the predicted label of the j-th class for an instance $\mathbf{x}$ is $\widehat{y}_j = 2 \cdot \mathbf{1}\{f_j'(\mathbf{x}) > 0\} - 1$. Therefore, we justify that our PCSE can be potentially applied to the CCMN problem. However, there are some difficulties in practical implementation. We have listed some tough points as follows:

1) The accurate estimation of the noise rates $\{\rho_{-1}^j\}_{j=1}^C$ and $\{\rho_{+1}^j\}_{j=1}^C$ is a hard task, since anchor points may be missing for multi-label datasets and the accurate estimations of those noisy posterior probabilities are intrinsically hard.
2) The examples with $y_j = +1$ and $y_j = -1$, $\forall j \in \{1, 2, \cdots, C\}$ may not follow the unimodal Gaussian distribution, and the assumption that examples for the j-th class follow the Gaussian distribution for all $j \in \{1, 2, \cdots, C\}$ may not hold necessarily, since the distributions for all the classes can be highly entangled. For example, we suppose that the examples with $y_j = +1$ follow the unimodal Gaussian distribution, and then some of the examples with $y_j = +1$ may also have $y_k = +1$ or $y_k = -1$ for some k. In this case, the features may be indistinguishable for the k-th class. However, the pre-trained feature space always shows clustering properties (*i.e.*, distinguishable for different classes), which possibly lead to a contradiction.
3) The appearances of some classes in multi-label datasets may be less frequent, and some of the binary problems may have imbalanced data. In our analysis (Sectioin 5.1 of the revised manuscript), more severe the class imbalance is, the larger the error upper bound will be. Therefore, it is still unclear that the final estimation can exactly recover

the true mean & covariance.

4 Proof of Theorem 1

Proof. First of all, we assume that the ground-truth transition matrix $\mathbf{T}$ is given. Since $\left|\frac{1}{\widetilde{n}}\mathbf{1}\{\widetilde{\mathbf{y}}_i = \mathbf{e}_k\} - \frac{1}{\widetilde{n}}\mathbf{1}\{\widetilde{\mathbf{y}}'_i = \mathbf{e}_k\}\right| \leq \frac{1}{\widetilde{n}}$ for any pair of $(\widetilde{\mathbf{y}}_i, \widetilde{\mathbf{y}}'_i)$, by using McDiarmid's inequality [16], for any $k \in [\![C]\!]$ and any $\epsilon > 0$, we have

$$P\left(\left|\widehat{\widetilde{\pi}}_k - \widetilde{\pi}_k\right| \geq \epsilon\right) \leq 2\exp\left(-2\widetilde{n}\epsilon^2\right). \tag{8}$$

For the multi-dimensional case, we have

$$P\left(\left\{\exists k \in [\![C]\!] : \left|\widehat{\widetilde{\pi}}_k - \widetilde{\pi}_k\right| \geq \epsilon\right\}\right) \leq 2C\exp\left(-2\widetilde{n}\epsilon^2\right). \tag{9}$$

Then by negation, we have

$$P\left(\left\{\forall k \in [\![C]\!] : \left|\widehat{\widetilde{\pi}}_k - \widetilde{\pi}_k\right| \leq \epsilon\right\}\right) \geq 1 - 2C\exp\left(-2\widetilde{n}\epsilon^2\right). \tag{10}$$

We denote the estimated noisy class prior and the ground-truth noisy class prior by $\widehat{\widetilde{\boldsymbol{\pi}}} = \left[\widehat{\widetilde{\pi}}_1, \widehat{\widetilde{\pi}}_2, \cdots, \widehat{\widetilde{\pi}}_C\right]^\top$ and $\widetilde{\boldsymbol{\pi}} = \left[\widetilde{\pi}_1, \widetilde{\pi}_2, \cdots, \widetilde{\pi}_C\right]^\top$, respectively. Besides, we use $\|\mathbf{v}\|_\infty$ to denote the ℓ_∞-norm of a vector $\mathbf{v}$, which is defined as $\|\mathbf{v}\|_\infty = \max_i |v_i|$ with v_i being the i-th element of $\mathbf{v}$. Then Eq. (10) implies that for any $\delta > 0$ with probability at least $1 - \delta$, we have

$$\left\|\widehat{\widetilde{\boldsymbol{\pi}}} - \widetilde{\boldsymbol{\pi}}\right\|_\infty \leq \sqrt{\frac{1}{2\widetilde{n}}\log\frac{2C}{\delta}}. \tag{11}$$

Now we consider the estimation error of the clean class prior $\boldsymbol{\pi} = [\pi_1, \pi_2, \cdots, \pi_C]^\top$. We also denote its estimated value by $\widehat{\boldsymbol{\pi}} = [\widehat{\pi}_1, \widehat{\pi}_2, \cdots, \widehat{\pi}_C]^\top$. Then, for any $\delta > 0$, with probability at least $1 - \delta$, we have

$$\|\widehat{\boldsymbol{\pi}} - \boldsymbol{\pi}\|_\infty \leq \left\|(\mathbf{T}^\top)^{-1}\right\|_\infty \left\|\widehat{\widetilde{\boldsymbol{\pi}}} - \widetilde{\boldsymbol{\pi}}\right\|_\infty \leq \left\|\mathbf{T}^{-1}\right\|_1 \sqrt{\frac{1}{2\widetilde{n}}\log\frac{2C}{\delta}}, \tag{12}$$

where we have $\|\mathbf{A}\|_\infty = \max_i \|\mathbf{A}_{i\cdot}\|_1$, and $\|\mathbf{A}\|_1 = \max_i \|\mathbf{A}_{\cdot i}\|_1$ for a matrix $\mathbf{A}$. Here $\mathbf{A}_{i\cdot}$ represents the i-th row of the matrix $\mathbf{A}$, and $\mathbf{A}_{\cdot i}$ stands for the i-th column of the matrix $\mathbf{A}$. The last inequality in Eq. (12) holds because $\left\|(\mathbf{T}^\top)^{-1}\right\|_\infty = \left\|\mathbf{T}^{-1}\right\|_1$.

Since the analysis of the first- and second-order statistics is similar, here we only prove the case of first-order statistics, and the case of second-order statistics holds in the same way. For brevity, we omit the superscript "(1)" in $\overline{X}^{(1)}$, $\widetilde{\boldsymbol{u}}_k^{(1)}$ and other symbols related to the first-order statistics. Moreover, we use ψ to denote ψ_1. We define $\widehat{\widetilde{\boldsymbol{u}}}_k = \frac{1}{\widetilde{n}_k}\sum_{i\in[\![n]\!], \widetilde{\mathbf{y}}_i = \mathbf{e}_k} \psi(\mathbf{x}_i)$ and $\widetilde{\boldsymbol{u}}_k = \mathbb{E}_{X|\widetilde{Y}=\mathbf{e}_k}[\psi(X)]$ for any $k \in [\![C]\!]$. Besides, we denote $\widetilde{\mathbf{U}} = [\widetilde{\boldsymbol{u}}_1, \widetilde{\boldsymbol{u}}_2, \cdots, \widetilde{\boldsymbol{u}}_C]$ and $\widehat{\widetilde{\mathbf{U}}} = [\widehat{\widetilde{\boldsymbol{u}}}_1, \widehat{\widetilde{\boldsymbol{u}}}_2, \cdots, \widehat{\widetilde{\boldsymbol{u}}}_C]$. Next, we bound the estimation error of the noisy first-order statistics, namely $\left\|\widehat{\widetilde{\mathbf{U}}} - \widetilde{\mathbf{U}}\right\|_2$. By leveraging the McDiarmid's inequality [16], for any $\delta > 0$ and $k \in [\![C]\!]$, with probability at least $1 - \delta$, we have

$$P\left(\left\|\widehat{\widetilde{\boldsymbol{u}}}_k - \widetilde{\boldsymbol{u}}_k\right\|_\infty \geq \epsilon\right) \leq 2d\exp\left(-\frac{\widetilde{n}_k\epsilon^2}{2\overline{X}^2}\right) \leq 2d\exp\left(-\frac{\min_k \widetilde{n}_k\epsilon^2}{2\overline{X}^2}\right). \tag{13}$$

Then by using the union bound inequality, we have that for any $\epsilon > 0$,

$$P\left(\left\{\exists k \in [\![C]\!] : \left\|\widehat{\widetilde{\boldsymbol{u}}}_k - \widetilde{\boldsymbol{u}}_k\right\|_\infty \geq \epsilon\right\}\right) \leq \sum_{k=1}^{C} P\left(\left\|\widehat{\widetilde{\boldsymbol{u}}}_k - \widetilde{\boldsymbol{u}}_k\right\|_\infty \geq \epsilon\right) \leq 2dC\exp\left(-\frac{\min_k \widetilde{n}_k\epsilon^2}{2\overline{X}^2}\right), \tag{14}$$

which implies that

$$P\left(\left\|\widehat{\widetilde{\mathbf{U}}} - \widetilde{\mathbf{U}}\right\|_{\max} \leq \epsilon\right) = P\left(\left\{\forall k \in [\![C]\!] : \left\|\widehat{\widetilde{\boldsymbol{u}}}_k - \widetilde{\boldsymbol{u}}_k\right\|_\infty \leq \epsilon\right\}\right) \geq 1 - 2dC\exp\left(-\frac{\min_k \widetilde{n}_k\epsilon^2}{2\overline{X}^2}\right), \tag{15}$$

where $\left\|\widehat{\widetilde{\mathbf{U}}} - \widetilde{\mathbf{U}}\right\|_{\max} = \max_{i,j}\left|\widehat{\widetilde{\mathbf{U}}}_{ij} - \widetilde{\mathbf{U}}_{ij}\right|$. Therefore, for any $\delta > 0$, with probability at least $1 - \delta$, we have

$$\left\|\widehat{\widetilde{\mathbf{U}}} - \widetilde{\mathbf{U}}\right\|_2 \leq \sqrt{dC}\left\|\widehat{\widetilde{\mathbf{U}}} - \widetilde{\mathbf{U}}\right\|_{\max} \leq \overline{X}\sqrt{dC}\sqrt{\frac{2}{\min_k \widetilde{n}_k}\log\frac{2dC}{\delta}}. \tag{16}$$

Subsequently, we bound the estimation error of each diagonal matrix. We first investigate the estimation error bound of $\boldsymbol{\Lambda}^{-1} = \mathrm{diag}(\boldsymbol{\pi})^{-1}$. Let $\widehat{\boldsymbol{\Lambda}}^{-1} = \mathrm{diag}(\widehat{\boldsymbol{\pi}})^{-1}$, and then for any $\delta > 0$, with probability at least $1 - \delta$, we have

$$\left\|\widehat{\boldsymbol{\Lambda}}^{-1} - \boldsymbol{\Lambda}^{-1}\right\|_2 \leq \frac{\|\boldsymbol{\pi} - \widehat{\boldsymbol{\pi}}\|_\infty}{\min_i \widehat{\pi}_i \min_i \pi_i} \leq \frac{\|\mathbf{T}^{-1}\|_1}{\min_i \widehat{\pi}_i \min_i \pi_i}\sqrt{\frac{1}{2\widetilde{n}}\log\frac{2C}{\delta}}. \tag{17}$$

Next, we consider the estimation error bound of $\widetilde{\mathbf{\Lambda}} = \text{diag}(\widetilde{\boldsymbol{\pi}})$. We further denote $\widehat{\widetilde{\mathbf{\Lambda}}} = \text{diag}(\widehat{\widetilde{\boldsymbol{\pi}}})$. Then, for any $\delta > 0$, with probability at least $1 - \delta$, we have

$$\left\|\widehat{\widetilde{\mathbf{\Lambda}}} - \tilde{\mathbf{\Lambda}}\right\|_2 \leq \left\|\widehat{\widetilde{\boldsymbol{\pi}}} - \widetilde{\boldsymbol{\pi}}\right\|_\infty \leq \sqrt{\frac{1}{2\widetilde{n}} \log \frac{2C}{\delta}}. \tag{18}$$

We denote the estimation error of $\mathbf{M}$ by $\delta\mathbf{M} = \widehat{\mathbf{M}} - \mathbf{M}$, where $\widehat{\mathbf{M}}$ is the estimated value of $\mathbf{M}$. Since for any permutation matrix, its maximum singular value is 1, we have $\|\mathbf{K}_{i\to j}\|_2 = 1$ for $\forall i, j \in [\![C]\!]$, then

$$\begin{aligned}
\left\|\mathbf{M} - \widehat{\mathbf{M}}\right\|_2 &= \left\| \sum_{i,j=1}^{C} \pi_i T_{ij} \mathbf{K}_{i\to j} - \sum_{i,j=1}^{C} \widehat{\pi}_i T_{ij} \mathbf{K}_{i\to j} \right\|_2 \\
&\leq \sum_{i,j=1}^{C} |\pi_i - \widehat{\pi}_i| \cdot T_{ij} \cdot \|\mathbf{K}_{i\to j}\|_2 \\
&= \sum_{i=1}^{C} |\pi_i - \widehat{\pi}_i| \sum_{j=1}^{C} T_{ij} \\
&= \|\boldsymbol{\pi} - \widehat{\boldsymbol{\pi}}\|_1.
\end{aligned} \tag{19}$$

Therefore, for any $\delta > 0$, with probability at least $1 - \delta$, we have

$$\|\delta\mathbf{M}\|_2 = \|\mathbf{M} - \widehat{\mathbf{M}}\|_2 \leq \|\boldsymbol{\pi} - \widehat{\boldsymbol{\pi}}\|_1 \leq C\|\boldsymbol{\pi} - \widehat{\boldsymbol{\pi}}\|_\infty \leq \|\mathbf{T}^{-1}\|_1 \cdot C\sqrt{\frac{1}{2\widetilde{n}} \log \frac{2C}{\delta}}. \tag{20}$$

If we further assume $\widetilde{n} > 2 \cdot C^2 \|\mathbf{T}^{-1}\|_1^2 \log \frac{2C}{\delta}$, then we have $1 - \|\mathbf{T}^{-1}\|_1 \cdot C\sqrt{\frac{1}{2\widetilde{n}} \log \frac{2C}{\delta}} > \frac{1}{2}$ for any $\delta > 0$. Therefore, by using the estimation error of the inverse of a matrix, with probability at least $1 - \delta$, we have

$$\begin{aligned}
\left\|\widehat{\mathbf{M}}^{-1} - \mathbf{M}^{-1}\right\|_2 &\leq \frac{\|\delta\mathbf{M}\|_2}{1 - \|\delta\mathbf{M}\|_2} \\
&\leq \frac{\|\mathbf{T}^{-1}\|_1 \cdot C\sqrt{\frac{1}{2\widetilde{n}} \log \frac{2C}{\delta}}}{1 - \|\mathbf{T}^{-1}\|_1 \cdot C\sqrt{\frac{1}{2\widetilde{n}} \log \frac{2C}{\delta}}} \\
&\leq 2\|\mathbf{T}^{-1}\|_1 \cdot C\sqrt{\frac{1}{2\widetilde{n}} \log \frac{2C}{\delta}}.
\end{aligned} \tag{21}$$

Subsequently, we derive the bounds of norms for each matrix. For the matrix $\mathbf{M}$, we have $\|\widehat{\mathbf{M}}\|_2 \leq 1$ and $\|\mathbf{M}\|_2 \leq 1$. Since $\|\mathbf{M}\|_2 > \max_{k,m} M_{km}$, and $0 < M_{km} < 2$ for any $k, m \in [\![C]\!]$, $\max_{k,m} M_{km}$ always exists. Therefore, we denote $\max\{\frac{1}{\max_{k,m} M_{km}}, \frac{1}{\max_{k,m} \widehat{M}_{km}}\}$ by ξ_M, which is a positive constant. Since $\|\mathbf{M}\|_2 \geq \|\mathbf{M}\|_{\max} = \max_{k,m} M_{km}$, we can respectively bound the norms of $\widehat{\mathbf{M}}^{-1}$ and $\mathbf{M}^{-1}$ by

$$\left\|\widehat{\mathbf{M}}^{-1}\right\|_2 = \frac{\text{cond}_2(\widehat{\mathbf{M}})}{\|\widehat{\mathbf{M}}\|_2} \leq \xi_M \cdot \text{cond}_2(\widehat{\mathbf{M}}), \tag{22}$$

and

$$\left\|\mathbf{M}^{-1}\right\|_2 = \frac{\text{cond}_2(\mathbf{M})}{\|\mathbf{M}\|_2} \leq \xi_M \cdot \text{cond}_2(\mathbf{M}). \tag{23}$$

Additionally, we can bound the diagonal matrices and the estimated noisy statistics by

$$\begin{cases} \|\widetilde{\mathbf{\Lambda}}\|_2 = \|\widetilde{\boldsymbol{\pi}}\|_\infty = \max_i \widetilde{\pi}_i \leq 1, \\ \|\mathbf{\Lambda}^{-1}\|_2 = \frac{1}{\min_i \pi_i}, \end{cases} \tag{24}$$

and

$$\left\|\widehat{\widetilde{\mathbf{U}}}\right\|_2 \leq \left\|\widehat{\widetilde{\mathbf{U}}}\right\|_{\text{F}} = \sqrt{\sum_i \left\| \frac{1}{\widetilde{n}_i} \sum_j \psi(\mathbf{x}_j^{(i)}) \right\|_2^2} \leq \sqrt{C} \cdot \overline{X}, \tag{25}$$

respectively, where $\mathbf{x}_j^{(i)}$ is an example for the i-th class, $\widetilde{n}_i$ is the number of examples for the i-th class, and $\|\mathbf{U}\|_{\text{F}} = \sqrt{\sum_{i,j} U_{ij}^2}$ for a matrix $\mathbf{U}$ with U_{ij} being the (i, j)-th element of $\mathbf{U}$.

Now we can bound the estimation error of the clean statistic by

$$\begin{aligned}\left\|\mathbf{U}-\widehat{\mathbf{U}}\right\|_2 &= \left\|\widetilde{\mathbf{U}}\widetilde{\mathbf{\Lambda}}\mathbf{M}^{-1}\mathbf{\Lambda}^{-1}-\widehat{\widetilde{\mathbf{U}}}\widehat{\widetilde{\mathbf{\Lambda}}}\widehat{\mathbf{M}}^{-1}\widehat{\mathbf{\Lambda}}^{-1}\right\|_2 \\ &\leq \left\|\widetilde{\mathbf{U}}\widetilde{\mathbf{\Lambda}}\mathbf{M}^{-1}\mathbf{\Lambda}^{-1}-\widehat{\widetilde{\mathbf{U}}}\widetilde{\mathbf{\Lambda}}\mathbf{M}^{-1}\mathbf{\Lambda}^{-1}+\widehat{\widetilde{\mathbf{U}}}\widetilde{\mathbf{\Lambda}}\mathbf{M}^{-1}\mathbf{\Lambda}^{-1}-\widehat{\widetilde{\mathbf{U}}}\widehat{\widetilde{\mathbf{\Lambda}}}\widehat{\mathbf{M}}^{-1}\widehat{\mathbf{\Lambda}}^{-1}\right\|_2 \\ &\leq \left\|\widehat{\widetilde{\mathbf{U}}}-\widetilde{\mathbf{U}}\right\|_2\left\|\widetilde{\mathbf{\Lambda}}\right\|_2\left\|\mathbf{M}^{-1}\right\|_2\left\|\mathbf{\Lambda}^{-1}\right\|_2+\left\|\widehat{\widetilde{\mathbf{U}}}\widetilde{\mathbf{\Lambda}}\mathbf{M}^{-1}\mathbf{\Lambda}^{-1}-\widehat{\widetilde{\mathbf{U}}}\widehat{\widetilde{\mathbf{\Lambda}}}\widehat{\mathbf{M}}^{-1}\widehat{\mathbf{\Lambda}}^{-1}\right\|_2 \\ &\leq \left\|\widehat{\widetilde{\mathbf{U}}}-\widetilde{\mathbf{U}}\right\|_2\left\|\widetilde{\mathbf{\Lambda}}\right\|_2\left\|\mathbf{M}^{-1}\right\|_2\left\|\mathbf{\Lambda}^{-1}\right\|_2+\left\|\widehat{\widetilde{\mathbf{U}}}\right\|_2\left\|\widetilde{\mathbf{\Lambda}}\mathbf{M}^{-1}\mathbf{\Lambda}^{-1}-\widehat{\widetilde{\mathbf{\Lambda}}}\widehat{\mathbf{M}}^{-1}\widehat{\mathbf{\Lambda}}^{-1}\right\|_2.\end{aligned} \tag{26}$$

For $\left\|\widetilde{\mathbf{\Lambda}}\mathbf{M}^{-1}\mathbf{\Lambda}^{-1}-\widehat{\widetilde{\mathbf{\Lambda}}}\widehat{\mathbf{M}}^{-1}\widehat{\mathbf{\Lambda}}^{-1}\right\|_2$, we have

$$\begin{aligned}&\left\|\widetilde{\mathbf{\Lambda}}\mathbf{M}^{-1}\mathbf{\Lambda}^{-1}-\widehat{\widetilde{\mathbf{\Lambda}}}\widehat{\mathbf{M}}^{-1}\widehat{\mathbf{\Lambda}}^{-1}\right\|_2 \\ &= \left\|\widehat{\widetilde{\mathbf{\Lambda}}}\widehat{\mathbf{M}}^{-1}\widehat{\mathbf{\Lambda}}^{-1}-\widehat{\widetilde{\mathbf{\Lambda}}}\mathbf{M}^{-1}\mathbf{\Lambda}^{-1}+\widehat{\widetilde{\mathbf{\Lambda}}}\mathbf{M}^{-1}\mathbf{\Lambda}^{-1}-\widetilde{\mathbf{\Lambda}}\mathbf{M}^{-1}\mathbf{\Lambda}^{-1}\right\|_2 \\ &\leq \left\|\widehat{\widetilde{\mathbf{\Lambda}}}\right\|_2\left\|\widehat{\mathbf{M}}^{-1}\widehat{\mathbf{\Lambda}}^{-1}-\mathbf{M}^{-1}\mathbf{\Lambda}^{-1}\right\|_2+\left\|\widehat{\widetilde{\mathbf{\Lambda}}}-\widetilde{\mathbf{\Lambda}}\right\|_2\left\|\mathbf{M}^{-1}\mathbf{\Lambda}^{-1}\right\|_2 \\ &\leq \left\|\widehat{\widetilde{\mathbf{\Lambda}}}\right\|_2\left\|\widehat{\mathbf{M}}^{-1}\widehat{\mathbf{\Lambda}}^{-1}-\mathbf{M}^{-1}\mathbf{\Lambda}^{-1}\right\|_2+\left\|\widehat{\widetilde{\mathbf{\Lambda}}}-\widetilde{\mathbf{\Lambda}}\right\|_2\left\|\mathbf{M}^{-1}\right\|_2\left\|\mathbf{\Lambda}^{-1}\right\|_2.\end{aligned} \tag{27}$$

To bound $\left\|\widehat{\mathbf{M}}^{-1}\widehat{\mathbf{\Lambda}}^{-1}-\mathbf{M}^{-1}\mathbf{\Lambda}^{-1}\right\|_2$, we have

$$\begin{aligned}&\left\|\widehat{\mathbf{M}}^{-1}\widehat{\mathbf{\Lambda}}^{-1}-\mathbf{M}^{-1}\mathbf{\Lambda}^{-1}\right\|_2 \\ &= \left\|\widehat{\mathbf{M}}^{-1}\widehat{\mathbf{\Lambda}}^{-1}-\widehat{\mathbf{M}}^{-1}\mathbf{\Lambda}^{-1}+\widehat{\mathbf{M}}^{-1}\mathbf{\Lambda}^{-1}-\mathbf{M}^{-1}\mathbf{\Lambda}^{-1}\right\|_2 \\ &\leq \left\|\widehat{\mathbf{M}}^{-1}\widehat{\mathbf{\Lambda}}^{-1}-\widehat{\mathbf{M}}^{-1}\mathbf{\Lambda}^{-1}\right\|_2+\left\|\widehat{\mathbf{M}}^{-1}\mathbf{\Lambda}^{-1}-\mathbf{M}^{-1}\mathbf{\Lambda}^{-1}\right\|_2 \\ &\leq \left\|\widehat{\mathbf{M}}^{-1}\right\|_2\left\|\widehat{\mathbf{\Lambda}}^{-1}-\mathbf{\Lambda}^{-1}\right\|_2+\left\|\widehat{\mathbf{M}}^{-1}-\mathbf{M}^{-1}\right\|_2\left\|\mathbf{\Lambda}^{-1}\right\|_2.\end{aligned} \tag{28}$$

By combining Eqs. (26)-(28), we can derive

$$\begin{aligned}\left\|\mathbf{U}-\widehat{\mathbf{U}}\right\|_2 &= \left\|\widetilde{\mathbf{U}}\widetilde{\mathbf{\Lambda}}\mathbf{M}^{-1}\mathbf{\Lambda}^{-1}-\widehat{\widetilde{\mathbf{U}}}\widehat{\widetilde{\mathbf{\Lambda}}}\widehat{\mathbf{M}}^{-1}\widehat{\mathbf{\Lambda}}^{-1}\right\|_2 \\ &\leq \left\|\widehat{\widetilde{\mathbf{U}}}-\widetilde{\mathbf{U}}\right\|_2\left\|\widetilde{\mathbf{\Lambda}}\right\|_2\left\|\mathbf{M}^{-1}\right\|_2\left\|\mathbf{\Lambda}^{-1}\right\|_2+\left\|\widehat{\widetilde{\mathbf{U}}}\right\|_2\left\|\widehat{\widetilde{\mathbf{\Lambda}}}\right\|_2\left\|\widehat{\mathbf{M}}^{-1}\right\|_2\left\|\widehat{\mathbf{\Lambda}}^{-1}-\mathbf{\Lambda}^{-1}\right\|_2 \\ &\quad+\left\|\widehat{\widetilde{\mathbf{U}}}\right\|_2\left\|\widehat{\widetilde{\mathbf{\Lambda}}}\right\|_2\left\|\widehat{\mathbf{M}}^{-1}-\mathbf{M}^{-1}\right\|_2\left\|\mathbf{\Lambda}^{-1}\right\|_2+\left\|\widehat{\widetilde{\mathbf{U}}}\right\|_2\left\|\widehat{\widetilde{\mathbf{\Lambda}}}-\widetilde{\mathbf{\Lambda}}\right\|_2\left\|\mathbf{M}^{-1}\right\|_2\left\|\mathbf{\Lambda}^{-1}\right\|_2.\end{aligned} \tag{29}$$

By recalling the upper bounds of matrix norms and the upper bounds of estimation errors for different terms, we obtain that the following inequalities hold with probability at least $1-\delta$, respectively, which are

1). $\left\|\widehat{\widetilde{\mathbf{U}}}-\widetilde{\mathbf{U}}\right\|_2\left\|\widetilde{\mathbf{\Lambda}}\right\|_2\left\|\mathbf{M}^{-1}\right\|_2\left\|\mathbf{\Lambda}^{-1}\right\|_2 \leq \overline{X}\sqrt{dC}\sqrt{\frac{2}{\min_i \widetilde{n}_i}\log\frac{2dC}{\delta}}\cdot\xi_M\mathrm{cond}_2(\mathbf{M})\cdot\frac{\max_i\widetilde{\pi}_i}{\min_i\pi_i}$;

2). $\left\|\widehat{\widetilde{\mathbf{U}}}\right\|_2\left\|\widehat{\widetilde{\mathbf{\Lambda}}}\right\|_2\left\|\widehat{\mathbf{M}}^{-1}\right\|_2\left\|\widehat{\mathbf{\Lambda}}^{-1}-\mathbf{\Lambda}^{-1}\right\|_2 \leq \sqrt{CX}\cdot\xi_M\mathrm{cond}_2(\widehat{\mathbf{M}})\cdot\frac{\left\|\mathbf{T}^{-1}\right\|_1}{\min_i\widehat{\pi}_i\min_i\pi_i}\sqrt{\frac{1}{2\widetilde{n}}\log\frac{2C}{\delta}}$;

3). $\left\|\widehat{\widetilde{\mathbf{U}}}\right\|_2\left\|\widehat{\widetilde{\mathbf{\Lambda}}}\right\|_2\left\|\widehat{\mathbf{M}}^{-1}-\mathbf{M}^{-1}\right\|_2\left\|\mathbf{\Lambda}^{-1}\right\|_2 \leq C\sqrt{CX}\frac{\max_i\widehat{\widetilde{\pi}}_i}{\min_i\pi_i}\cdot 2\left\|\mathbf{T}^{-1}\right\|_1\sqrt{\frac{1}{2\widetilde{n}}\log\frac{2C}{\delta}}$;

4). $\left\|\widehat{\widetilde{\mathbf{U}}}\right\|_2\left\|\widehat{\widetilde{\mathbf{\Lambda}}}-\widetilde{\mathbf{\Lambda}}\right\|_2\left\|\mathbf{M}^{-1}\right\|_2\left\|\mathbf{\Lambda}^{-1}\right\|_2 \leq \sqrt{CX}\cdot\sqrt{\frac{1}{2\widetilde{n}}\log\frac{2C}{\delta}}\cdot\xi_M\mathrm{cond}_2(\mathbf{M})\cdot\frac{1}{\min_i\pi_i}$.

Now we define the following constants:

$$\begin{cases}\zeta = \frac{\overline{X}\sqrt{C}\xi_M\mathrm{cond}_2(\widehat{\mathbf{M}})}{\min_i\widehat{\pi}_i\min_i\pi_i}+\frac{2\overline{X}C\sqrt{C}\max_i\widehat{\widetilde{\pi}}_i}{\min_i\pi_i} \\ \beta = \frac{\overline{X}\sqrt{C}\xi_M\mathrm{cond}_2(\mathbf{M})}{\min_i\pi_i} \\ \gamma = \frac{\overline{X}\cdot\xi_M\mathrm{cond}_2(\mathbf{M})\max_i\widetilde{\pi}_i}{\min_i\pi_i}\end{cases}. \tag{30}$$

Then by combining Eqs. (29) and the upper bounds for each terms, with probability at least $1-\delta$, we have

$$\left\|\mathbf{U}-\widehat{\mathbf{U}}\right\|_2 \leq \gamma\sqrt{\frac{2dC}{\min_k\widetilde{n}_k}\log\frac{8dC}{\delta}}+(\zeta\|\mathbf{T}^{-1}\|_1+\beta)\cdot\sqrt{\frac{1}{2\widetilde{n}}\log\frac{8C}{\delta}}. \tag{31}$$

Therefore, for $s=1$ (the first-order statistics) or $s=2$ (the second-order statistics), if $\widetilde{n} > 2C^2\|\mathbf{T}^{-1}\|_1^2\log\frac{8C}{\delta}$, with probability at least $1-\delta$, we have

$$\left\|\mathbf{U}^{(s)}-\widehat{\mathbf{U}}^{(s)}\right\|_2 \leq \gamma^{(s)}\sqrt{\frac{2d^sC}{\min_k\widetilde{n}_k}\log\frac{8d^sC}{\delta}}+(\zeta^{(s)}\|\mathbf{T}^{-1}\|_1+\beta^{(s)})\cdot\sqrt{\frac{1}{2\widetilde{n}}\log\frac{8C}{\delta}}. \tag{32}$$

Therefore, we complete the proof. □

5 Proof of Theorem 2

Proof. Now we consider the estimation error on the transition matrix $\mathbf{T}$. The estimated transition matrix is denoted by $\widehat{\mathbf{T}}$. Let $\Delta\mathbf{T} = \mathbf{T} - \widehat{\mathbf{T}}$, then we can get $\mathbf{T} = \widehat{\mathbf{T}} + \Delta\mathbf{T}$. By using the Woodbury identity and the triangle inequality, we have

$$\begin{aligned}\|\mathbf{T}^{-1}\|_1 &= \|(\widehat{\mathbf{T}} + \Delta\mathbf{T})^{-1}\|_1 \\ &= \|\widehat{\mathbf{T}}^{-1} - \widehat{\mathbf{T}}^{-1}(\mathbf{I} + \Delta\mathbf{T}\widehat{\mathbf{T}}^{-1})^{-1}\Delta\mathbf{T}\widehat{\mathbf{T}}^{-1}\|_1 \\ &\leq \|\widehat{\mathbf{T}}^{-1}\|_1 + \|\widehat{\mathbf{T}}^{-1}\|_1^2 \cdot \|(\mathbf{I} + \Delta\mathbf{T}\widehat{\mathbf{T}}^{-1})^{-1}\|_1 \cdot \|\Delta\mathbf{T}\|_1.\end{aligned} \tag{33}$$

Then we substitute this inequality into Eq. (32). Since we assume that $\mathbf{I} + (\mathbf{T} - \widehat{\mathbf{T}})\widehat{\mathbf{T}}^{-1}$ is invertible, and the norm of its inverse matrix is upper bounded, this theorem can be proved readily, namely for any $\delta > 0$ and $s \in \{1, 2\}$, with probability at least $1 - \delta$, we have

$$\left\|\mathbf{U}^{(s)} - \widehat{\mathbf{U}}^{(s)}\right\|_2 \leq \gamma^{(s)}\sqrt{\frac{2d^sC}{\min_k \widetilde{n}_k}\log\frac{8d^sC}{\delta}} + (\zeta^{(s)}\|\widehat{\mathbf{T}}^{-1}\|_1 + \beta^{(s)}) \cdot \sqrt{\frac{1}{2\widetilde{n}}\log\frac{8C}{\delta}} + \mathcal{O}\left(\frac{\|\mathbf{T} - \widehat{\mathbf{T}}\|_1}{\sqrt{\widetilde{n}}}\right). \tag{34}$$

Therefore, the proof is completed. □

6 Proof of Theorem 3

Proof. First of all, we consider the symmetric label noise. We recall the estimators of NESC provided in [6], namely

$$\begin{cases}[\boldsymbol{\mu}_P, \boldsymbol{\mu}_N] = [\widetilde{\boldsymbol{\mu}}_P, \widetilde{\boldsymbol{\mu}}_N]\mathbf{S} \\ [\boldsymbol{\sigma}_P, \boldsymbol{\sigma}_N] = [\widetilde{\boldsymbol{\sigma}}_P, \widetilde{\boldsymbol{\sigma}}_N]\mathbf{S}\end{cases}, \tag{35}$$

where $\mathbf{S} = \begin{bmatrix}\frac{(1-\widetilde{\pi})\cdot(1-\eta)}{1-\widetilde{\pi}-\eta} & \frac{-\eta(1-\widetilde{\pi})}{\widetilde{\pi}-\eta} \\ \frac{-\widetilde{\pi}\cdot\eta}{1-\widetilde{\pi}-\eta} & \frac{\widetilde{\pi}\cdot(1-\eta)}{\widetilde{\pi}-\eta}\end{bmatrix}$ is the coefficient matrix. Now we need to derive the coefficient matrix of our PCSE. By simple calculation, we have

$$\mathbf{M} = \sum_{i=1}^{C}\pi_i\sum_{j=1}^{C}T_{ij}\mathbf{K}_{i\to j} = \begin{bmatrix}1-\eta & \eta \\ \eta & 1-\eta\end{bmatrix}. \tag{36}$$

Since $\widetilde{\pi} = (1-2\eta)\pi + \eta$, we have

$$\begin{aligned}\widetilde{\boldsymbol{\Lambda}}\mathbf{M}^{-1}\boldsymbol{\Lambda}^{-1} &= \begin{bmatrix}1-\widetilde{\pi} & 0 \\ 0 & \widetilde{\pi}\end{bmatrix}\begin{bmatrix}1-\eta & \eta \\ \eta & 1-\eta\end{bmatrix}^{-1}\begin{bmatrix}1-\pi & 0 \\ 0 & \pi\end{bmatrix}^{-1} \\ &= \frac{1}{1-2\eta}\begin{bmatrix}1-\widetilde{\pi} & 0 \\ 0 & \widetilde{\pi}\end{bmatrix}\begin{bmatrix}1-\eta & -\eta \\ -\eta & 1-\eta\end{bmatrix}\begin{bmatrix}1-\pi & 0 \\ 0 & \pi\end{bmatrix}^{-1} \\ &= \begin{bmatrix}\frac{1-\widetilde{\pi}}{1-\pi}\cdot\frac{1-\eta}{1-2\eta} & \frac{(1-\widetilde{\pi})(-\eta)}{\pi(1-2\eta)} \\ \frac{\widetilde{\pi}(-\eta)}{(1-\pi)(1-2\eta)} & \frac{\widetilde{\pi}(1-\eta)}{\pi(1-2\eta)}\end{bmatrix}.\end{aligned} \tag{37}$$

Since $\pi = \frac{\widetilde{\pi}-\eta}{1-2\eta}$ and $1-\pi = \frac{1-\eta-\widetilde{\pi}}{1-2\eta}$, we have

$$\widetilde{\boldsymbol{\Lambda}}\mathbf{M}^{-1}\boldsymbol{\Lambda}^{-1} = \begin{bmatrix}\frac{(1-\widetilde{\pi})\cdot(1-\eta)}{1-\widetilde{\pi}-\eta} & \frac{-\eta(1-\widetilde{\pi})}{\widetilde{\pi}-\eta} \\ \frac{-\widetilde{\pi}\cdot\eta}{1-\widetilde{\pi}-\eta} & \frac{\widetilde{\pi}\cdot(1-\eta)}{\widetilde{\pi}-\eta}\end{bmatrix} = \mathbf{S}. \tag{38}$$

Therefore, our estimation is given by

$$\mathbf{U} = [\boldsymbol{\mu}_P, \boldsymbol{\mu}_N] = \widetilde{\mathbf{U}}\widetilde{\boldsymbol{\Lambda}}\mathbf{M}^{-1}\boldsymbol{\Lambda}^{-1} = [\widetilde{\boldsymbol{\mu}}_P, \widetilde{\boldsymbol{\mu}}_N]\mathbf{S}. \tag{39}$$

which is the same as the estimation result of NESC. For the second-order statistics, we can derive the same formulation as NESC, which is given by

$$[\boldsymbol{\sigma}_P, \boldsymbol{\sigma}_N] = [\widetilde{\boldsymbol{\sigma}}_P, \widetilde{\boldsymbol{\sigma}}_N]\mathbf{S}. \tag{40}$$

Therefore, PCSE and NESC have the same estimators of per-class sample mean and covariance under symmetric label noise.

For asymmetric label noise, our estimation has the form $[\boldsymbol{\mu}_P, \boldsymbol{\mu}_N] = [\widetilde{\boldsymbol{\mu}}_P, \widetilde{\boldsymbol{\mu}}_N]\mathbf{S}'$, where $\mathbf{S}' = [S'_{ij}] \in \mathbb{R}^{2\times 2}$ is the coefficient matrix. Its elements are

$$\begin{cases}S'_{00} = \frac{(1-\widetilde{\pi})(1-\eta_P-\eta_N)[1-\eta_N-\pi(\eta_P-\eta_N)]}{[1-2\eta_N-2\pi(\eta_P-\eta_N)](1-\widetilde{\pi}-\eta_P)} \\ S'_{01} = \frac{(1-\widetilde{\pi})(1-\eta_P-\eta_N)[-\eta_N-\pi(\eta_P-\eta_N)]}{[1-2\eta_N-2\pi(\eta_P-\eta_N)](\widetilde{\pi}-\eta_N)} \\ S'_{10} = \frac{\widetilde{\pi}(1-\eta_P-\eta_N)[-\eta_N-\pi(\eta_P-\eta_N)]}{[1-2\eta_N-2\pi(\eta_P-\eta_N)](1-\widetilde{\pi}-\eta_N)} \\ S'_{11} = \frac{\widetilde{\pi}(1-\eta_P-\eta_N)[1-\eta_N-\pi(\eta_P-\eta_N)]}{[1-2\eta_N-2\pi(\eta_P-\eta_N)](\widetilde{\pi}-\eta_N)}\end{cases}. \tag{41}$$

TABLE 2
Properties of seven additional UCI Benchmark datasets.

Datasets	n	d	n_+	n_-
GammaTele	19020	10	12332	6688
Banana	5300	2	2376	2924
Ringnorm	7400	20	3664	3736
Splice	2991	60	1344	1647
Thyroid	215	5	65	150
Twonorm	7400	20	3703	3697
Waveform	5000	21	1647	3353

Here π and $\widetilde{\pi}$ are two scalars, and they have the relationship $\pi = (1 - \eta_P - \eta_N)\widetilde{\pi} + \eta_N$.

By contrast, the estimation of NESC has the form $[\boldsymbol{\mu}_P, \boldsymbol{\mu}_N] = [\widetilde{\boldsymbol{\mu}}_P, \widetilde{\boldsymbol{\mu}}_N]\mathbf{S}''$, where $\mathbf{S}'' = [S''_{ij}] \in \mathbb{R}^{2\times 2}$ is the coefficient matrix. This matrix can be expressed analytically as

$$S'' = \begin{bmatrix} \frac{(1-\widetilde{\pi})(1-\eta_P)}{1-\widetilde{\pi}-\eta_P} & \frac{(1-\widetilde{\pi})(-\eta_N)}{\widetilde{\pi}-\eta_N} \\ \frac{-\widetilde{\pi}\eta_P}{1-\widetilde{\pi}-\eta_P} & \frac{\widetilde{\pi}(1-\eta_N)}{\widetilde{\pi}-\eta_N} \end{bmatrix}. \tag{42}$$

We find that $\mathbf{S}' \neq \mathbf{S}''$, so for asymmetric label noise, the estimators of NESC and our PCSE for the first-order statistics are different. For the second-order statistics, we can draw such conclusion in the same manner. □

7 Additional Experiments

7.1 Experimental Settings under Binary Classification

In this section, we elaborate on the experimental settings for various baseline methods under binary classification. Specifically, the compared methods are CrossEntropy introduced in Section 6.1.2, GCE [23], $\mathcal{L}_{\text{DMI}}$ [21], Co-teaching [9], CEGE [8], CWD [7], ULE [17], RoG [14], MC-LDCE [4], ASL [24], ROBOT [22], NESC [6], and our PCSE. For some methods that require label flip rates, such as CEGE, CWD, ULE, MC-LDCE, ROBOT, NESC, and our PCSE, we directly provide them with the real values of η_P and η_N. For all methods, we adopt the same three-layer MLP as the backbone network, where the number of nodes in the input layer is the same as the feature dimensionality d, and the number of nodes in the output layer is set to 2. The number of nodes in the hidden layer is decided as $\lceil 2/3 \times (d+2) \rceil$ ($\lceil \cdot \rceil$ is the ceiling function) as recommended in [10] and [7] to achieve satisfactory performance. For $\mathcal{L}_{\text{DMI}}$, according to the official implementation[1], we pre-train the MLP with the vanilla Cross-Entropy loss and fine-tune the network with the DMI loss. For ULE, the logistic loss is leveraged to construct an unbiased loss function. For GCE, the hyperparameter q for the negative Box-Cox transformation is set to 0.7, as recommended by [23]. Additionally, for CEGE and CWD, the commonly used squared loss function is adopted to build the unbiased loss function. For ASL, the NCE+AGCE loss is used, and the hyperparameters a, q, α, and β is set to 1, 0.9, 1, and 1, respectively. For ROBOT, the forward loss is adopted, and the RCE loss is used in the outer loop, as recommended in the original paper. For NESC, since this method is originally proposed to estimate label flip rates, here we combine its estimation results with the generative classifier deployed in RoG for model inference. For RoG, NESC, and our PCSE, the adopted pre-training method is CrossEntropy, since it is easy to implement and our PCSE can already obtain satisfactory performance with this non-robust pre-training method. The hidden-layer features extracted from the pre-trained network are used to calculate the sample means and covariance matrices. After that, the sample means and covariance matrices are further utilized in the label inference process. The Adam optimizer [12] with default parameters is employed for network training in all experiments. We select the learning rate via searching the grid $\{0.1, 0.01, 0.001\}$.

7.2 Additional experiments on UCI benchmark datasets

In our main paper, five UCI datasets have been adopted for the evaluation of our method under binary label noise. To confirm the effectiveness of the proposed method, more datasets in UCI benchmark repository are included here for further empirical evaluation. The seven adopted additional benchmark datasets regarding binary classification include *GammaTele*, *Banana*, *Ringnorm*, *Splice*, *Thyroid*, *Twonorm*, and *Waveform*[2]. A brief introduction of the datasets is presented in Table 2, which contains some essential configurations such as the number of examples n, the feature dimensionality d, the number of positive examples n_+, and the number of negative examples n_-. The features for each dataset have been normalized and standardized.

The experiments on seven adopted UCI datasets are provided in Table 3. As shown in this table, our method surpasses other compared approaches in most cases. For the average accuracy over all the datasets under different label flip rates, our PCSE achieves a record of 82.0%, which leads the second and third best methods by a margin of 0.4% and 1.8%, respectively. To summarize, the results in Table 3 clearly verify the robustness and discriminativeness of our PCSE over other baseline methods in dealing with label noise.

TABLE 3
Comparison of various approaches on seven UCI benchmark datasets. The best two records on each dataset are highlighted in red and blue, respectively. The "√"("×") denotes that our PCSE is significantly better (worse) than the corresponding compared method revealed by the paired t-test with significance level 0.1.

Dataset	(η_P, η_N)	CrossEntropy	GCE [23]	$\mathcal{L}_{\text{DMI}}$ [21]	Co-teaching [9]	CEGE [8]	CWD [7]	ULE [17]	RoG [14]	MC-LDCE [4]	ASL [24]	ROBOT [22]	NESC [6]	PCSE
GammaTele	(0.0, 0.0)	84.9 ± 0.2	67.4 ± 1.9√	84.9 ± 0.2	80.2 ± 1.0√	83.6 ± 0.2√	83.4 ± 0.3√	85.2 ± 0.1	85.0 ± 0.1	83.6 ± 0.6√	81.9 ± 0.7√	84.8 ± 0.8	85.0 ± 0.3	85.0 ± 0.3
	(0.2, 0.2)	81.5 ± 0.5√	75.0 ± 2.2√	82.6 ± 0.3	63.2 ± 0.9√	82.6 ± 0.7	81.4 ± 0.2√	82.2 ± 0.7√	83.0 ± 0.5√	81.9 ± 1.1	74.4 ± 3.1√	81.6 ± 0.3√	83.3 ± 0.4	83.3 ± 0.4
	(0.3, 0.1)	79.0 ± 1.5√	79.1 ± 2.3	81.0 ± 1.6	69.2 ± 1.0√	78.4 ± 1.7√	77.7 ± 0.8√	81.0 ± 0.6√	82.6 ± 0.4	78.2 ± 0.7√	74.2 ± 2.1√	79.7 ± 0.4√	81.9 ± 0.2√	82.9 ± 0.2
	(0.4, 0.4)	62.7 ± 3.4√	71.3 ± 3.3	64.5 ± 3.4	51.6 ± 0.9√	72.4 ± 1.9	68.5 ± 3.9	65.3 ± 6.2	63.1 ± 4.0√	74.8 ± 1.9	67.5 ± 3.2	70.0 ± 2.8	72.9 ± 1.8	72.9 ± 1.8
Banana	(0.0, 0.0)	71.9 ± 1.4	64.7 ± 3.6√	71.4 ± 1.4√	72.4 ± 2.9	71.7 ± 2.0	71.2 ± 2.8	73.0 ± 2.9	61.3 ± 1.5	70.8 ± 1.0√	67.3 ± 3.0	64.3 ± 3.7	73.8 ± 2.4	73.8 ± 2.4
	(0.2, 0.2)	64.7 ± 1.7√	67.7 ± 5.9	65.2 ± 2.1	55.7 ± 1.5√	67.6 ± 4.3	65.8 ± 2.3	65.5 ± 2.0	56.8 ± 3.6	62.4 ± 2.1√	64.0 ± 1.7	64.9 ± 1.3	65.8 ± 1.7	65.8 ± 1.7
	(0.3, 0.1)	64.4 ± 1.8√	64.1 ± 0.5√	63.9 ± 1.6	65.1 ± 2.6	59.1 ± 6.2	63.7 ± 4.6	63.8 ± 3.4	60.7 ± 1.2√	64.2 ± 1.8√	55.8 ± 0.9√	60.1 ± 3.3√	58.5 ± 2.2√	65.9 ± 1.2
	(0.4, 0.4)	50.5 ± 3.3	54.7 ± 2.0	54.3 ± 2.1	50.0 ± 1.4√	55.0 ± 0.5	52.1 ± 3.1	54.8 ± 1.3	52.5 ± 1.8√	55.0 ± 0.5	54.5 ± 1.2	55.5 ± 1.7	56.5 ± 0.8	56.5 ± 0.8
Ringnorm	(0.0, 0.0)	93.4 ± 0.3	92.6 ± 0.2√	92.9 ± 0.7	92.8 ± 0.4	63.0 ± 1.2√	91.4 ± 1.5	92.5 ± 0.8	89.3 ± 0.3√	91.2 ± 0.9√	88.6 ± 0.8√	92.3 ± 0.4√	93.5 ± 0.1	93.5 ± 0.1
	(0.2, 0.2)	85.6 ± 1.9	64.0 ± 3.9√	84.1 ± 1.3√	74.9 ± 0.3√	62.7 ± 2.2√	85.1 ± 2.8	85.3 ± 3.2	83.6 ± 1.8	87.4 ± 2.6	81.9 ± 2.1√	84.7 ± 2.6√	87.7 ± 1.7	87.7 ± 1.7
	(0.3, 0.1)	80.8 ± 0.8√	60.6 ± 14.3	80.3 ± 1.8√	74.9 ± 0.6√	58.5 ± 2.1√	82.0 ± 1.2	79.4 ± 0.4√	84.8 ± 0.6	82.2 ± 0.6√	73.2 ± 0.6√	81.2 ± 2.5	84.8 ± 0.4	84.7 ± 0.5
	(0.4, 0.4)	57.7 ± 5.8√	73.6 ± 4.1	64.0 ± 3.8√	54.7 ± 0.9√	63.5 ± 1.6√	71.9 ± 2.1	65.2 ± 2.1√	69.7 ± 4.0	72.8 ± 3.6√	64.1 ± 7.2√	73.9 ± 1.2	73.9 ± 3.4	73.9 ± 3.4
Splice	(0.0, 0.0)	89.1 ± 0.6	62.1 ± 9.9√	88.8 ± 0.3	88.5 ± 0.2	87.2 ± 1.4	75.6 ± 3.8√	88.9 ± 0.4	89.3 ± 0.4	70.2 ± 4.7√	73.1 ± 2.0√	88.1 ± 0.9√	89.3 ± 0.5	89.3 ± 0.5
	(0.2, 0.2)	79.5 ± 0.1√	58.0 ± 4.1√	79.9 ± 0.9√	70.3 ± 1.1√	80.1 ± 1.2	62.9 ± 4.2√	79.8 ± 1.3√	81.8 ± 0.2	70.3 ± 4.8√	67.8 ± 1.1√	82.0 ± 0.4	82.6 ± 0.4	82.6 ± 0.4
	(0.3, 0.1)	75.0 ± 3.3√	57.2 ± 2.9√	77.5 ± 1.1√	71.2 ± 1.4√	76.5 ± 1.7√	60.5 ± 0.6√	77.0 ± 2.5√	80.1 ± 2.1	62.0 ± 2.4√	57.7 ± 0.5√	75.3 ± 3.4√	80.1 ± 2.5	80.5 ± 1.9
	(0.4, 0.4)	56.7 ± 2.3	54.1 ± 0.8	53.9 ± 2.3	54.0 ± 1.0	59.3 ± 4.6	54.0 ± 0.9	49.3 ± 2.8√	57.2 ± 2.9	58.6 ± 4.0	52.8 ± 3.6√	55.7 ± 3.4	57.3 ± 1.8	57.3 ± 1.8
Thyroid	(0.0, 0.0)	92.3 ± 0.8	89.6 ± 2.6	91.6 ± 1.3	90.1 ± 1.1√	89.0 ± 0.6√	83.4 ± 0.9√	92.2 ± 0.7	91.6 ± 2.7	86.2 ± 3.8	88.1 ± 1.6√	90.3 ± 2.7	93.1 ± 0.6	93.1 ± 0.6
	(0.2, 0.2)	91.2 ± 1.3√	86.5 ± 1.4√	88.1 ± 1.2	72.2 ± 2.2√	87.0 ± 1.7√	84.2 ± 2.6√	87.3 ± 1.9	79.5 ± 6.6√	83.5 ± 1.7√	89.0 ± 0.9	88.9 ± 2.4	91.6 ± 1.4	91.6 ± 1.4
	(0.3, 0.1)	83.0 ± 0.8√	81.9 ± 2.1√	80.0 ± 3.1√	75.2 ± 2.1√	81.9 ± 2.1√	78.0 ± 3.8√	83.3 ± 2.7	83.3 ± 1.4√	78.1 ± 3.0√	79.7 ± 3.1√	85.7 ± 1.8	87.4 ± 1.7	87.8 ± 0.6
	(0.4, 0.4)	73.9 ± 4.6	73.2 ± 4.7	73.5 ± 2.9	52.1 ± 2.3√	74.4 ± 4.4	71.1 ± 0.3	70.7 ± 6.4	67.7 ± 1.5	74.8 ± 6.5	76.2 ± 6.0	70.4 ± 7.4	69.3 ± 5.9	69.3 ± 5.9
Twonorm	(0.0, 0.0)	97.5 ± 0.1×	93.7 ± 3.4	97.5 ± 0.1×	97.5 ± 0.0×	96.0 ± 1.9	97.7 ± 0.0×	97.5 ± 0.1×	86.5 ± 4.5	97.8 ± 0.1×	96.7 ± 0.4×	96.9 ± 0.9×	90.8 ± 3.0	90.8 ± 3.0
	(0.2, 0.2)	92.9 ± 0.4	85.0 ± 6.7	88.8 ± 5.3	78.3 ± 0.2√	91.0 ± 4.0	94.7 ± 0.2	93.5 ± 0.9	89.7 ± 0.1	93.4 ± 0.0	94.1 ± 1.8	94.6 ± 1.7	92.1 ± 4.4	92.1 ± 4.4
	(0.3, 0.1)	90.1 ± 3.3	77.0 ± 13.5	89.6 ± 5.2	78.2 ± 0.3√	88.6 ± 5.3	91.2 ± 3.5	90.9 ± 3.4	89.2 ± 0.3	92.0 ± 3.1	89.5 ± 3.6	88.8 ± 1.9	89.5 ± 0.4	92.3 ± 1.8
	(0.4, 0.4)	83.8 ± 5.7	84.9 ± 2.9	86.3 ± 4.3	57.3 ± 1.3√	82.0 ± 0.4√	85.8 ± 4.2	83.5 ± 2.8	85.4 ± 2.7	86.2 ± 4.3	84.6 ± 2.4	82.7 ± 3.0	87.3 ± 0.3	87.3 ± 0.3
Waveform	(0.0, 0.0)	90.0 ± 0.6	75.0 ± 4.6√	90.5 ± 0.2√	90.4 ± 0.2	90.6 ± 0.3	90.2 ± 0.6	89.0 ± 1.9	90.7 ± 0.2√	90.1 ± 0.1√	89.4 ± 0.5√	90.8 ± 0.2√	91.0 ± 0.1	91.0 ± 0.1
	(0.2, 0.2)	87.9 ± 0.3√	67.1 ± 0.0√	88.9 ± 0.4√	73.9 ± 0.4√	89.2 ± 0.7	89.7 ± 0.4	89.5 ± 0.2√	89.9 ± 0.6√	88.6 ± 0.4√	86.5 ± 0.5√	90.3 ± 0.5	90.5 ± 0.4	90.5 ± 0.4
	(0.3, 0.1)	87.2 ± 1.4√	68.6 ± 2.1√	87.3 ± 1.2√	77.3 ± 0.4√	87.5 ± 0.8√	86.5 ± 1.2√	87.0 ± 0.2√	89.6 ± 1.1	85.0 ± 0.7√	83.9 ± 1.7√	89.0 ± 0.7√	89.9 ± 0.1	90.2 ± 0.2
	(0.4, 0.4)	79.2 ± 0.6√	81.7 ± 0.7√	80.0 ± 1.8√	56.0 ± 0.7√	80.4 ± 1.3√	79.8 ± 4.9	75.8 ± 1.3√	83.2 ± 1.7√	82.2 ± 0.9	69.5 ± 2.8√	82.1 ± 1.4√	85.1 ± 0.9	85.1 ± 0.9
Average		**79.5**	**72.5**	**79.7**	**71.0**	**77.1**	**77.8**	**79.6**	**78.8**	**78.7**	**75.9**	**80.2**	**81.6**	**82.0**

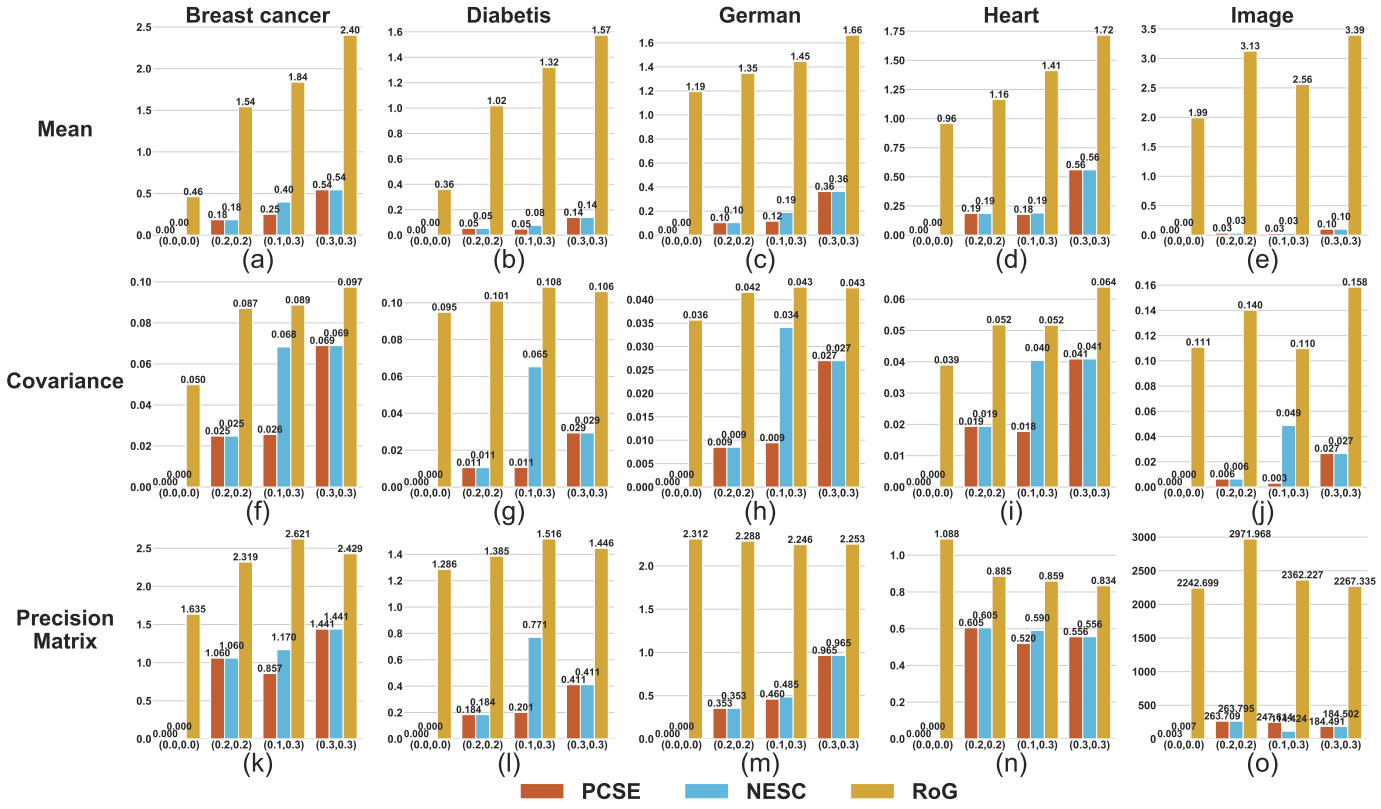


Fig. 2. Comparison of NESC, RoG and our PCSE on per-class mean, covariance and precision matrix estimation. The x-axis in each figure shows various pairs of (η_P, η_N). The first, second and third rows show the average estimation errors of mean, covariance, and precision matrix, respectively. Each column displays the results on a specific dataset. This figure clearly illustrates that our PCSE is able to achieve more precise estimations of per-class statistics than NESC and RoG on binary classification datasets.

7.3 Estimation Error Analysis under Binary Classification

To further evaluate the statistic estimation performance of our PCSE on the adopted UCI benchmark datasets, we compare the estimation results of our method on each binary dataset with those generated by two existing methods, including NESC [6] and RoG [14], as they are representative LNL methods relying on statistic estimation. Specifically, we evaluate the estimation errors of sample mean, covariance, and precision matrix on four pairs of label flip rates: $(\eta_P, \eta_N) = (0.0, 0.0)$, $(\eta_P, \eta_N) = (0.2, 0.2)$, $(\eta_P, \eta_N) = (0.3, 0.3)$, and $(\eta_P, \eta_N) = (0.1, 0.3)$, where the first pair corresponds to the noise-free case, while the second and third pairs stand for the symmetric label noise. The last one corresponds to the asymmetric label noise. The adopted evaluation metrics for per-class statistics are the same as Eq (27) in the original paper. Here the original features (instead of the features output by a DNN) of each dataset are used to calculate the statistics under the evaluation metrics. The average estimation errors over ten independent trials are recorded for evaluation. Notably, for NESC and PCSE, the actual noise rates η_P and η_N are used in their unbiased estimators.

The estimation errors of various methods on five UCI benchmark datasets are shown in Fig. 2, where the first, second, and third rows correspond to the estimation results of mean, covariance, and precision matrix, respectively. In this figure, we identify that in the cases of symmetric label noise, our PCSE indeed obtains the same results as NESC on the estimation of both per-class mean and covariance, which confirms the theoretical findings in Theorem 3. Moreover, under asymmetric label noise, our PCSE performs slightly better than NESC in terms of per-class mean estimation and consistently outperforms NESC in covariance estimation across all datasets. Additionally, on the estimation of precision matrix, our PCSE obtains smaller estimation errors than NESC and RoG. These results justify that our proposed estimators can obtain more precise estimations than NESC. It is worth noting that both PCSE and NESC outperform RoG on each dataset, which suggests that RoG often produces biased estimations on mean and covariance due to the sample selection process. Since our PCSE does not involve sample selection and it further takes into account the pairwise label flip rates, it can successfully obtain more precise estimations than RoG and NESC.

REFERENCES

[1] A. Blum and T. Mitchell, "Combining labeled and unlabeled data with co-training," in *Annual Conference on Computational Learning Theory*, 1998, p. 92–100.

[2] M. Bucarelli, L. Cassano, F. Siciliano, A. Mantrach, and F. Silvestri, "Leveraging inter-rater agreement for classification in the presence of noisy labels," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2023, pp. 3439–3448.

[3] S. E. Decatur, "PAC learning with constant-partition classification noise and applications to decision tree induction," in *International Workshop on Artificial Intelligence and Statistics*, 1997, pp. 147–156.

[4] Y. Ding, T. Zhou, C. Zhang, Y. Luo, J. Tang, and C. Gong, "Multi-class label noise learning via loss decomposition and centroid estimation," in *SIAM International Conference on Data Mining*, 2022, pp. 253–261.

[5] X. Fu, K. Huang, B. Yang, W.-K. Ma, and N. D. Sidiropoulos, "Robust volume minimization-based matrix factorization for remote sensing and document clustering," *IEEE Transactions on Signal Processing*, vol. 64, no. 23, pp. 6254–6268, 2016.

[6] W. Gao, T. Zhang, B.-B. Yang, and Z.-H. Zhou, "On the noise estimation statistics," *Artificial Intelligence*, vol. 293, p. 103451, 2021.

[7] C. Gong, Y. Ding, B. Han, G. Niu, J. Yang, J. You, D. Tao, and M. Sugiyama, "Class-wise denoising for robust learning under label noise," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 3, pp. 2835–2848, 2023.

[8] C. Gong, J. Yang, 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*, vol. 44, no. 6, pp. 2841–2855, 2020.

[9] 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," *Advances in Neural Information Processing Systems*, vol. 31, pp. 8527–8537, 2018.

[10] J. Heaton, *Introduction to Neural Networks for Java*, 2nd ed. Heaton Research, Inc., 2008.

[11] R. A. Horn and C. R. Johnson, *Matrix Analysis*, 2nd ed. Cambridge University Press, 2013.

[12] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *International Conference on Learning Representations*, 2014.

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

[14] 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*. PMLR, 2019, pp. 3763–3772.

[15] Y. Liu, H. Cheng, and K. Zhang, "Identifiability of label noise transition matrix," in *International Conference on Machine Learning*, 2023, pp. 21 475–21 496.

[16] M. Mohri, A. Rostamizadeh, and A. Talwalkar, *Foundations of Machine Learning*. MIT press, 2018.

[17] N. Natarajan, I. S. Dhillon, P. K. Ravikumar, and A. Tewari, "Learning with noisy labels," *Advances in Neural Information Processing Systems*, vol. 26, pp. 1196–1204, 2013.

[18] C. Scott, G. Blanchard, and G. Handy, "Classification with asymmetric label noise: Consistency and maximal denoising," in *Annual Conference on Learning Theory*, 2013, pp. 489–511.

[19] B. van Rooyen and R. C. Williamson, "A theory of learning with corrupted labels," *Journal of Machine Learning Research*, vol. 18, no. 228, pp. 1–50, 2018.

[20] M.-K. Xie and S.-J. Huang, "CCMN: A general framework for learning with class-conditional multi-label noise," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 1, pp. 154–166, 2022.

[21] Y. Xu, P. Cao, Y. Kong, and Y. Wang, "L_DMI: An information-theoretic noise-robust loss function," *arXiv preprint arXiv:1909.03388*, 2019.

[22] L. Yong, R. Pi, W. ZHANG, X. Xia, J. Gao, X. Zhou, T. Liu, and B. Han, "A holistic view of label noise transition matrix in deep learning and beyond," in *International Conference on Learning Representations*, 2023.

[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] X. Zhou, X. Liu, D. Zhai, J. Jiang, and X. Ji, "Asymmetric loss functions for noise-tolerant learning: Theory and applications," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 7, pp. 8094–8109, 2023.

[25] Z. Zhu, Y. Song, and Y. Liu, "Clusterability as an alternative to anchor points when learning with noisy labels," in *International Conference on Machine Learning*. PMLR, 2021, pp. 12 912–12 923.

1. The official implementation is available at https://github.com/Newbeeer/L_DMI
2. These datasets are available at http://theoval.cmp.uea.ac.uk/matlab which have already been preprocessed.