To Aggregate or Not? Learning with Separate Noisy Labels

Jiaheng Wei*
University of California, Santa Cruz
Santa Cruz, CA, USA

Zhaowei Zhu*
University of California, Santa Cruz
Santa Cruz, CA, USA

Tianyi Luo
Amazon Search Science and AI
Palo Alto, CA, USA

Ehsan Amid
Google Research, Brain Team
Mountain View, CA, USA

Abhishek Kumar
Google Research, Brain Team
Mountain View, CA, USA

Yang Liu†
University of California, Santa Cruz
Santa Cruz, CA, USA

ABSTRACT

The rawly collected training data often comes with separate noisy labels collected from multiple imperfect annotators (e.g., via crowdsourcing). A typical way of using these separate labels is to first aggregate them into one and apply standard training methods. The literature has also studied extensively on effective aggregation approaches. This paper revisits this choice and aims to provide an answer to the question of whether one should aggregate separate noisy labels into single ones or use them separately as given. We theoretically analyze the performance of both approaches under the empirical risk minimization framework for a number of popular loss functions, including the ones designed specifically for the problem of learning with noisy labels. Our theorems conclude that label separation is preferred over label aggregation when the noise rates are high, or the number of labelers/annotations is insufficient. Extensive empirical results validate our conclusions.

CCS CONCEPTS

• **Computing methodologies** → **Supervised learning by classification**.

KEYWORDS

Crowdsourcing, Label Aggregation, Label Noise, Human Annotation

ACM Reference Format:
Jiaheng Wei, Zhaowei Zhu, Tianyi Luo, Ehsan Amid, Abhishek Kumar, and Yang Liu. 2023. To Aggregate or Not? Learning with Separate Noisy Labels. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '23), August 6–10, 2023, Long Beach, CA, USA.* ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/3580305.3599522

1 INTRODUCTION

Training high-quality deep neural networks for classification tasks typically requires a large quantity of annotated data. The raw training data often comes with separate noisy labels collected from multiple imperfect annotators [62] or weak proxy models [78]. For example, the popular data collection paradigm crowdsourcing [10, 17, 29] offers the platform to collect such annotations from unverified crowds; medical records are often accompanied by diagnoses from multiple doctors [1, 49]; news articles can receive multiple checkings (of the article being fake or not) from different experts [36, 39]. This leads to the situation considered in this paper: learning with multiple separate noisy labels.

The most popular approach to learning from the multiple separate labels would be aggregating the given labels for each instance [32, 44, 46, 48, 64], through an Expectation-Maximization (EM) inference technique. Each instance will then be provided with one single label, and applied with the standard training procedure.

The primary goal of this paper is to revisit the choice of aggregating separate labels and hope to provide practitioners with understandings for the following question:

Should the learner aggregate separate noisy labels for one instance into a single label or not?

Our main contributions can be summarized as follows:

- We provide theoretical insights on how separation methods and aggregation ones result in different biases (Theorem 3.4, 4.2, 4.6) and variances (Theorem 3.7, 4.3, 4.7) of the output classifier from training. Our analysis considers both the standard loss functions in use, as well as popular robust losses that are designed for the problem of learning with noisy labels.
- By comparing the analytical proxy of the worst-case performance bounds, our theoretical results reveal that separating multiple noisy labels is preferred over label aggregation when the noise rates are high, or the number of labelers/annotations is insufficient. The results are consistent for both the basic loss function ℓ and robust designs, including loss correction and peer loss.
- We carry out extensive experiments using both synthetic and real-world datasets to validate our theoretical findings.

1.1 Related Works

Label separation vs label aggregation. Existing works mainly compare the separation with aggregation by empirical results. For example, it has been shown that label separation could be effective in improving model performance and may be potentially more preferable than aggregated labels through majority voting [18]. When training with the cross-entropy loss, Sheng et.al [50] observe that label separation reduces the bias and roughness, and outperforms majority-voting aggregated labels. However, it is unclear whether the results hold when robust treatments are employed. Similar problems have also been studied in corrupted label detection with a result leaning towards separation but not proved [73]. Another line of approach concentrates on the end-to-end training

*Both authors contributed equally to this research.
†Correspondence to yangliu@ucsc.edu.

KDD '23, August 6–10, 2023, Long Beach, CA, USA

ACM ISBN 979-8-4007-0103-0/23/08.
https://doi.org/10.1145/3580305.3599522

scheme or ensemble methods which take all the separate noisy labels as the input during the training process [5, 13, 47, 59, 72], and learning from separate noisy labels directly.

Learning with noisy labels. Popular approaches in learning with noisy labels could be broadly divided into following categories, i.e., (i) Adjusting the loss on noisy labels by: using the knowledge of noise label transition matrix [22, 37, 38, 68, 76, 77]; re-weighting the per-sample loss by down-weighting instances with potentially wrong labels [3, 4, 20, 26, 34, 63]; or refurbishing the noisy labels [30, 45, 60, 69]; (ii) Robust loss designs that do not require the knowledge of noise transition matrix [2, 28, 33, 54, 55, 61, 74]; (iii) Regularization techniques to prevent deep neural networks from memorizing noisy labels [7, 24, 25, 57, 58, 65]; (iv) Dynamical sample selection procedure which behaves in a semi-supervised manner and begins with a clean sample selection procedure, then makes use of the wrongly-labeled samples [6, 23, 31, 66]. For example, several methods [15, 56, 70] adopt a mentor/peer network to select small-loss samples as "clean" ones for the student/peer network. See [14, 52] for a more detailed survey of existing noise-robust techniques.

2 FORMULATION

Consider an M-class classification task and let $X \in \mathcal{X}$ and $Y \in \mathcal{Y} := \{1, 2, ..., M\}$ denote the input examples and their corresponding labels, respectively. We assume that $(X, Y) \sim \mathcal{D}$, where $\mathcal{D}$ is the joint data distribution. Samples (x, y) are generated according to random variables (X, Y). In the clean and ideal scenario, the learner has access to N training data points $D := \{(x_n, y_n)\}_{n\in[N]}$. Instead of having access to ground truth labels y_ns, we only have access to a set of noisy labels $\{\tilde{y}^\circ_{n,i}\}_{i\in[K]}$ for $n \in [N]$. For ease of presentation, we adopt the decorator $\circ$ to denote separate labels, and $\bullet$ for aggregated labels specified later. Noisy labels $\tilde{y}^\circ_n$s are generated according to the random variable $\widetilde{Y}^\circ$. We consider the class-dependent label noise transition [26, 37] where $\widetilde{Y}^\circ$ is generated according to a transition matrix T° with its entries defined as follows:

$$T^\circ_{k,l} := \mathbb{P}(\widetilde{Y}^\circ = l | Y = k).$$

Most of the existing results on learning with noisy labels have considered the setting where each x_n is paired with only one noisy label $\tilde{y}^\circ_n$. In practice, we often operate in a setting where each data point x_n is associated with multiple separate labels drawn from the same noisy label generation process [11, 27]. We consider this setting and assume that for each x_n, there are K independent noisy labels $\tilde{y}^\circ_{n,1}, ..., \tilde{y}^\circ_{n,K}$ obtained from K annotators [53].

We are interested in two popular ways to leverage multiple separate noisy labels:

- Keep the separate labels as separate ones and apply standard learning with noisy labels techniques to each of them.
- Aggregate noisy labels into one label, and then apply standard learning with noisy data techniques.

We will look into each of the above two settings separately and then answer the question:
"Should the learner aggregate multiple separate noisy labels or not?"

2.1 Label Separation

Denote the column vector $\mathbb{P}_{\widetilde{Y}^\circ} := [\mathbb{P}(\widetilde{Y}^\circ = 1), \cdots, \mathbb{P}(\widetilde{Y}^\circ = M)]^\top$ as the marginal distribution of $\widetilde{Y}^\circ$. Accordingly, we can define $\mathbb{P}_Y$ for Y. Clearly, we have the relation: $\mathbb{P}_{\widetilde{Y}^\circ} = T^\circ \cdot \mathbb{P}_Y, \mathbb{P}_Y = (T^\circ)^{-1} \cdot \mathbb{P}_{\widetilde{Y}^\circ}$. Denote by $\rho^\circ_1 := \mathbb{P}(\widetilde{Y}^\circ = 0|Y = 1), \rho^\circ_0 := \mathbb{P}(\widetilde{Y}^\circ = 1|Y = 0)$. The noise transition matrix T has the following form when $M = 2$:

$$T^\circ = \begin{bmatrix} 1-\rho^\circ_0 & \rho^\circ_0 \\ \rho^\circ_1 & 1-\rho^\circ_1 \end{bmatrix}.$$

For label separation, we define the per-sample loss function as:

$$\ell(f(x_n), \tilde{y}^\circ_{n,1}, ..., \tilde{y}^\circ_{n,K}) = \frac{1}{K} \sum_{i\in[K]} \ell(f(x_n), \tilde{y}^\circ_{n,i}).$$

For simplicity, we shorthand $\ell(f(x_n), \tilde{y}^\circ_n) := \ell(f(x_n), \tilde{y}^\circ_{n,1}, ..., \tilde{y}^\circ_{n,K})$ for the loss of label separation method when there is no confusion.

2.2 Label Aggregation

The other way to leverage multiple separate noisy labels is generating a single label via label aggregation methods using K noisy ones:

$$\tilde{y}^\bullet_n := \text{Aggregation}(\tilde{y}^\circ_{n,1}, \tilde{y}^\circ_{n,2}, ..., \tilde{y}^\circ_{n,K}),$$

where the aggregated noisy labels $\tilde{y}^\bullet_n$s are generated according to the random variable $\widetilde{Y}^\bullet$. Denote the confusion matrix for this single & aggregated noisy label as $T^\bullet$. Popular aggregation methods include majority vote and EM inference, which are covered by our theoretical insights since our analyses in later sections would be built on the general label aggregation method. For a better understanding, we introduce the majority vote as an example.

An Example of Majority Vote. Given the majority voted label, we could compute the transition matrix between $\widetilde{Y}^\bullet$ and the true label Y using the knowledge of T°. The lemma below gives the closed form for $T^\bullet$ in terms of T°, when adopting majority vote.

Lemma 2.1. *Assume K is odd and recall that in the binary classification task, $T^\circ_{i,j} = \mathbb{P}(\widetilde{Y}^\circ = j|Y = i)$, the noise transition matrix of the (majority voting) aggregated noisy labels $T^\bullet_{p,q}$ becomes:*

$$T^\bullet_{p,q} = \sum_{i=0}^{\frac{K+1}{2}-1} \binom{K}{i} (T^\circ_{p,q})^{K-i} (T^\circ_{p,1-q})^i, \quad p, q \in \{0, 1\}.$$

When $K = 3$, then $T^\bullet_{1,0} = \mathbb{P}(\widetilde{Y}^\bullet = 0|Y = 1) = (T^\circ_{1,0})^3 + \binom{3}{1}(T^\circ_{1,0})^2(T^\circ_{1,1})$. Note it still holds that $T^\bullet_{p,q} + T^\bullet_{p,1-q} = 1$.

For the aggregation method, as illustrated in Figure 1, the x-axis indicates the number of labelers K, and the y-axis denotes the aggregated noise rate given that the overall noise rate is in $[0.2, 0.4, 0.6, 0.8]$. When the number of labelers is large (i.e., $K < 10$) and the noise rate is small, both majority vote and EM label aggregation methods significantly reduce the noise rate. Although the expectation maximization method consumes much more time when generating the aggregated label, it frequently results in a lower aggregated noise rate than majority vote.

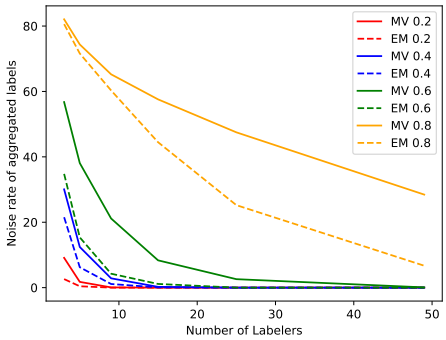

Figure 1: Noise rates of the aggregated labels in synthetic noisy CIFAR-10. MV: majority vote. EM: expectation maximization. 0.2–0.8: Original noise rates before aggregation.

3 BIAS AND VARIANCE ANALYSES W.R.T. ℓ-LOSS

In this section, we provide theoretical insights on how label separation and aggregation methods result in different biases and variances of the classifier prediction, when learning with the standard loss function ℓ.

Suppose the clean training samples $\{(x_n, y_n)\}_{n\in[N]}$ are given by variables (X, Y) such that $(X, Y) \sim \mathcal{D}$. Recall that instead of having access to a set of clean training samples $D = \{(x_n, y_n)\}_{n\in[N]}$, the learner only observes K noisy labels $\tilde{y}^\circ_{n,1}, ..., \tilde{y}^\circ_{n,K}$ for each x_n, denoted by $\widetilde{D}^\circ := \{(x_n, \tilde{y}^\circ_{n,1}, ..., \tilde{y}^\circ_{n,K})\}_{n\in[N]}$. For separation methods, the noisy training samples are obtained through variables $(X, \widetilde{Y}^\circ_1)$, ..., $(X, \widetilde{Y}^\circ_K)$ where $(X, \widetilde{Y}^\circ_i) \sim \widetilde{\mathcal{D}}^\circ$ for $i \in [K]$. For aggregation methods such as majority vote, we assume the data points and aggregated noisy labels $\widetilde{D}^\bullet := \{(x_n, \tilde{y}^\bullet_n)\}_{n\in[N]}$ are drawn from $(X, \widetilde{Y}^\bullet) \sim \widetilde{\mathcal{D}}^\bullet$ where $\widetilde{Y}^\bullet$ is produced through the majority voting of $\widetilde{Y}^\circ_1, ..., \widetilde{Y}^\circ_K$. When we mention "noise rate", we usually refer to the average noise: $\mathbb{P}(\widetilde{Y}^u \neq Y)$.

ℓ-risk under the distribution. Given the loss ℓ, note that $\ell(f(x_n), \tilde{y}^\circ_n)$ is denoted as $\ell(f(x_n), \tilde{y}^\circ_{n,1}, ..., \tilde{y}^\circ_{n,K}) = \frac{1}{K}\sum_{i\in[K]} \ell(f(x_n), \tilde{y}^\circ_{n,i})$, we define the empirical ℓ-risk for learning with separated/aggregated labels under noisy labels as: $\hat{R}_{\ell,\widetilde{D}^u}(f) = \frac{1}{N}\sum_{i=1}^N \ell\left(f(x_i), \tilde{y}^u_i\right)$, $u \in \{\circ, \bullet\}$ unifies the treatment which is either separation $\circ$ or aggregation $\bullet$. By increasing the sample size N, we would expect $\hat{R}_{\ell,\widetilde{D}^u}(f)$ to be close to the following ℓ-risk under the noisy distribution $\widetilde{\mathcal{D}}^u$: $R_{\ell,\widetilde{\mathcal{D}}^u}(f) = \mathbb{E}_{(X,\widetilde{Y}^u)\sim\widetilde{\mathcal{D}}^u}[\ell(f(X), \widetilde{Y}^u)]$.

3.1 Bias of a Given Classifier w.r.t. ℓ-Loss

We denote by $f^* \in \mathcal{F}$ the optimal classifier obtained through the clean data distribution $(X, Y) \sim \mathcal{D}$ within the hypothesis space $\mathcal{F}$. We formally define the bias of a given classifier $\hat{f}$ as:

Definition 3.1 (Classifier Prediction Bias of ℓ-Loss)**.** Denote by $R_{\ell,\mathcal{D}}(\hat{f}) := \mathbb{E}_{\mathcal{D}}[\ell(\hat{f}(X), Y)]$, $R_{\ell,\mathcal{D}}(f^*) := \mathbb{E}_{\mathcal{D}}[\ell(f^*(X), Y)]$. The bias of classifier $\hat{f}$ writes as: $\text{Bias}(\hat{f}) = R_{\ell,\mathcal{D}}(\hat{f}) - R_{\ell,\mathcal{D}}(f^*)$.

The Bias term quantifies the prediction bias (excess risk) of a given classifier $\hat{f}$ on the clean data distribution $\mathcal{D}$ w.r.t. the optimal achievable classifier f^*, which can be decomposed as [75]

$$\text{Bias}(\hat{f}) = \underbrace{R_{\ell,\mathcal{D}}(\hat{f}) - R_{\ell,\widetilde{\mathcal{D}}^u}(\hat{f})}_{\text{Distribution shift}} + \underbrace{R_{\ell,\widetilde{\mathcal{D}}^u}(\hat{f}) - R_{\ell,\mathcal{D}}(f^*)}_{\text{Estimation error}}. \quad (1)$$

Now we bound the distribution shift and the estimation error in the following two lemmas.

Lemma 3.2 (Distribution shift)**.** *Denote by $p_i := \mathbb{P}(Y = i)$, assume ℓ is upper bounded by $\bar{\ell}$ and lower bounded by $\underline{\ell}$. The distribution shift in Eqn. (1) is upper bounded by*

$$R_{\ell,\mathcal{D}}(\hat{f}) - R_{\ell,\widetilde{\mathcal{D}}^u}(\hat{f}) \leq \overline{\Delta}^{u,1}_R := (\rho^u_0 p_0 + \rho^u_1 p_1)\cdot\left(\bar{\ell} - \underline{\ell}\right). \quad (2)$$

Lemma 3.3 (Estimation error)**.** *Suppose the loss function $\ell(f(x), y)$ is L-Lipschitz for any feasible y. $\forall f \in \mathcal{F}$, with probability at least $1-\delta$, the estimation error is upper bounded by*

$$R_{\ell,\widetilde{\mathcal{D}}^u}(\hat{f}) - R_{\ell,\mathcal{D}}(f^*) \leq \overline{\Delta}^{u,2}_R := 4L\cdot\mathfrak{R}(\mathcal{F}) + (\bar{\ell}-\underline{\ell})\cdot\sqrt{\frac{2\log(1/\delta)}{\eta^u_K N}} + \overline{\Delta}^{u,1}_R,$$

where $u \in \{\circ, \bullet\}$ denotes either separation or aggregation methods, $\eta^\circ_K = \frac{K\cdot\log(\frac{1}{\delta})}{2\left(\log(\frac{K+1}{\delta})\right)^2}$ and $\eta^\bullet_K \equiv 1$ indicate the richness factor, which characterizes the effect of the number of labelers, and $\mathfrak{R}(\mathcal{F})$ is the Rademacher complexity of $\mathcal{F}$.

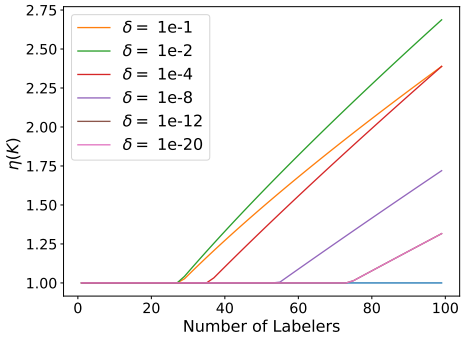

Figure 2: The visualization of estimated η°_K given varied δ.

Noting that the number of unique instances x_i are the same for both treatments, the duplicated copies of x_i are supposed to introduce at least no less effective samples, i.e., the richness factor satisfies that $\eta^u_K \geq 1$. Thus, we update $\eta^\circ_K := \max\{\eta^\circ_K, 1\}$, and Figure 2 visualizes the estimated η°_K given different number of labelers as well as δ. It is clear that when the number of labelers is larger, or δ is smaller, $\eta^\circ_K > \eta^\bullet_K$. Later we shall show how η^u_K influences the bias and variance of the classifier prediction.

To give a more intuitive comparison of the performance of both mechanisms, we adopt the worst-case bias upper bound $\overline{\Delta}^u_R := \overline{\Delta}^{u,1}_R + \overline{\Delta}^{u,2}_R$ from Lemma 3.2 and Lemma 3.3 as a proxy and derive Theorem 3.4.

Theorem 3.4. *Denote by $\alpha_K := (\rho^\circ_0 p_0 + \rho^\circ_1 p_1) - (\rho^\bullet_0 p_0 + \rho^\bullet_1 p_1)$, $\gamma = \sqrt{\log(1/\delta)/2N}$. With probability $\geq 1-\delta$, the separation bias proxy $\overline{\Delta}^\circ_R$ is smaller than the aggregation bias proxy $\overline{\Delta}^\bullet_R$ if and only if*

$$\alpha_K \cdot \frac{1}{1-(\eta^\circ_K)^{-\frac{1}{2}}} \leq \gamma. \quad (3)$$

Note that α_K and η_K° are non-decreasing w.r.t. the increase of K, in Section 4.3, we will explore how the LHS of Eqn. (3) is influenced by K: a short answer is that the LHS of Eqn. (3) is (generally) monotonically increasing w.r.t. K when K is small, indicating that Eqn. (3) is easier to be achieved given fixed δ, N and a smaller K than a larger one.

To extend the theoretical conclusions w.r.t. ℓ loss to the multi-class setting, we only need to modify the upper bound of the distribution shift in Eqn. (2), as specified in the following corollary.

Corollary 3.5 (Multi-Class Extension (ℓ-Loss)). *In the M-class classification case, the upper bound of the distribution shift in Eqn. (2) becomes:*

$$R_{\ell,\mathcal{D}}(\hat{f}) - R_{\ell,\widetilde{\mathcal{D}}^u}(\hat{f}) \leq \overline{\Delta}_R^{u,1} := \sum_{j\in[M]} p_j \cdot (1 - T^u_{j,j}) \cdot \left(\overline{\ell} - \underline{\ell}\right). \quad (4)$$

3.2 Variance of a Given Classifier w.r.t. ℓ-Loss

We now move on to explore the variance of a given classifier when learning with ℓ-loss, prior to the discussion, we define the variance of a given classifier as:

Definition 3.6 (Classifier Prediction Variance of ℓ-Loss). The variance of a given classifier $\hat{f}$ when learned with separation ($\circ$) or aggregation ($\bullet$) is defined as:

$$\mathrm{Var}(\hat{f}) = \mathbb{E}_{(X,\widetilde{Y}^u)\sim\widetilde{\mathcal{D}}^u}\left[\ell(\hat{f}(X),\widetilde{Y}^u) - \mathbb{E}_{(X,\widetilde{Y}^u)\sim\widetilde{\mathcal{D}}^u}[\ell(\hat{f}(X),\widetilde{Y}^u)]\right]^2.$$

For $g(x) = x - x^2$, we derive the closed form of Var and the corresponding upper bound as below.

Theorem 3.7. *When $\eta_K^u \geq \frac{2\log(1/\delta)}{N}$, given ℓ is 0-1 loss, we have:*

$$Var(\hat{f}^u) = g(R_{\ell,\widetilde{\mathcal{D}}^u}(\hat{f}^u)) \leq \overbrace{g\left(\sqrt{\frac{2\log(1/\delta)}{\eta_K^u N}}\right)}^{\text{Variance proxy}}. \quad (5)$$

The variance proxy of $Var(\hat{f}^\circ)$ in Eqn. (5) is smaller than that of $Var(\hat{f}^\bullet)$.

4 BIAS AND VARIANCE ANALYSES WITH ROBUST TREATMENTS

Intuitively, the learning of noisy labels problem could benefit from more robust loss functions build upon the generic ℓ loss, i.e., backward correction (surrogate loss) [37, 38], and peer loss functions [28]. We move on to explore the best way to learn with multiple copies of noisy labels, when combined with existing robust approaches.

4.1 Backward Loss Correction

When combined with the backward loss correction approach ($\ell \to \ell_\leftarrow$), the empirical ℓ risks become: $\hat{R}_{\ell_\leftarrow,\widetilde{D}^u}(f) = \frac{1}{N}\sum_{i=1}^N \ell_\leftarrow(f(x_i),\tilde{y}_i^u)$, where the corrected loss in the binary case is defined as

$$\ell_\leftarrow(f(x),\tilde{y}^u) = \frac{(1-\rho_{1-\tilde{y}^u})\cdot\ell(f(x),\tilde{y}^u) - \rho_{\tilde{y}^u}\cdot\ell(f(x),1-\tilde{y}^u)}{1-\rho_0^u-\rho_1^u}.$$

Bias of given classifier w.r.t. $\ell_\leftarrow$. Suppose the loss function $\ell(f(x),y)$ is L-Lipschitz for any feasible y. Define $L^u_\leftarrow := L^u_{\leftarrow 0}\cdot L$, where $L^u_{\leftarrow 0} := \frac{(1+|\rho_0^u-\rho_1^u|)}{1-\rho_0^u-\rho_1^u}$. Denote by $R_{\ell,\mathcal{D}}(\hat{f})$ the ℓ-risk of the classifier $\hat{f}$ under the clean data distribution $\mathcal{D}$, with $\hat{f} = \hat{f}^u_\leftarrow = \arg\min_{f\in\mathcal{F}}\hat{R}_{\ell_\leftarrow,\widetilde{D}^u}(f)$. Lemma 4.1 gives the upper bound of classifier prediction bias when learning with $\ell_\leftarrow$ via separation or aggregation methods.

Lemma 4.1. *With probability at least $1-\delta$, we have:*

$$R_{\ell,\mathcal{D}}(\hat{f}^u_\leftarrow) - R_{\ell,\mathcal{D}}(f^*) \leq \overline{\Delta}^u_{R\leftarrow} := 4L^u_\leftarrow\cdot\mathfrak{R}(\mathcal{F}) + L^u_{\leftarrow 0}\cdot(\overline{\ell}-\underline{\ell})\cdot\sqrt{\frac{2\log(1/\delta)}{\eta_K^u N}}.$$

Lemma 4.1 offers the upper bound of the performance gap for the given classifier f w.r.t the clean distribution $\mathcal{D}$, comparing to the minimum achievable risk. We consider the bound $\overline{\Delta}^u_{R\leftarrow}$ as a proxy of the bias, and we are interested in the case where training the classifier separately yields a smaller bias proxy compared to that of the aggregation method, formally $\overline{\Delta}^\circ_{R\leftarrow} < \overline{\Delta}^\bullet_{R\leftarrow}$. For any finite hypothesis class $\mathcal{F} \subset \{f : X \to \{0,1\}\}$, and the sample set $S = \{x_1, ..., x_N\}$, denote by d the VC-dimension of $\mathcal{F}$, we give conditions when training separately yields a smaller bias proxy.

Theorem 4.2. *Denote by $\alpha_K := 1 - L^\bullet_\leftarrow/L^\circ_\leftarrow, \gamma = 1/\left(1 + \frac{4L}{\overline{\ell}-\underline{\ell}}\sqrt{\frac{d\log(N)}{\log(1/\delta)}}\right)$, where d is the VC-dimension of $\mathcal{F}$. With probability $\geq 1-\delta$, for backward loss correction, the separation bias proxy $\overline{\Delta}^\circ_{R\leftarrow}$ is smaller than the aggregation bias proxy $\overline{\Delta}^\bullet_{R\leftarrow}$ if and only if*

$$\alpha_K \cdot \frac{1}{1-(\eta_K^\circ)^{-\frac{1}{2}}} \leq \gamma. \quad (6)$$

We defer our empirical analysis of the monotonicity of the LHS in Eqn. (6) to Section 4.3 as well, which shares similar monotonicity behavior to learning w.r.t. ℓ.

Variance of given classifiers with Backward Loss Correction. Similar to the previous subsection, we now move on to check how separation and aggregation methods result in different variance when training with loss correction.

Theorem 4.3. *When $L^u_{\leftarrow 0}(\eta_K^u)^{-\frac{1}{2}} < \sqrt{\frac{N}{2(\overline{\ell}-\underline{\ell})^2\log(1/\delta)}}$, $Var(\hat{f}^u_\leftarrow)$ (w.r.t. the 0-1 loss) satisfies:*

$$Var(\hat{f}^u_\leftarrow) = g(R_{\ell,\widetilde{\mathcal{D}}^u}(\hat{f}^u_\leftarrow)) \leq \overbrace{g\left(L^u_{\leftarrow 0}\cdot(\overline{\ell}-\underline{\ell})\cdot\sqrt{\frac{2\log(1/\delta)}{\eta_K^u N}}\right)}^{\text{Variance proxy}}. \quad (7)$$

The variance proxy of $Var(\hat{f}^\circ_\leftarrow)$ in Eqn. (7) is smaller than that of $Var(\hat{f}^\bullet_\leftarrow)$ if $\sqrt{\eta_K^\circ} > \frac{L^\circ_\leftarrow}{L^\bullet_\leftarrow}$.

Moving a bit further, when the noise transition matrix is symmetric for both methods, the requirement $\sqrt{\eta_K^u} > \frac{L^\circ_\leftarrow}{L^\bullet_\leftarrow}$ could be further simplified as: $\sqrt{\eta_K^u} > \frac{L^\circ_\leftarrow}{L^\bullet_\leftarrow} = \frac{1-\rho_0^\bullet-\rho_1^\bullet}{1-\rho_0^\circ-\rho_1^\circ}$. For a fixed K, a more efficient aggregation method decreases $\rho_i^\bullet$, which makes it harder to satisfy this condition.

Recall $L^u_{\leftarrow} := L^u_{\leftarrow 0} \cdot L$, the theoretical insights of $\ell_{\leftarrow}$ between binary case and the multi-class setting could be bridged by replacing L^u_0 with the multi-class constant specified in the following corollary.

Corollary 4.4 (Multi-Class Extension ($\ell_{\leftarrow}$-Loss)). *Given a diagonal-dominant transition matrix T^u, we have*

$$L^u_{\leftarrow 0} = \frac{2\sqrt{M}}{\lambda_{\min}(T^u)},$$

where $\lambda_{\min}(T^u)$ denotes the minimal eigenvalue of the matrix T^u. Particularly, if $T^u_{ii} < 0.5, \forall i \in [M]$, we further have

$$L^u_{\leftarrow 0} = \min\left\{\frac{1}{1-2e^u}, \frac{2\sqrt{M}}{\lambda_{\min}(T^u)}\right\}, \quad \text{where } e^u := \max_{i\in[M]}(1 - T^u_{ii}).$$

4.2 Peer Loss Functions

Peer Loss function [28] is a family of loss functions that are shown to be robust to label noise, without requiring the knowledge of noise rates. Formally, $\ell_{\leftrightarrow}(f(x_i), \tilde{y}_i) := \ell(f(x_i), \tilde{y}_i) - \ell(f(x_i^1), \tilde{y}_i^2)$, where the second term checks on mismatched data samples with $(x_i, \tilde{y}_i)$, $(x_i^1, \tilde{y}_i^1)$, $(x_i^2, \tilde{y}_i^2)$, which are randomly drawn from the same data distribution. When combined with the peer loss approach, i.e., $\ell \to \ell_{\leftrightarrow}$, the two risks become: $\hat{R}_{\ell_{\leftrightarrow}, \widetilde{D}^u}(f) = \frac{1}{N}\sum_{i=1}^N \ell_{\leftrightarrow}(f(x_i), \tilde{y}_i^u), u \in \{\circ, \bullet\}$.

Bias of given classifier w.r.t. $\ell_{\leftrightarrow}$. Suppose the loss function $\ell(f(x), y)$ is L-Lipschitz for any feasible y. Let $L^u_{\leftrightarrow 0} := 1/(1-\rho_0^u-\rho_1^u), L^u_{\leftrightarrow} := L^u_{\leftrightarrow 0} \cdot L$ and $\hat{f}^u_{\leftrightarrow} = \arg\min_{f\in\mathcal{F}} \hat{R}_{\ell_{\leftrightarrow}, \widetilde{D}^u}(f)$.

Lemma 4.5. *With probability at least $1-\delta$, we have:*

$$R_{\ell,\mathcal{D}}(\hat{f}^u_{\leftrightarrow}) - R_{\ell,\mathcal{D}}(f^*)$$
$$\leq \overline{\Delta}^u_{R_{\leftrightarrow}} := 8L^u_{\leftrightarrow} \cdot \mathfrak{R}(\mathcal{F}) + L^u_{\leftrightarrow 0} \cdot \sqrt{\frac{2\log(4/\delta)}{\eta_K^u N}} \cdot \left(1 + 2(\bar{\ell} - \underline{\ell})\right).$$

To evaluate the performance of a given classifier yielded by the optimization w.r.t. $\ell_{\leftrightarrow}$, Lemma 4.5 provides the bias proxy $\overline{\Delta}^u_{R_{\leftrightarrow}}$ for both treatments. Similarly, we adopt such a proxy to analyze which treatment is more preferable.

Theorem 4.6. *Denote by $\alpha_K := 1 - L^{\bullet}_{\leftrightarrow}/L^{\circ}_{\leftrightarrow}, \gamma = \frac{1+2(\bar{\ell}-\underline{\ell})}{2L}\sqrt{\frac{\log(4/\delta)}{4d\log(N)}}$, where d denotes the VC-dimension of $\mathcal{F}$. With probability $\geq 1-\delta$, for peer loss, the separation bias proxy $\overline{\Delta}^{\circ}_{R_{\leftrightarrow}}$ is smaller than the aggregation bias proxy $\overline{\Delta}^{\bullet}_{R_{\leftrightarrow}}$ if and only if*

$$\alpha_K \cdot \frac{1}{L^{\bullet}_{\leftrightarrow}/L^{\circ}_{\leftrightarrow} - (\eta_K^{\circ})^{-\frac{1}{2}}} \leq \gamma. \tag{8}$$

Note that the condition in Eqn. (8) shares a similar pattern to that which appeared in the basic loss ℓ and $\ell_{\leftarrow}$, we will empirically illustrate the monotonicity of its LHS in Section 4.3.

Variance of given classifiers with Peer Loss. We now move on to check how separation and aggregation methods result in different variances when training with peer loss. Similarly, we can obtain:

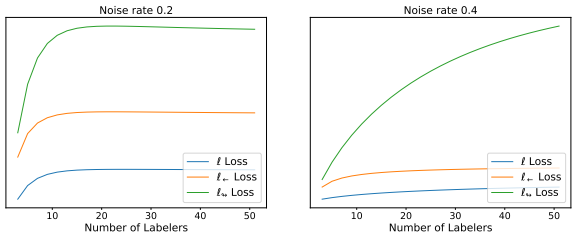

Figure 3: The monotonicity of the LHS in Eqn. (3, 6, 8) w.r.t. the increase of K: Y-axis of each line indicates the value of the LHS given fixed values of parameters that are irrelevant to K (i.e., δ, N, L, upper/lower bound of loss ℓ, etc).

Theorem 4.7. *When $\sqrt{\eta_K^u} \geq \sqrt{\frac{2\log(4/\delta)}{N}} \cdot (1 + 2(\bar{\ell} - \underline{\ell}))$, $Var(\hat{f}^u_{\leftrightarrow})$ (w.r.t. the 0-1 loss) satisfies:*

$$Var(\hat{f}^u_{\leftrightarrow}) = g(R_{\ell,\widetilde{\mathcal{D}}^u}(\hat{f}^u_{\leftrightarrow})) \leq \overbrace{g\left(L^u_{\leftrightarrow 0} \cdot \sqrt{\frac{\log(4/\delta)}{2\eta_K^u N}} \cdot \left(1 + 2(\bar{\ell} - \underline{\ell})\right)\right)}^{\text{Variance proxy}}. \tag{9}$$

The variance proxy of $Var(\hat{f}^{\circ}_{\leftrightarrow})$ in Eqn. (9) is smaller than that of $Var(\hat{f}^{\bullet}_{\leftrightarrow})$ if $\sqrt{\eta_K^{\circ}} \geq \frac{L^{\circ}_{\leftrightarrow}}{L^{\bullet}_{\leftrightarrow}}$.

Theoretical insights of $\ell_{\leftrightarrow}$ also have the multi-class extensions, we only need to generate $L^u_{\leftrightarrow 0}$ to the multi-class setting along with additional conditions specified as below:

Corollary 4.8 (Multi-Class Extension ($\ell_{\leftrightarrow}$-Loss)). *Assume $\ell_{\leftrightarrow}$ is classification-calibrated in the multi-class setting, and the clean label Y has equal prior $P(Y=j) = \frac{1}{M}, \forall j \in [M]$. For the uniform noise transition matrix [61] such that $T^u_{j,i} = \rho_i^u, \forall j \in [M]$, we have: $L^u_{\leftrightarrow 0} = 1/(1 - \sum_{i\in[M]} \rho_i^u)$.*

4.3 Analysis of the Theoretical Conditions

Recall that the established conditions in Theorems 3.4, 4.2, 4.6 are implicitly relevant to the number of labelers K, and the RHS of Eqns. (3, 6, 8) are constants. We proceed to analyze the monotonicity of the corresponding LHS (in the form of $\alpha_K \cdot \frac{1}{\beta_K - (\eta_K^{\circ})^{-\frac{1}{2}}}$) w.r.t. the increase of K, where $\beta_K = 1$ for ℓ and $\ell_{\leftarrow}$, $\beta_K = L^{\bullet}_{\leftrightarrow}/L^{\circ}_{\leftrightarrow}$ for $\ell_{\leftrightarrow}$. Thus, we have: $O(\text{LHS}) = O(\alpha_K \cdot (\beta_K - O(\frac{\log(K)}{\sqrt{K}}))^{-1})$. We visualize this order under different symmetric T° in Figure 3. It can be observed that, when K is small (e.g., $K \leq 5$), the LHS parts of these conditions increase with K, while they may decrease with K if K is sufficiently large. Recall that separation is better if LHS is less than the constant value γ. Therefore, Figure 3 shows the trends that aggregation is generally better than separation when K is sufficiently large.

Tightness of the bias proxies. In Theorems 3.4, 4.2, 4.6, we view the error bounds $\overline{\Delta}^u_R, \overline{\Delta}^u_{R_{\leftarrow}}, \overline{\Delta}^u_{R_{\leftrightarrow}}$ as proxies of the worst-case performance of the trained classifier. For the standard loss function ℓ, it has been proven that [21, 35] under mild conditions of ℓ and $\mathcal{F}$, the lower bound of the performance gap between a trained classifier

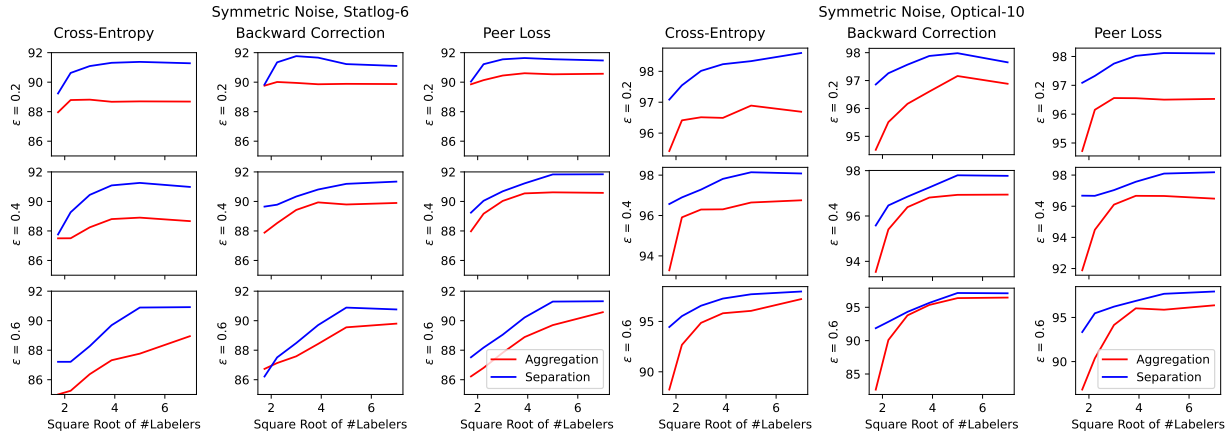

Figure 4: The performances of Cross-Entropy, Backward Loss Correction, and Peer Loss trained on synthetic noisy Statlog-6/Optical-10 aggregated labels (we report the better results between majority vote and EM inference for each K, and noise rate ϵ), and separated labels. X-axis: the value of the number of labelers $\sqrt{K}$; Y-axis: the best test accuracy achieved.

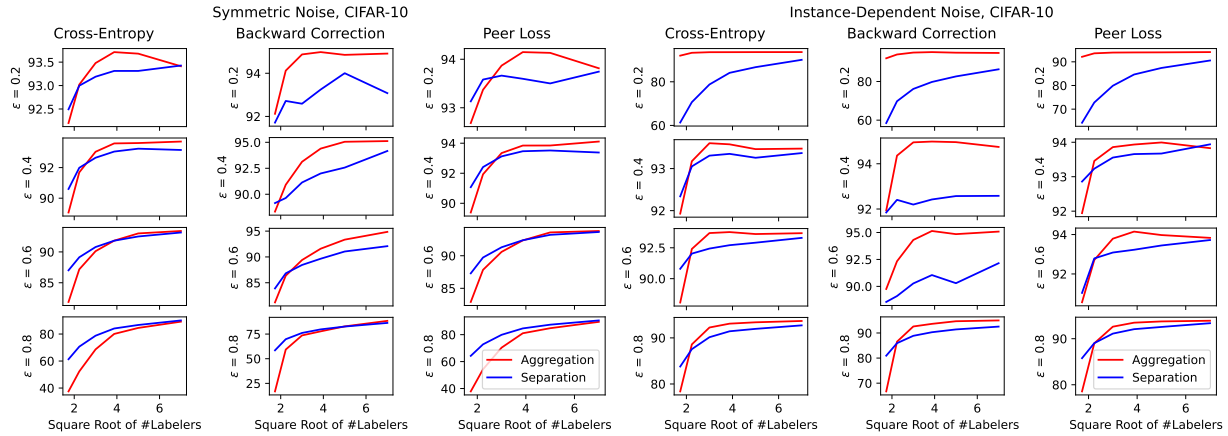

Figure 5: The performances of Cross-Entropy, Backward Loss Correction, and Peer Loss trained on synthetic noisy CIFAR-10 aggregated labels (we report the better results between majority vote, EM inference for each K, and noise rate ϵ), and separated labels. X-axis: the value of $\sqrt{K}$ where K denotes the number of labels per training example; Y-axis: the best achieved test accuracy.

$(\hat{f})$ and the optimal achievable one (i.e., f^*) $R_{\ell,\mathcal{D}}(\hat{f}) - R_{\ell,\mathcal{D}}(f^*)$ is of the order $O(\sqrt{1/N})$, which is of the same order as that in Theorem 3.4. Noting the behavior concluded from the worst-case bounds may not always hold for each individual case, we further use experiments to validate our analyses in the next section.

5 EXPERIMENTAL RESULTS

In this section, we empirically compare the performance of different treatments on the multiple noisy labels when learning with robust loss functions (CE loss, forward loss correction, and peer loss). We consider several treatments including label aggregation methods (majority vote and EM inference (also known as DS [8])) and the label separation method. Assuming that multiple noisy labels have different weights, EM inference can be used to solve the problem under this assumption by treating the aggregated labels as hidden variables [8, 41, 44, 51]. In the E-step, the probabilities of the aggregated labels are estimated using the weighted aggregation approach based on the fixed weights of multiple noisy labels. In the M-step, EM inference method re-estimates the weights of multiple noisy labels based on the current aggregated labels. This iteration continues until all aggregated labels remain unchanged. As for label separation, we adopted the mini-batch separation method, i.e., each training sample x_n is assigned with K noisy labels in each batch.

5.1 Experiment on Synthetic Noisy Datasets

Experimental results on synthetic noisy UCI datasets [9]. We adopt several UCI datasets (two binary (Breast and German) and two multiclass (StatLog and Optical) UCI classification datasets) to empirically compare the performances of label separation and aggregation methods when learning with CE loss, backward correction [37, 38],

Table 1: The performances of CE/BW/PeerLoss trained on 2 UCI datasets (Breast, and German datasets), with aggregated labels (majority vote, EM inference), and separated labels. We highlight the results with Green (for the separation method) and Red (for aggregation methods) if the performance gap is larger than 0.05. (K is the number of labels per training image)

UCI-Breast (symmetric) CE							*UCI-German (symmetric) CE*						
$\epsilon = 0.2$	$K=3$	$K=5$	$K=9$	$K=15$	$K=25$	$K=49$	$\epsilon = 0.2$	$K=3$	$K=5$	$K=9$	$K=15$	$K=25$	$K=49$
MV	96.05	96.05	96.49	96.93	97.37	97.37	MV	69.00	71.50	71.50	73.50	73.00	73.00
EM	96.93	96.05	96.49	96.93	97.37	97.37	EM	58.75	63.50	65.75	66.50	65.50	65.50
Sep	96.49	95.18	96.49	96.93	97.81	98.25	Sep	70.00	70.75	66.00	69.75	70.75	69.25
$\epsilon = 0.4$	$K=3$	$K=5$	$K=9$	$K=15$	$K=25$	$K=49$	$\epsilon = 0.4$	$K=3$	$K=5$	$K=9$	$K=15$	$K=25$	$K=49$
MV	96.05	96.49	95.18	95.18	96.49	96.93	MV	65.75	62.25	62.75	68.50	71.75	70.50
EM	96.05	92.98	89.47	94.30	96.05	96.93	EM	61.00	60.00	61.50	54.00	62.00	63.25
Sep	92.11	94.30	95.61	96.49	96.93	96.93	Sep	68.25	65.50	65.00	64.50	64.75	69.50
UCI-Breast (symmetric) BW							*UCI-German (symmetric) BW*						
$\epsilon = 0.2$	$K=3$	$K=5$	$K=9$	$K=15$	$K=25$	$K=49$	$\epsilon = 0.2$	$K=3$	$K=5$	$K=9$	$K=15$	$K=25$	$K=49$
MV	95.61	96.49	96.05	96.93	96.93	96.93	MV	72.75	71.50	74.00	75.50	76.50	76.50
EM	95.61	96.49	96.05	96.93	96.93	96.93	EM	62.75	61.50	59.25	64.50	62.50	62.50
Sep	95.18	93.42	96.49	96.05	97.37	98.25	Sep	70.50	70.50	73.75	68.25	70.00	72.75
$\epsilon = 0.4$	$K=3$	$K=5$	$K=9$	$K=15$	$K=25$	$K=49$	$\epsilon = 0.4$	$K=3$	$K=5$	$K=9$	$K=15$	$K=25$	$K=49$
MV	89.91	96.05	94.74	94.30	96.05	96.49	MV	65.25	69.50	67.50	69.50	70.50	71.75
EM	81.14	94.30	92.11	94.74	92.54	96.49	EM	57.75	60.25	55.25	53.50	54.00	62.25
Sep	91.67	93.42	94.30	89.47	92.54	97.37	Sep	60.25	63.50	63.00	64.25	69.00	64.75
UCI-Breast (symmetric) PeerLoss							*UCI-German (symmetric) PeerLoss*						
$\epsilon = 0.2$	$K=3$	$K=5$	$K=9$	$K=15$	$K=25$	$K=49$	$\epsilon = 0.2$	$K=3$	$K=5$	$K=9$	$K=15$	$K=25$	$K=49$
MV	96.05	96.49	96.49	96.93	96.93	96.93	MV	72.75	71.75	73.00	73.00	72.50	72.50
EM	96.05	96.49	96.49	96.93	96.93	96.93	EM	62.25	64.50	63.75	64.25	62.75	62.75
Sep	94.74	94.30	96.93	96.93	96.93	97.81	Sep	70.25	68.00	70.50	70.00	67.00	73.50
$\epsilon = 0.4$	$K=3$	$K=5$	$K=9$	$K=15$	$K=25$	$K=49$	$\epsilon = 0.4$	$K=3$	$K=5$	$K=9$	$K=15$	$K=25$	$K=49$
MV	92.11	95.61	95.18	92.54	96.49	96.05	MV	69.50	66.25	69.50	68.75	69.00	70.00
EM	92.11	92.11	86.40	93.86	95.61	96.93	EM	62.50	61.25	64.25	57.75	59.75	65.00
Sep	92.11	94.30	95.18	95.18	95.61	96.05	Sep	64.00	61.25	66.50	68.00	69.25	69.00

and Peer Loss [28]. As for the splitting of training and testing, the original settings are used when training and testing files are provided. Otherwise, we adopt 50/50 splitting: the numbers of (training, testing) samples in Breast, German, StatLog, and Optical datasets are (285, 284), (500, 500), (4435, 2000), and (3823, 1797). The noisy annotations given by multiple annotators are simulated by *symmetric label noise*, which assumes $T_{i,j} = \frac{\epsilon}{M-1}$ for $j \neq i$ for each annotator, where ϵ quantifies the overall noise rate of the generated noisy labels. In Figure 4, we adopt two UCI datasets (StatLog: ($M = 6$); Optical: ($M = 10$)) for illustration. From the results in Figure 4, it is quite clear that: the *label separation method outperforms both aggregation methods (majority-vote and EM inference) consistently, and is considered to be more beneficial on such small-scale datasets.* More details are deferred to the Appendix.

Experimental results on synthetic noisy CIFAR-10 dataset [19]. On CIFAR-10 dataset, we consider two types of simulation for the separate noisy labels: *symmetric label noise* model and *instance-dependent label noise* [6, 76], where ϵ is the average noise rate and different labelers follow different instance-dependent noise transition matrices. For a fair comparison, we adopt the ResNet-34 model [16], the same training procedure and batch-size for all considered treatments on the separate noisy labels.

Figure 5 shares the following insights regarding the preference of the treatments: in the low noise regime or when K is large, aggregating separate noisy labels significantly reduces the noise rates and aggregation methods tend out to have a better performance; while in the high noise regime or when K is small, the performances of separation methods tend out to be more promising. With the increasing of K or ϵ, we can observe a preference transition from label separation to label aggregation methods.

Table 2: Empirical verification of Theorem 3.4 on Breast & German UCI datasets.

Dataset	ρ_i°	p_0	N	$(1-\delta, S_K)$
Breast	0.2	0.3726	569	$(0.62, \{K > 49\})$
Breast	0.4	0.3726	569	$(0.62, \{K > 49\})$
German	0.2	0.3	1000	$(0.98, \{K > 15\})$
German	0.4	0.3	1000	$(0.98, \{K > 23\})$

5.2 Empirical Verification of the Theoretical Bounds

To verify the comparisons of bias proxies (i.e., Theorem 3.4) through an empirical perspective, we adopt two binary classification UCI datasets for demonstration: Breast and German datasets, as shown in Table 1. Clearly, on these two binary classification tasks, label aggregation methods tend to outperform label separation, and we attribute this phenomenon to the fact that "denoising effect of label aggregation is more significant in the binary case".

Table 3: Experimental results on CIFAR-10N and CIFAR-10H dataset with $K = 3$. We highlight the results with Green (for separation method) and Red (for aggregation methods) if the performance gap is large than 0.05.

CIFAR-10N ($\epsilon \approx 0.18$)	**CE**	**BW**	**PL**
Majority-Vote	89.52	89.23	89.84
EM-Inference	89.19	88.88	88.92
Separation	89.77	89.20	89.97
CIFAR-10H ($\epsilon \approx 0.09$)	**CE**	**BW**	**PL**
Majority-Vote	80.86	82.72	82.11
EM-Inference	80.81	82.43	81.73
Separation	76.75	79.07	78.08

For Theorem 3.4 (CE loss), the condition requires $\alpha_K / \left(1 - (\eta_K^\circ)^{-\frac{1}{2}}\right)$, where $\alpha = (\rho_0^\circ p_0 + \rho_1^\circ p_1) - (\rho_0^\bullet p_0 + \rho_1^\bullet p_1)$, $\gamma = \sqrt{\log(1/\delta)/2N}$. For two binary UCI datasets (Breast & German), the information could be summarized in Table 2, where the column $(1-\delta, S_K)$ means: when the number of annotators belongs to the set S_K, the label separation method is likely to under-perform label aggregation (i.e., majority vote) with probability at least $1-\delta$. For example, in the last row of Table 2, when training on UCI German dataset with CE loss under noise rate 0.4 (the noise rate of separate noisy labels), Theorem 3.4 reveals that with probability at least 0.98, label aggregation (with majority vote) is better than label separation when $K > 23$, which aligns well with our empirical observations (label separation is better only when $K < 15$).

5.3 Experiments on Realistic Noisy Datasets

Note that in real-world scenarios, the label-noise pattern may differ due to the expertise of each human annotator. We further compare the different treatments on two realistic noisy datasets: CIFAR-10N [62], and CIFAR-10H [40]. CIFAR-10N provides each CIFAR-10 train image with 3 independent human annotations, while CIFAR-10H gives ≈ 50 annotations for each CIFAR-10 test image.

In Table 3, we repeat the reproduce of three robust loss functions with three different treatments on the separate noisy labels. We report the best achieved test accuracy for Cross-Entropy/Backward Correction/Peer Loss methods when learning with label aggregation methods (majority-vote and EM inference) and the separation method (soft-label). We observe that the separation method tends to have a better performance than aggregation ones. This may be attributed to the relative high noise rate ($\epsilon \approx 0.18$) in CIFAR-N and the insufficient amount of labelers ($K = 3$). Note that since the noise level in CIFAR-10H is low ($\epsilon \approx 0.07$ wrong labels), label aggregation methods can infer higher quality labels, and thus, result in a better performance than separation methods (Red colored cells in Table 3 and 4).

5.4 Hypothesis Testing

We adopt the paired t-test to show which treatment on the separate noisy labels is better, under certain conditions. In Table 5, we report the statistic and p-value given by the hypothesis testing results. The column "Methods" indicate the two methods we want to compare (A & B). Positive statistics means that A is better than B in the metric

Table 4: Experimental results on CIFAR10-H with $K \geq 5$. We highlight the results with Green (for separation method) and Red (for aggregation methods) if the performance gap > 0.05.

CE	$K=5$	$K=9$	$K=15$	$K=25$	$K=49$
Majority-Vote	80.69	80.73	81.37	81.79	81.66
EM-Inference	80.97	80.96	81.24	81.01	81.68
Separation	79.65	80.91	81.07	80.78	80.81
BW	$K=5$	$K=9$	$K=15$	$K=25$	$K=49$
Majority-Vote	82.51	82.75	83.27	83.59	83.68
EM-Inference	82.30	82.68	82.74	82.89	83.08
Separation	82.14	82.48	81.92	81.72	81.69
PL	$K=5$	$K=9$	$K=15$	$K=25$	$K=49$
Majority-Vote	81.84	81.85	82.39	82.98	82.83
EM-Inference	81.89	82.30	82.53	82.86	82.73
Separation	80.25	81.89	81.00	80.71	80.89

Table 5: Hypothesis testing results of the comparisons between label aggregation methods and the label separation method on realistic noisy datasets.

Setting	Methods	Statistic	p-value
CIFAR-10N ($K=3$, high noise)	MV & EM	2.650	0.057
CIFAR-10N ($K=3$, high noise)	MV & Sep	-0.401	0.708
CIFAR-10N ($K=3$, high noise)	EM & Sep	-2.596	0.060
CIFAR-10H ($K<15$, low noise)	MV & EM	-0.003	0.998
CIFAR-10H ($K<15$, low noise)	MV & Sep	2.336	0.033
CIFAR-10H ($K<15$, low noise)	EM & Sep	2.390	0.030
CIFAR-10H ($K\geq 15$, low noise)	MV & EM	0.805	0.433
CIFAR-10H ($K\geq 15$, low noise)	MV & Sep	4.426	0.000
CIFAR-10H ($K\geq 15$, low noise)	EM & Sep	3.727	0.002

of test accuracy. Given a specific setting, denote by $\text{Acc}_{\text{method}}$ as the list of test accuracy that belongs to this setting (i.e., CIFAR-10N, $K = 3$), including CE, BW, PL loss functions, the basic hypothesis could be summarized as below:

- **Null hypothesis**: there exists zero mean difference between (1) Acc_{MV} and Acc_{EM}; or (2) Acc_{MV} and Acc_{Sep}; or (3) Acc_{EM} and Acc_{Sep};
- **Alternative hypothesis**: there exists non-zero mean difference between (1) Acc_{MV} and Acc_{EM}; or (2) Acc_{MV} and Acc_{Sep}; or (3) Acc_{EM} and Acc_{Sep}.

To clarify, the three cases in the above hypothesis are tested independently. For test accuracy comparisons of CIFAR-10N in Table 3, the setting of hypothesis test is $K = 3$ and the label noise rate is relatively high (18%). All p-values are larger than 0.05, indicating that we should reject the null hypothesis, and we can conclude that the performance of these three methods on CIFAR-10N (high noise, small K) satisfies: EM<MV<Sep.

For CIFAR-10H (Table 3, 4), all the label noise rate is relatively low. We consider two scenarios ($K < 15$: the number of annotators is small; $K \geq 15$: the number of annotators is large). p-values among MV and EM are always large, which means that the denoising effect of the advanced label aggregation method (EM) is negligible under CIFAR-10H dataset. However, p-values of remaining settings are larger than 0.05, indicating that we should reject the null hypothesis, and we can conclude that the performance of these 3 methods on CIFAR-10H (low noise, small/large K) satisfies: EM/MV > Sep.

Table 6: Information on the additional datasets. Regime indicates the setting (a) whether the noise rate of separated labels is large or not; (b) whether the number of annotators K is sufficient or not. The rank of label separation (Sep) denotes the performance rank (test accuracy) of label separation when compared to all other label aggregation methods. Rank 1 indicates the highest accuracy.

Dataset (# of class)	Num of annotators	Num of Train/Test data	Noise/Worker Regime	Rank of Sep
Youtube (2)	10	1586 / 250	High noise; Small K	1
Yelp (2)	8	30400 / 3800	High noise; Small K	3
Agnews (4)	9	96000 / 12000	High noise; Small K	2
Imdb (2)	5	20000 / 2500	High noise; Small K	2
Sms (2)	73	4571 / 500	High noise; Large K	1
Census (2)	83	10083 / 16281	High noise; large K	1
Mushroom (2)	22	6499 / 813	High noise; large K	1
Spambase (2)	15	3680 / 461	High noise; small K	1
Phishing (2)	15	8844 / 1106	High noise; small K	1

5.5 More Datasets and Aggregation Methods

We further made use of a popular and comprehensive weak supervision benchmark [71] and provide additional experiment results, including

- **Experiments on 5 text classification datasets**
  We adopt several popular text datasets, Youtube, Yelp, AGnews, IMDB, SMS for illustration. We compare the performance of label separation with 6 label aggregation solutions, including Majority Voting (MV), Majority Weighted Voting (MWV), Dawid-Skene (DS [8]), Data Programming (DP [43]), Flying Squid (FS [12]), and MeTaL (MTL [42]).
- **Experiments on 4 numerical classification datasets**
  We adopt several numerical datasets, Census, Mushroom, Spambase, and Phishing Websites for illustration. We compare the performance of label separation with 4 label aggregation solutions, including Majority Voting (MV), Majority Weighted Voting (MWV), Dawid-Skene (DS [8]), Data Programming (DP [43]).

Dataset information. Each sample of the above-mentioned datasets is provided with a set of labels given by certain labeling functions [71]. Please refer to Table 6 for dataset information and a brief summary of the performance comparison.

Experiment details. For text datasets, we extract features with a pre-trained BERT model. The labels of each training sample are given by the aggregated/soft label from a list of annotations. An MLP is then trained on such training datasets, with an Adam optimizer and a linear learning rate scheduler. The model is trained for 100K steps and will be early stopped by monitoring the performance on the validation set. For each setting, we report the test accuracy when achieving the best validation accuracy.

Detailed results of additional experiments. We continue to share our additional experiment results. We take the CE loss for illustration. We report the noise rate and the test accuracy for each method in Table 7, for results on the text datasets and numerical datasets, respectively. Experiment results demonstrate that label separation is competitive by comparing it with a list of label aggregation methods. Although certain label aggregation methods could reduce the noise rate effectively, **learning with separate noisy labels is always top 3 in high noise regimes, especially when the number of annotators is insufficient as well.** On the 4 numerical datasets, label separation is always the best, compared with the 4 label aggregation methods. These additional experiment results further illustrate our theoretical insights.

Table 7: Performance comparisons between label separation and label aggregation methods on text and numerical datasets, when learning with CE loss. Methods include Separation (Sep), Majority Voting (MV), Majority Weighted Voting (MWV), Dawid-Skene (DS [8]), Data Programming (DP [43]), Flying Squid (FS [12]), and MeTaL (MTL [42]). The overall noise rate (percentage of wrong labels among the training set) and the corresponding test accuracy are reported for each method under each setting. Results in bold: the best-achieved test accuracy in each dataset.

Dataset	Statistics	Sep	MV	MWV	DS	FS	DP	MTL
Youtube	Noise Rate	0.58	0.19	0.25	0.18	0.21	0.23	0.18
Youtube	Test Acc	**0.917**	0.900	0.875	0.867	0.867	0.875	0.867
Yelp	Noise Rate	0.45	0.32	0.34	0.28	0.33	0.32	0.35
Yelp	Test Acc	0.805	0.807	0.767	**0.814**	0.790	0.794	0.690
Agnews	Noise Rate	0.75	0.36	0.36	0.36	0.38	0.36	0.37
Agnews	Test Acc	0.820	0.817	**0.824**	0.772	0.741	0.810	0.752
Imdb	Noise Rate	0.49	0.29	0.29	0.29	0.30	0.29	0.30
Imdb	Test Acc	0.762	0.758	0.754	**0.766**	0.762	0.754	0.756
Sms	Noise Rate	0.27	0.32	0.32	0.09	0.13	0.32	0.13
Sms	Test Acc	**0.982**	0.940	0.940	0.974	-	0.950	-
Census	Noise Rate	0.49	0.20	0.23	0.53	-	0.22	-
Census	Test Acc	**0.797**	**0.797**	0.778	0.528	-	0.780	-
Mushroom	Noise Rate	0.46	0.13	0.14	0.14	-	0.14	-
Mushroom	Test Acc	**0.883**	0.868	0.868	0.862	-	0.866	-
Spambase	Noise Rate	0.49	0.26	0.27	0.29	-	0.29	-
Spambase	Test Acc	**0.802**	0.789	0.780	0.743	-	0.770	-
Phishing	Noise Rate	0.48	0.22	0.22	0.27	-	0.25	-
Phishing	Test Acc	**0.809**	0.800	0.784	0.741	-	0.772	-

6 CONCLUSIONS

When learning with separate noisy labels, we explore the answer to the question "whether one should aggregate separate noisy labels into single ones or use them separately as given". In the empirical risk minimization framework, we theoretically show that label separation could be more beneficial than label aggregation when the noise rates are high or the number of labelers is insufficient. These insights hold for a number of popular loss functions including several robust treatments. Empirical results on synthetic and real-world datasets validate our conclusion.

Broader Impacts. This work provides hands-on suggestions for practitioners to decide whether to keep the labels separate or not, especially for the scenario where practitioners want to make use of additional learning-with-noisy-label loss designs to further improve the robustness of the model under imperfect human annotations. What is more, treating labels separately could improve time efficiency by skipping the label aggregation procedure. We believe this takeaway will find broad applications in a variety of learning tasks.

Acknowledgement. This work is partially supported by the National Science Foundation (NSF) under grants IIS-2007951 and IIS-2143895.

REFERENCES

[1] Shadi Albarqouni, Christoph Baur, Felix Achilles, Vasileios Belagiannis, Stefanie Demirci, and Nassir Navab. 2016. Aggnet: deep learning from crowds for mitosis detection in breast cancer histology images. *IEEE transactions on medical imaging* 35, 5 (2016), 1313–1321.

[2] Ehsan Amid, Manfred K Warmuth, Rohan Anil, and Tomer Koren. 2019. Robust bi-tempered logistic loss based on Bregman divergences. *Advances in Neural Information Processing Systems* 32 (2019).

[3] Noga Bar, Tomer Koren, and Raja Giryes. 2021. Multiplicative Reweighting for Robust Neural Network Optimization. *arXiv preprint arXiv:2102.12192* (2021).

[4] Haw-Shiuan Chang, Erik Learned-Miller, and Andrew McCallum. 2017. Active bias: Training more accurate neural networks by emphasizing high variance samples. *Advances in Neural Information Processing Systems* 30 (2017).

[5] Zhijun Chen, Huimin Wang, Hailong Sun, Pengpeng Chen, Tao Han, Xudong Liu, and Jie Yang. 2020. Structured Probabilistic End-to-End Learning from Crowds.. In *IJCAI*. 1512–1518.

[6] Hao Cheng, Zhaowei Zhu, Xingyu Li, Yifei Gong, Xing Sun, and Yang Liu. 2021. Learning with Instance-Dependent Label Noise: A Sample Sieve Approach. In *International Conference on Learning Representations*. https://openreview.net/forum?id=2VXyy9mIyU3

[7] Hao Cheng, Zhaowei Zhu, Xing Sun, and Yang Liu. 2023. Mitigating Memorization of Noisy Labels via Regularization between Representations. In *International Conference on Learning Representations (ICLR)*.

[8] Alexander Philip Dawid and Allan M Skene. 1979. Maximum likelihood estimation of observer error-rates using the EM algorithm. *Journal of the Royal Statistical Society: Series C (Applied Statistics)* 28, 1 (1979), 20–28.

[9] Dheeru Dua and Casey Graff. 2017. UCI Machine Learning Repository. http://archive.ics.uci.edu/ml

[10] Enrique Estellés-Arolas and Fernando González-Ladrón-de Guevara. 2012. Towards an integrated crowdsourcing definition. *Journal of Information science* 38, 2 (2012), 189–200.

[11] Vitaly Feldman. 2020. Does learning require memorization? a short tale about a long tail. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*. 954–959.

[12] Daniel Fu, Mayee Chen, Frederic Sala, Sarah Hooper, Kayvon Fatahalian, and Christopher Ré. 2020. Fast and three-rious: Speeding up weak supervision with triplet methods. In *International Conference on Machine Learning*. PMLR, 3280–3291.

[13] Melody Guan, Varun Gulshan, Andrew Dai, and Geoffrey Hinton. 2018. Who said what: Modeling individual labelers improves classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 32.

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

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

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

[17] Jeff Howe et al. 2006. The rise of crowdsourcing. *Wired magazine* 14, 6 (2006), 1–4.

[18] Panagiotis G Ipeirotis, Foster Provost, Victor S Sheng, and Jing Wang. 2014. Repeated labeling using multiple noisy labelers. *Data Mining and Knowledge Discovery* 28, 2 (2014), 402–441.

[19] Alex Krizhevsky, Geoffrey Hinton, et al. 2009. *Learning multiple layers of features from tiny images*. Technical Report. Citeseer.

[20] Abhishek Kumar and Ehsan Amid. 2021. Constrained Instance and Class Reweighting for Robust Learning under Label Noise. *arXiv preprint arXiv:2111.05428* (2021).

[21] Guillaume Lecué and Shahar Mendelson. 2010. Sharper lower bounds on the performance of the empirical risk minimization algorithm. *Bernoulli* (2010), 605–613.

[22] Shikun Li, Xiaobo Xia, Hansong Zhang, Yibing Zhan, Shiming Ge, and Tongliang Liu. 2022. Estimating noise transition matrix with label correlations for noisy multi-label learning. In *Advances in Neural Information Processing Systems*.

[23] Sheng Liu, Kangning Liu, Weicheng Zhu, Yiqiu Shen, and Carlos Fernandez-Granda. 2021. Adaptive Early-Learning Correction for Segmentation from Noisy Annotations. *arXiv preprint arXiv:2110.03740* (2021).

[24] Sheng Liu, Jonathan Niles-Weed, Narges Razavian, and Carlos Fernandez-Granda. 2020. Early-learning regularization prevents memorization of noisy labels. *Advances in neural information processing systems* 33 (2020), 20331–20342.

[25] Sheng Liu, Zhihui Zhu, Qing Qu, and Chong You. 2022. Robust Training under Label Noise by Over-parameterization. *arXiv preprint arXiv:2202.14026* (2022).

[26] Tongliang Liu and Dacheng Tao. 2016. Classification with noisy labels by importance reweighting. *IEEE Transactions on pattern analysis and machine intelligence* 38, 3 (2016), 447–461.

[27] Yang Liu. 2021. Understanding instance-level label noise: Disparate impacts and treatments. In *International Conference on Machine Learning*. PMLR, 6725–6735.

[28] Yang Liu and Hongyi Guo. 2020. Peer loss functions: Learning from noisy labels without knowing noise rates. In *International Conference on Machine Learning*. PMLR, 6226–6236.

[29] Yang Liu and Mingyan Liu. 2015. An online learning approach to improving the quality of crowd-sourcing. *ACM SIGMETRICS Performance Evaluation Review* 43, 1 (2015), 217–230.

[30] Michal Lukasik, Srinadh Bhojanapalli, Aditya Menon, and Sanjiv Kumar. 2020. Does label smoothing mitigate label noise?. In *International Conference on Machine Learning*. PMLR, 6448–6458.

[31] Tianyi Luo, Xingyu Li, Hainan Wang, and Yang Liu. 2020. Research Replication Prediction Using Weakly Supervised Learning. In *In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*.

[32] Tianyi Luo and Yang Liu. 2019. Machine truth serum. *arXiv preprint arXiv:1909.13004* (2019).

[33] Xingjun Ma, Hanxun Huang, Yisen Wang, Simone Romano, Sarah Erfani, and James Bailey. 2020. Normalized loss functions for deep learning with noisy labels. In *International Conference on Machine Learning*. PMLR, 6543–6553.

[34] Negin Majidi, Ehsan Amid, Hossein Talebi, and Manfred K. Warmuth. 2021. Exponentiated Gradient Reweighting for Robust Training Under Label Noise and Beyond. *arXiv preprint arXiv:2104.01493* (2021).

[35] Shahar Mendelson. 2008. Lower bounds for the empirical minimization algorithm. *IEEE Transactions on Information Theory* 54, 8 (2008), 3797–3803.

[36] Tanushree Mitra and Eric Gilbert. 2015. Credbank: A large-scale social media corpus with associated credibility annotations. In *Proceedings of the International AAAI Conference on Web and Social Media*, Vol. 9. 258–267.

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

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

[39] Gordon Pennycook and David G Rand. 2019. Fighting misinformation on social media using crowdsourced judgments of news source quality. *Proceedings of the National Academy of Sciences* 116, 7 (2019), 2521–2526.

[40] Joshua C Peterson, Ruairidh M Battleday, Thomas L Griffiths, and Olga Russakovsky. 2019. Human uncertainty makes classification more robust. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 9617–9626.

[41] Nguyen Quoc Viet Hung, Nguyen Thanh Tam, Lam Ngoc Tran, and Karl Aberer. 2013. An evaluation of aggregation techniques in crowdsourcing. In *International Conference on Web Information Systems Engineering*. Springer, 1–15.

[42] Alexander Ratner, Braden Hancock, Jared Dunnmon, Frederic Sala, Shreyash Pandey, and Christopher Ré. 2019. Training complex models with multi-task weak supervision. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 4763–4771.

[43] Alexander J Ratner, Christopher M De Sa, Sen Wu, Daniel Selsam, and Christopher Ré. 2016. Data programming: Creating large training sets, quickly. *Advances in neural information processing systems* 29 (2016).

[44] Vikas C Raykar, Shipeng Yu, Linda H Zhao, Gerardo Hermosillo Valadez, Charles Florin, Luca Bogoni, and Linda Moy. 2010. Learning from crowds. *Journal of machine learning research* 11, 4 (2010).

[45] Scott Reed, Honglak Lee, Dragomir Anguelov, Christian Szegedy, Dumitru Erhan, and Andrew Rabinovich. 2014. Training deep neural networks on noisy labels with bootstrapping. *arXiv preprint arXiv:1412.6596* (2014).

[46] Filipe Rodrigues, Mariana Lourenco, Bernardete Ribeiro, and Francisco C Pereira. 2017. Learning supervised topic models for classification and regression from crowds. *IEEE transactions on pattern analysis and machine intelligence* 39, 12 (2017), 2409–2422.

[47] Filipe Rodrigues and Francisco Pereira. 2018. Deep learning from crowds. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 32.

[48] Filipe Rodrigues, Francisco Pereira, and Bernardete Ribeiro. 2014. Gaussian process classification and active learning with multiple annotators. In *International conference on machine learning*. PMLR, 433–441.

[49] Arnaud Arindra Adiyoso Setio, Alberto Traverso, Thomas De Bel, Moira SN Berens, Cas Van Den Bogaard, Piergiorgio Cerello, Hao Chen, Qi Dou, Maria Evelina Fantacci, Bram Geurts, et al. 2017. Validation, comparison, and combination of algorithms for automatic detection of pulmonary nodules in computed tomography images: the LUNA16 challenge. *Medical image analysis* 42 (2017), 1–13.

[50] Victor S Sheng, Jing Zhang, Bin Gu, and Xindong Wu. 2017. Majority voting and pairing with multiple noisy labeling. *IEEE Transactions on Knowledge and Data Engineering* 31, 7 (2017), 1355–1368.

[51] Padhraic Smyth, Usama Fayyad, Michael Burl, Pietro Perona, and Pierre Baldi. 1994. Inferring ground truth from subjective labelling of venus images. *Advances in neural information processing systems* 7 (1994).

[52] Hwanjun Song, Minseok Kim, Dongmin Park, Yooju Shin, and Jae-Gil Lee. 2022. Learning from noisy labels with deep neural networks: A survey. *IEEE Transactions on Neural Networks and Learning Systems* (2022).
[53] Wei Tang, Ming Yin, and Chien-Ju Ho. 2019. Leveraging peer communication to enhance crowdsourcing. In *The World Wide Web Conference*. 1794–1805.
[54] Jingkang Wang, Hongyi Guo, Zhaowei Zhu, and Yang Liu. 2021. Policy Learning Using Weak Supervision. *Advances in Neural Information Processing Systems* 34 (2021).
[55] Yisen Wang, Xingjun Ma, Zaiyi Chen, Yuan Luo, Jinfeng Yi, and James Bailey. 2019. Symmetric cross entropy for robust learning with noisy labels. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 322–330.
[56] Hongxin Wei, Lei Feng, Xiangyu Chen, and Bo An. 2020. Combating noisy labels by agreement: A joint training method with co-regularization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 13726–13735.
[57] Hongxin Wei, Lue Tao, Renchunzi Xie, and Bo An. 2021. Open-set label noise can improve robustness against inherent label noise. *Advances in Neural Information Processing Systems* 34 (2021).
[58] Hongxin Wei, Lue Tao, Renchunzi Xie, Lei Feng, and Bo An. 2023. Mitigating memorization of noisy labels by clipping the model prediction. In *International Conference on Machine Learning (ICML)*. PMLR.
[59] Hongxin Wei, Renchunzi Xie, Lei Feng, Bo Han, and Bo An. 2022. Deep Learning From Multiple Noisy Annotators as A Union. *IEEE Transactions on Neural Networks and Learning Systems* (2022).
[60] Jiaheng Wei, Hangyu Liu, Tongliang Liu, Gang Niu, Masashi Sugiyama, and Yang Liu. 2022. To Smooth or Not? When Label Smoothing Meets Noisy Labels. In *International Conference on Machine Learning*. PMLR, 23589–23614.
[61] Jiaheng Wei and Yang Liu. 2020. When optimizing f-divergence is robust with label noise. *arXiv preprint arXiv:2011.03687* (2020).
[62] Jiaheng Wei, Zhaowei Zhu, Hao Cheng, Tongliang Liu, Gang Niu, and Yang Liu. 2022. Learning with Noisy Labels Revisited: A Study Using Real-World Human Annotations. In *International Conference on Learning Representations*. https://openreview.net/forum?id=TBWA6PLJZQm
[63] Jiaheng Wei, Zhaowei Zhu, Gang Niu, Tongliang Liu, Sijia Liu, Masashi Sugiyama, and Yang Liu. 2023. Fairness Improves Learning from Noisily Labeled Long-Tailed Data. *arXiv preprint arXiv:2303.12291* (2023).
[64] Jacob Whitehill, Ting-fan Wu, Jacob Bergsma, Javier Movellan, and Paul Ruvolo. 2009. Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. *Advances in neural information processing systems* 22 (2009).
[65] Xiaobo Xia, Tongliang Liu, Bo Han, Chen Gong, Nannan Wang, Zongyuan Ge, and Yi Chang. 2020. Robust early-learning: Hindering the memorization of noisy labels. In *International conference on learning representations*.
[66] Xiaobo Xia, Tongliang Liu, Bo Han, Mingming Gong, Jun Yu, Gang Niu, and Masashi Sugiyama. 2022. Sample Selection with Uncertainty of Losses for Learning with Noisy Labels. In *International Conference on Learning Representations*. https://openreview.net/forum?id=xENf4QUL4LW
[67] Xiaobo Xia, Tongliang Liu, Bo Han, Nannan Wang, Mingming Gong, Haifeng Liu, Gang Niu, Dacheng Tao, and Masashi Sugiyama. 2020. Part-dependent label noise: Towards instance-dependent label noise. *Advances in Neural Information Processing Systems* 33 (2020), 7597–7610.
[68] Xiaobo Xia, Tongliang Liu, Nannan Wang, Bo Han, Chen Gong, Gang Niu, and Masashi Sugiyama. 2019. Are anchor points really indispensable in label-noise learning? *Advances in Neural Information Processing Systems* 32 (2019).
[69] Renchunzi Xie, Hongxin Wei, Lei Feng, and Bo An. 2022. GearNet: Stepwise dual learning for weakly supervised domain adaptation. *AAAI Conference on Artificial Intelligence* (2022).
[70] Xingrui Yu, Bo Han, Jiangchao Yao, Gang Niu, Ivor Tsang, and Masashi Sugiyama. 2019. How does disagreement help generalization against label corruption?. In *International Conference on Machine Learning*. PMLR, 7164–7173.
[71] Jieyu Zhang, Yue Yu, Yinghao Li, Yujing Wang, Yaming Yang, Mao Yang, and Alexander Ratner. 2021. Wrench: A comprehensive benchmark for weak supervision. *arXiv preprint arXiv:2109.11377* (2021).
[72] Zhi-Hua Zhou. 2012. *Ensemble methods: foundations and algorithms*. CRC press.
[73] Zhaowei Zhu, Zihao Dong, and Yang Liu. 2022. Detecting corrupted labels without training a model to predict. In *International Conference on Machine Learning (ICML)*.
[74] Zhaowei Zhu, Tongliang Liu, and Yang Liu. 2021. A second-order approach to learning with instance-dependent label noise. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10113–10123.
[75] Zhaowei Zhu, Tianyi Luo, and Yang Liu. 2021. The Rich Get Richer: Disparate Impact of Semi-Supervised Learning. *arXiv preprint arXiv:2110.06282* (2021).
[76] Zhaowei Zhu, Yiwen Song, and Yang Liu. 2021. Clusterability as an alternative to anchor points when learning with noisy labels. In *International Conference on Machine Learning*. PMLR, 12912–12923.
[77] Zhaowei Zhu, Jialu Wang, and Yang Liu. 2022. Beyond Images: Label Noise Transition Matrix Estimation for Tasks with Lower-Quality Features. In *International Conference on Machine Learning (ICML)*. PMLR.
[78] Zhaowei Zhu, Yuanshun Yao, Jiankai Sun, Hang Li, and Yang Liu. 2023. Weak Proxies are Sufficient and Preferable for Fairness with Missing Sensitive Attributes. In *International Conference on Machine Learning (ICML)*.

APPENDICES

A PROOF SKETCH OF CORE THEOREMS

We briefly introduce the proof sketch of Lemma 4.1 because it sets up the foundation for the analyses on Backward Loss Correction and it covers the proofs of the standard ℓ loss in Section 3 as a special case.

A.1 Proof of Lemma 4.1

Proof. Our proof can be divided into four steps as follows.

Step 1: Apply Hoeffding's inequality for each group. We divide the noisy train samples $\{(x_n, \tilde{y}^\circ_{n,k})\}_{n\in[N]}$ into K groups, for $k \in [K]$, i.e., $\{(x_n, \tilde{y}^\circ_{n,1})\}_{n\in[N]}$, $\cdots$, $\{(x_n, \tilde{y}^\circ_{n,K})\}_{n\in[N]}$. Note within each group, e.g., group $\{(x_n, \tilde{y}^\circ_{n,1})\}_{n\in[N]}$, all the N training samples are i.i.d. Additionally, training samples between any two different groups are also i.i.d. given feature set $\{x_n\}_{n\in[N]}$. Thus, with one group $\{(x_n, \tilde{y}_{n,1})\}_{n\in[N]}$, w.p. $1-\delta_0$, we have

$$\left|\hat{R}_{\mathbf{1}^\circ_{\leftarrow}|\text{Group-1}}(f) - R_{\mathbf{1}^\circ_{\leftarrow}}(f)\right| \leq \left(\overline{\mathbf{1}^\circ_{\leftarrow}} - \underline{\mathbf{1}^\circ_{\leftarrow}}\right) \cdot \sqrt{\frac{\log(1/\delta_0)}{2N}}, \forall f.$$

where we have $\overline{\mathbf{1}^\circ_{\leftarrow}} - \underline{\mathbf{1}^\circ_{\leftarrow}} := L^\circ_{\leftarrow 0} = \frac{(1+|\rho^\circ_0 - \rho^\circ_1|)}{1-\rho^\circ_0-\rho^\circ_1}$.

Step 2: Adopt the union bound for all groups. Applying the above technique on the other groups and by the union bound, we know that w.p. at least $1 - K\delta_0$, $\forall k \in [K]$, each $\hat{R}_{\mathbf{1}^\circ_{\leftarrow}|\text{Group-k}}(f), k \in [K]$ can be seen as a random variable within range:

$$\left[R_{\mathbf{1}^\circ_{\leftarrow}}(f) - L^\circ_{\leftarrow 0} \cdot \sqrt{\frac{\log(1/\delta_0)}{2N}}, R_{\mathbf{1}^\circ_{\leftarrow}}(f) + L^\circ_{\leftarrow 0} \cdot \sqrt{\frac{\log(1/\delta_0)}{2N}}\right].$$

The randomness is from noisy labels $\tilde{y}_{n,k}$.

Step 3: Hoeffding inequality for $\hat{R}_{\mathbf{1}^\circ_{\leftarrow}|\text{Group-}k}(f), k \in [K]$. These K random variables are i.i.d. when the feature set is fixed. By Hoeffding's inequality, w.p. at least $1 - K\delta_0 - \delta_1$, $\forall f$, we have

$$\left|\hat{R}_{\mathbf{1}^\circ_{\leftarrow}}(f) - R_{\mathbf{1}^\circ_{\leftarrow}}(f)\right| \leq L^\circ_{\leftarrow 0} \cdot \sqrt{\frac{\log(1/\delta_1)\log(1/\delta_0)}{NK}}.$$

Step 4: Rademacher bound on the maximal deviation. For $\delta_0 = \delta_1 = \frac{\delta}{K+1}$, with the Rademacher bound on the maximal deviation between risks and empirical ones, for $f^* \in \mathcal{F}$ and the separation method, with probability at least $1-\delta$, we have:

$$\begin{aligned}
&\max_{f\in\mathcal{F}}\left|\hat{R}_{\ell_{\leftarrow},\widetilde{D}^\circ}(f) - R_{\ell_{\leftarrow},\widetilde{\mathcal{D}}^\circ}(f)\right| \\
&\leq 2\mathfrak{R}^\circ(\ell_{\leftarrow}\circ\mathcal{F}) + L^\circ_{\leftarrow 0}\cdot(\overline{\ell}-\underline{\ell})\cdot\log\left(\frac{K+1}{\delta}\right)\cdot\sqrt{\frac{1}{NK}}, \\
&\max_{f\in\mathcal{F}}\left|\hat{R}_{\ell_{\leftarrow},\widetilde{D}^\bullet}(f) - R_{\ell_{\leftarrow},\widetilde{\mathcal{D}}^\bullet}(f)\right| \\
&\leq 2\mathfrak{R}^\bullet(\ell_{\leftarrow}\circ\mathcal{F}) + L^\bullet_{\leftarrow 0}\cdot(\overline{\ell}-\underline{\ell})\cdot\sqrt{\frac{\log(1/\delta)}{2N}},
\end{aligned}$$

where we define $\overline{\ell}, \underline{\ell}$ as the upper and lower bound of loss function ℓ respectively, and $\mathfrak{R}^u(\ell_{\leftarrow}\circ\mathcal{F})$ is the Rademacher complexity.

Step 5: Adopt the Lipshitz composition property of Rademacher averages. If ℓ is $L-$Lipshitz, then for separation and aggregation methods, $\ell_{\leftarrow}$ is $L^u_{\leftarrow}$ Lipshitz with $L^u_{\leftarrow} = \frac{(1+|\rho^u_0-\rho^u_1|)L}{1-\rho^u_0-\rho^u_1}$.

Step 6: Triangle inequality. Bound with the triangle inequality:

$$\begin{aligned}
&R_{\ell,\mathcal{D}}(\hat{f}^u_{\leftarrow}) - R_{\ell,\mathcal{D}}(f^*) = \underline{R_{\ell_{\leftarrow},\widetilde{\mathcal{D}}^u}(\hat{f}^u_{\leftarrow}) - R_{\ell_{\leftarrow},\widetilde{\mathcal{D}}^u}(f^*)} \\
&= \underline{R_{\ell_{\leftarrow},\widetilde{\mathcal{D}}^u}(\hat{f}^u_{\leftarrow})} - \hat{R}_{\ell_{\leftarrow},\widetilde{D}^u}(\hat{f}^u_{\leftarrow}) + \hat{R}_{\ell_{\leftarrow},\widetilde{D}^u}(f^*) - \underline{R_{\ell_{\leftarrow},\widetilde{\mathcal{D}}^u}(f^*)} \\
&\quad + \hat{R}_{\ell_{\leftarrow},\widetilde{D}^u}(\hat{f}^u_{\leftarrow}) - \hat{R}_{\ell_{\leftarrow},\widetilde{D}^u}(f^*) \\
&\leq 0 + 2\max_{f\in\mathcal{F}}|\hat{R}_{\ell_{\leftarrow},\widetilde{D}^u}(f) - R_{\ell_{\leftarrow},\widetilde{\mathcal{D}}^u}(f)|.
\end{aligned}$$

Conclusions could be derived then.

□

B ADDITIONAL RESULTS AND DETAILS

B.1 Assumption of the Data Simulation

The assumptions, i.e., label follows a random flipping model, and workers have the same noise rate, are made for ease of theoretical analyses. We experimented without these assumptions on real-world human annotated datasets (CIFAR-10N, CIFAR-100N, and CIFAR-10H), as shown in the main paper. For these real-world datasets, the human annotations are not simulated and are instead collected from Amazon Mechanical Turk. Therefore, the label errors are not random flipping and different workers have different noise rates. The empirical results reveal that:

- Label separation is preferable in a relatively high noise regime and an insufficient number of annotators. Please see the results of CIFAR-10N, where we only have 3 annotators for each image, and the noise rate is relatively large (around 18% wrong labels);
- Label aggregation is preferable in a low-noise regime or a sufficient number of annotators. Please see the results of CIFAR-10H in the main paper, where we only have >48 annotators for each image, and the noise rate is small (<10% wrong labels);

Besides, the assumption that workers have the same noise rate is not a strong assumption in practice. Specifically, taking the human-annotated CIFAR-10N as an example, we can view each of the three randomly collected labels as a data source, e.g., the 1st random label for each feature is from data source 1. The data source can be approximated as an "annotator." From CIFAR-N, we observe that different data sources (approximated "annotators") have almost the same error rates (data source 1: 17.23%, data source 2: 18.12%, and data source 3: 17.64). Therefore, we can safely assume that data from random data sources have the same noise rates.

B.2 Experiment Details on UCI Datasets

Generating the noisy labels on UCI datasets. For each UCI dataset adopted in this paper, the label of each sample in the training dataset will be flipped to the other classes with the probability ϵ (noise rate). For the multiclass classification datasets, the specific label which will be flipped is randomly selected with equal probabilities. For binary and multiclass classification datasets, (0.1, 0.2, 0.3, 0.4) and (0.2, 0.4, 0.6, 0.8) are used as different lists of noise rates respectively.

Implementation details. We implemented a simple two-layer ReLU Multi-Layer Perceptron (MLP) for the classification task on these four UCI datasets. The Adam optimizer is used with a learning rate of 0.001 and the batch size is 128.

B.3 Experiment Details on CIFAR-10 Datasets

The generation of symmetric noisy dataset is adopted from [61]. As for the instance-dependent label noise, the generating algorithm follows the state-of-the-art method [67]. Both cases adopt noise rates: $[0.2, 0.4, 0.6, 0.8]$. The basic hyper-parameters settings for all methods are listed as follows: mini-batch size (128), optimizer (SGD), initial learning rate (0.1), momentum (0.9), weight decay (0.0005), number of epochs (120) and learning rate decay (0.1 at 50 epochs). Standard data augmentation is applied to each dataset. All experiments run on 8 Nvidia RTX A5000 GPUs.

CIFAR-10, Symmetric CE

$\epsilon = 0.2$	$K = 3$	$K = 5$	$K = 9$	$K = 15$	$K = 25$	$K = 49$
MV	92.21	92.98	93.54	93.43	93.73	93.40
EM	92.08	92.93	93.54	93.64	93.35	93.37
Sep	92.52	92.89	93.35	93.15	93.42	93.40
$\epsilon = 0.4$	$K = 3$	$K = 5$	$K = 9$	$K = 15$	$K = 25$	$K = 49$
MV	89.09	91.59	93.18	93.43	93.26	93.44
EM	88.83	91.02	92.54	93.45	93.69	93.68
Sep	90.61	91.95	92.70	92.92	93.32	93.13
$\epsilon = 0.6$	$K = 3$	$K = 5$	$K = 9$	$K = 15$	$K = 25$	$K = 49$
MV	81.85	87.33	89.88	91.88	92.96	93.40
EM	81.04	85.91	89.76	91.57	92.55	93.10
Sep	87.00	89.19	90.70	91.97	92.40	93.17
$\epsilon = 0.8$	$K = 3$	$K = 5$	$K = 9$	$K = 15$	$K = 25$	$K = 49$
MV	20.94	44.62	70.91	79.61	84.83	89.09
EM	37.91	50.78	67.19	75.26	82.97	87.97
Sep	61.47	70.10	79.61	83.93	86.82	90.04

CIFAR-10, Symmetric BW

$\epsilon = 0.2$	$K = 3$	$K = 5$	$K = 9$	$K = 15$	$K = 25$	$K = 49$
MV	92.08	94.09	94.92	94.90	94.79	94.90
EM	92.13	93.08	94.90	94.91	94.90	94.86
Sep	91.74	92.61	92.75	92.59	94.44	92.97
$\epsilon = 0.4$	$K = 3$	$K = 5$	$K = 9$	$K = 15$	$K = 25$	$K = 49$
MV	88.28	91.11	92.73	94.60	94.62	94.81
EM	87.41	90.23	92.83	94.77	94.80	95.18
Sep	89.14	89.68	91.07	92.46	92.26	94.24
$\epsilon = 0.6$	$K = 3$	$K = 5$	$K = 9$	$K = 15$	$K = 25$	$K = 49$
MV	81.21	86.29	89.51	91.33	93.52	94.81
EM	78.13	84.33	89.44	91.17	92.45	94.60
Sep	83.84	87.05	88.10	89.80	90.95	92.11
$\epsilon = 0.8$	$K = 3$	$K = 5$	$K = 9$	$K = 15$	$K = 25$	$K = 49$
MV	16.43	60.97	71.11	77.86	82.72	88.41
EM	10.00	45.97	66.02	74.37	80.08	87.42
Sep	58.48	69.86	76.03	79.79	82.60	86.31

CIFAR-10, Symmetric PeerLoss

$\epsilon = 0.2$	$K = 3$	$K = 5$	$K = 9$	$K = 15$	$K = 25$	$K = 49$
MV	92.69	93.35	93.90	94.12	94.15	93.81
EM	92.39	93.25	93.76	93.93	93.52	93.77
Sep	93.15	93.51	93.77	93.51	93.56	93.73
$\epsilon = 0.4$	$K = 3$	$K = 5$	$K = 9$	$K = 15$	$K = 25$	$K = 49$
MV	89.40	91.88	93.42	93.84	93.83	94.04
EM	89.23	91.41	93.06	93.83	93.85	94.11
Sep	91.08	92.38	93.17	93.40	93.56	93.37
$\epsilon = 0.6$	$K = 3$	$K = 5$	$K = 9$	$K = 15$	$K = 25$	$K = 49$
MV	82.88	87.95	90.42	92.31	93.61	93.79
EM	81.64	86.45	90.09	91.98	93.23	93.58
Sep	87.28	89.80	91.19	92.42	93.18	93.65
$\epsilon = 0.8$	$K = 3$	$K = 5$	$K = 9$	$K = 15$	$K = 25$	$K = 49$
MV	21.82	48.71	72.81	80.32	85.27	89.38
EM	38.29	52.63	68.70	77.42	83.94	88.45
Sep	64.32	72.52	80.31	84.65	87.40	90.56

Table 8: The performances of CE/BW/PeerLoss trained on (symmetric noise) CIFAR-10 aggregated labels (majority vote, EM inference), and separated labels. (Different number of labels per training image)

B.4 Detailed Results on CIFAR-10 Dataset

Table 8 and 9 include all the detailed accuracy values appeared in Figure 5. The results on the synthetic noisy CIFAR-10 dataset align well with the theoretical observations: label separation is preferred over label aggregation when the noise rates are high, or the number of labelers/annotations is insufficient.

CIFAR-10, Instance CE

$\epsilon = 0.2$	$K = 3$	$K = 5$	$K = 9$	$K = 15$	$K = 25$	$K = 49$
MV	91.99	93.29	93.57	93.47	93.68	93.60
EM	91.92	93.21	93.55	93.61	93.44	93.44
Sep	92.36	92.97	93.43	93.24	93.33	93.35
$\epsilon = 0.4$	$K = 3$	$K = 5$	$K = 9$	$K = 15$	$K = 25$	$K = 49$
MV	87.14	91.15	93.10	93.15	93.23	93.48
EM	88.07	92.40	93.70	93.58	93.74	93.53
Sep	90.83	91.90	92.63	92.46	93.08	93.26
$\epsilon = 0.6$	$K = 3$	$K = 5$	$K = 9$	$K = 15$	$K = 25$	$K = 49$
MV	49.22	83.95	89.45	91.60	92.88	93.65
EM	78.34	88.79	91.95	92.97	93.46	93.65
Sep	83.79	87.55	90.15	91.58	91.86	92.74
$\epsilon = 0.8$	$K = 3$	$K = 5$	$K = 9$	$K = 15$	$K = 25$	$K = 49$
MV	14.59	25.25	34.47	57.99	57.51	87.08
EM	20.03	26.54	65.16	80.10	88.59	92.14
Sep	26.16	28.89	50.35	74.15	71.39	87.54

CIFAR-10, Instance BW

$\epsilon = 0.2$	$K = 3$	$K = 5$	$K = 9$	$K = 15$	$K = 25$	$K = 49$
MV	92.03	93.87	95.12	95.11	94.97	94.75
EM	91.93	94.39	94.90	94.84	95.05	94.54
Sep	91.93	92.07	92.70	91.75	93.02	92.47
$\epsilon = 0.4$	$K = 3$	$K = 5$	$K = 9$	$K = 15$	$K = 25$	$K = 49$
MV	86.61	90.64	93.00	94.73	94.72	94.72
EM	89.83	92.04	94.74	95.00	94.94	94.80
Sep	88.86	87.89	92.09	89.92	91.05	91.96
$\epsilon = 0.6$	$K = 3$	$K = 5$	$K = 9$	$K = 15$	$K = 25$	$K = 49$
MV	43.78	82.59	88.56	91.47	93.27	95.06
EM	44.92	87.33	91.39	93.58	94.72	94.99
Sep	80.88	86.22	88.45	90.69	91.16	92.61
$\epsilon = 0.8$	$K = 3$	$K = 5$	$K = 9$	$K = 15$	$K = 25$	$K = 49$
MV	16.00	25.03	33.80	67.91	68.52	86.49
EM	16.06	22.73	53.96	76.24	86.74	92.02
Sep	27.84	26.68	32.72	37.27	54.41	83.37

CIFAR-10, Instance PeerLoss

$\epsilon = 0.2$	$K = 3$	$K = 5$	$K = 9$	$K = 15$	$K = 25$	$K = 49$
MV	92.13	93.53	94.00	93.78	94.13	94.08
EM	91.93	93.51	93.78	93.88	94.03	93.82
Sep	92.86	93.23	93.56	93.72	93.63	93.95
$\epsilon = 0.4$	$K = 3$	$K = 5$	$K = 9$	$K = 15$	$K = 25$	$K = 49$
MV	88.15	91.61	93.21	93.64	93.84	93.69
EM	90.59	92.60	93.95	94.02	94.06	93.68
Sep	91.06	92.70	93.22	92.92	93.65	93.67
$\epsilon = 0.6$	$K = 3$	$K = 5$	$K = 9$	$K = 15$	$K = 25$	$K = 49$
MV	60.66	84.99	90.30	91.93	93.16	93.81
EM	78.53	89.11	92.44	93.17	93.96	93.85
Sep	85.76	89.07	91.05	92.22	92.45	93.39
$\epsilon = 0.8$	$K = 3$	$K = 5$	$K = 9$	$K = 15$	$K = 25$	$K = 49$
MV	14.35	24.83	40.49	65.47	69.28	88.05
EM	26.52	28.43	66.72	80.71	89.40	92.41
Sep	33.87	37.49	57.36	77.43	80.51	89.15

Table 9: The performances of CE/BW/PeerLoss trained on (instance noise) CIFAR-10 aggregated labels (majority vote, EM inference), and separated labels. (Different number of labels per training image)