Intra-Cluster Mixup: An Effective Data Augmentation Technique for Complementary-Label Learning

Anonymous authors
Paper under double-blind review

Abstract

In this paper, we investigate the challenges of *complementary-label learning (CLL)*, a specialized form of *weakly-supervised learning (WSL)* where models are trained with labels indicating classes to which instances do not belong, rather than standard ordinary labels. This alternative supervision is appealing because collecting complementary labels is generally cheaper and less labor-intensive. Although most existing research in CLL emphasizes the development of novel loss functions, the potential of data augmentation in this domain remains largely underexplored. In this work, we uncover that the widely-used Mixup data augmentation technique is ineffective when directly applied to CLL. Through in-depth analysis, we identify that the complementary-label noise generated by Mixup negatively impacts the performance of CLL models. We then propose an improved technique called *Intra-Cluster Mixup (ICM)*, which only synthesizes augmented data from nearby examples, to mitigate the noise effect. ICM carries the benefits of encouraging complementary label sharing of nearby examples, and leads to substantial performance improvements across synthetic and real-world labeled datasets. In particular, our wide spectrum of experimental results on both balanced and imbalanced CLL settings justifies the potential of ICM in allying with state-of-the-art CLL algorithms, achieving significant accuracy increases of 30% and 10% on MNIST and CIFAR datasets, respectively.

1 Introduction

Obtaining high-quality labels is often expensive, time-consuming, and sometimes impossible in real-world applications. To address this challenge, *weakly-supervised learning (WSL)* has been extensively studied in recent years. WSL aims to train a proper classifier with inaccurate, incomplete, or inexact supervision (Zhou, 2017). Contemporary WSL studies have significantly expanded our understanding of machine learning capabilities, encompassing areas such as learning from complementary labels (Ishida et al., 2017), learning from multiple complementary labels (Feng et al., 2020), learning from partial labels (Jin & Ghahramani, 2002), or a mixture of ordinary and complementary labels (Ishida et al., 2017).

This paper focuses on *complementary-label learning (CLL)* (Ishida et al., 2017), a WSL problem where a complementary label designates a class to which a specific instance *does not belong*. The CLL problem assumes that the learner only has access to complementary labels during training while still expecting the learner to predict the ordinary labels correctly during testing. Complementary labels serve as a viable alternative when it is difficult or costly to acquire ordinary labels (Ishida et al., 2017). For instance, collecting ordinary labels on numerous classes not only demands annotators with excellent knowledge for selecting the correct labels but also requires more time for accurate labeling. CLL models can extend the horizon of machine learning and make multi-class classification potentially more realistic when ordinary labels cannot be easily obtained (Ishida et al., 2017).

Existing CLL studies have primarily focused on designing loss functions that are converted from well-known ordinary classification losses (Ishida et al., 2017; Chou et al., 2020b; Ishida et al., 2018), often under the assumption that complementary labels are uniformly generated (Cao et al., 2022; Feng et al., 2020). Those loss-designing studies tackle the CLL problem *algorithmically* and enrich our understanding on possible CLL

models. Despite this focus, the potential of data augmentation in CLL remains largely unexplored. This notable gap in the literature motivates our study on developing and assessing data augmentation techniques to improve the efficacy of CLL models.

Data augmentation techniques are known to be powerful *"add-ons"* to machine learning models for enhancing their performance by improving generalization, robustness to noise, and invariance to transformations (Mikołajczyk & Grochowski, 2018; Rebuffi et al., 2021). Across a range of classification scenarios (Chou et al., 2020a), successful data augmentation techniques exhibit seamless integration with algorithmic approaches to boost their performance. Some data augmentation techniques create pseudo examples that are variations of the original examples, without re-labeling them (Shorten & Khoshgoftaar, 2019; Jiang et al., 2021). Other techniques construct synthetic examples with modified labels (Lin & Lin, 2023). Motivated by recent studies on multiple complementary-label learning (Cao et al., 2022; Feng et al., 2020), we conjecture that utilizing multiple complementary labels through label sharing has the potential to improve existing CLL models, and thus resort to label-modification techniques. Among such techniques, we choose Mixup (Zhang et al., 2018) because of its natural potential in encouraging complementary-label sharing. While Mixup is well-known for its simplicity and effectiveness (Li & Jia, 2025; Navarro & Segarra, 2023; Xie et al., 2023), the application of Mixup for CLL remains unexplored prior to our work.

With a wide spectrum of experiments across balanced and imbalanced CLL settings, we confirm that applying Mixup on the complementary labels has the potential to improve various state-of-the-art CLL models by encouraging label sharing, which helps the machine identify the ordinary label more efficiently (Lin & Lin, 2023; Chou et al., 2020b; Yu et al., 2018). But the potential comes with a serious side effect. In particular, sometimes the complementary labels on which Mixup manipulates contains an *ordinary* label of one of the examples, which introduces noise to the label sharing process. The noise significantly deteriorates the performance of the CLL model because of overfitting. The side effect suggests that original Mixup does not work off-the-shelf for CLL.

To mitigate the side effect, we design a novel data augmentation technique called *Intra-Cluster Mixup (ICM)*. ICM clusters the examples before applying Mixup *within each cluster*. The clustering design reduces the noise introduced by Mixup while keeping its potential benefits. Our empirical experiments demonstrate that ICM consistently enhances the CLL performance across a variety of state-of-the-art CLL models and a broad spectrum of settings, ranging from balanced to imbalanced classification. Furthermore, we expand our empirical comparison from 4 common benchmark datasets in existing studies to 7, including both synthetic and real-world labeled datasets. Our efforts significantly broaden the scope of benchmarks in the field. Our unique contributions can be summarized as follows:

- To the best of our knowledge, we are *the first* to introduce a novel data augmentation technique specifically designed for CLL contexts. We identify two critical insights: (i) the original Mixup fails in CLL settings due to noise introduced during the label sharing process, and (ii) mixing samples within the same class proves to be a more effective strategy.
- We propose ICM, a tailored data augmentation technique that addresses the unique challenges of CLL and consistently enhances the performance across various CLL models.
- We conduct extensive benchmarking on large CLL datasets, covering a range from *synthetic* to *real-world* labeled datasets. Our studies span a diverse spectrum of settings, from *balanced* to *imbalanced* CLL, justifying the effectiveness of our framework.

2 Problem Setup

2.1 Complementary-Label Learning

In CLL, we are given a dataset $\bar{D} = \{(\mathbf{x}_i, \bar{y}_i)\}_{i=1}^{N}$, where each instance $\mathbf{x}_i \in \mathbb{R}^d$ represents an input image, and $\bar{y}_i \in \mathbb{R}^K$ is a complementary label. The complementary label $\bar{y}_i$ indicates a class that the image $\mathbf{x}_i$ does not belong. The dataset consists of N samples, and the goal of CLL is to use this complementary label information to train a classifier. In this context, the complementary label satisfies the condition

$\bar{y}_i \in [K] \backslash \{y_i\}$, where y_i is the ordinary label of $\mathbf{x}_i$, K denotes the total number of classes in the dataset, with $K > 2$. The set $[K] = \{1, 2, \ldots, K\}$ represents the set of all possible class labels. This implies that $\bar{y}_i$ is one of the incorrect classes for the instance $\mathbf{x}_i$. The training set $\bar{D}$ is denoted as $\bar{D} = X \times \bar{Y}$, where X contains the input images and $\bar{Y}$ contains the corresponding complementary labels. In contrast to traditional multi-class classification, where the ordinary label y_i is used to train a classifier, the CLL setup trains the model using the complementary label $\bar{y}_i$. However, the objective in CLL remains the same: to train a classifier g that accurately predicts the ordinary label y_i for unseen instances. Generally, the classifier g is realized through a decision function $g\colon \mathbb{R}^d \to \mathbb{R}^K$, with the classification determined by taking the argmax on h. For example, $g(\mathbf{x}) = \operatorname{argmax}_{k \in [K]} h(\mathbf{x})_k$, where $h(\mathbf{x})_k$ represents the score or confidence that the instance $\mathbf{x}$ belongs to class k. The classifier selects the class with the highest score.

2.2 Recent Approaches of Complementary-Label Learning

Recent approaches to CLL share a common characteristic: they apply various surrogate loss functions to the standard classifier. For instance, (Ishida et al., 2017; 2019) developed an *unbiased risk estimator (URE)* for arbitrary losses on the standard classifier when employing a uniform transition matrix. When the risk is defined as the classification error, the URE serves as a surrogate metric for performance evaluation. However, UREs are prone to severe negative empirical risks during training, which is indicative of overfitting. To mitigate such overfitting in algorithm design, (Chou et al., 2020b) proposed the *surrogate complementary loss (SCL)*, which is based on minimizing the likelihood associated with complementary label. They justified this approach by showing that SCL constitutes an upper bound to a constant multiple of the standard classification error when the transition matrix is uniform. Another study by (Yu et al., 2018) examined scenarios where the complementary label are not uniformly generated. (Yu et al., 2018) introduced a framework called *forward correction (FWD)* that adapts techniques from noisy label learning (Patrini et al., 2017) to adjust the softmax cross-entropy loss. This is achieved by adding a transition layer to the output of the model: $\bar{g}(\mathbf{x}) = T^T g(\mathbf{x})$, and then utilizing the cross-entropy loss between $\bar{g}(\mathbf{x})$ and the complementary labels $\bar{y}$. Other research efforts have explored advanced applications beyond single complementary label, including learning from multiple complementary labels (Cao et al., 2022; Feng et al., 2020), and integrating learning from both ordinary and complementary labels (Katsura & Uchida, 2020).

3 Proposed Method

In this section, we propose ICM, a novel data-augmentation technique for CLL. First, we evaluate the performance of the standard Mixup method under various experimental conditions to identify the factors that undermine its effectiveness. Next, we develop enhanced augmentation algorithms that explicitly address these limitations. Finally, we derive and introduce a surrogate complementary-label loss function that seamlessly integrates ICM into the training process.

3.1 Why Mixup does not work?

Applying Mixup naively in the CLL setting results in substantial *complementary-label noise*. This noise arises when the ordinary label appears in the synthetic data generated via original Mixup, thereby violating the core assumption of CLL: $\bar{y}_i \in [K] \setminus y_i$. To empirically verify this claim, we conduct an ablation study measuring the noise ratio introduced by Mixup in a controlled CLL setting. Although CLL typically operates under the assumption that ordinary labels are unavailable or costly to obtain, we adopt a *proof-of-concept* setup where ordinary labels are accessible solely for quantifying the noise. Using the SCL-NL loss (Chou et al., 2020b) and a ResNet18 backbone (He et al., 2016) trained on CIFAR10, we observe that Mixup introduce a noise level of 15.81% (as indicated by the green triangle in Figure 2a). Notably, when training is performed under noise-free conditions, model accuracy improved by 7%, indicating high sensitivity to label noise (highlighted by the orange circle in Figure 2a). These results highlight that label noise in CLL substantially degrades performance. From these observations, we introduce a mathematical framework for analyzing complementary classification error under Mixup augmentation.

Definition 1 (Complementary classification error)**.** Let $\{(\mathbf{x}_i, \bar{y}_i)\}_{i=1}^N$ be the training examples, where $\mathbf{x}_i \in \mathbb{R}^d$ is the input and $\bar{y}_i \in \{1, \ldots, K\}$ is the complementary hard label of the i-th example. Let $g : \mathbb{R}^d \to$

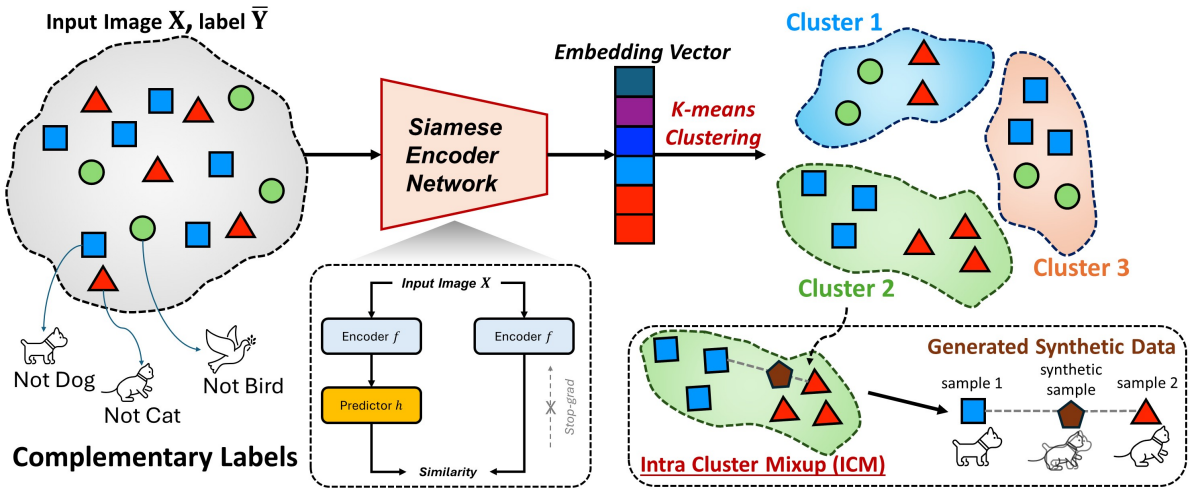


Figure 1: Illustration of the *Intra-Cluster Mixup (ICM)* framework. *Top*: Embedding features are extracted using a pretrained *SimSiam* encoder and clustered using k-means, aiming to group samples with similar ordinary labels. *Bottom right*: Within each cluster, ICM generates synthetic samples by interpolating features and labels, which are then used to train the classifier.

$\{1, \ldots, K\}$ be a classifier and let $\ell(\cdot, \cdot)$ denote a loss function. For any input $\mathbf{x}$, define the per-class loss vector $\ell\big(g(\mathbf{x})\big) = \big[\ell\big(1, g(\mathbf{x})\big), \ldots, \ell\big(K, g(\mathbf{x})\big)\big]$. Given two training samples $\mathbf{x}_i$ and $\mathbf{x}_j$ from the same cluster (j is an index randomly sampled from the same cluster as i), we construct a mixed input $\tilde{\mathbf{x}}_{i,j} = \lambda \mathbf{x}_i + (1-\lambda)\mathbf{x}_j$, where $\lambda \sim \mathrm{Beta}(\alpha, \alpha)$ and $\tilde{y}_{i,j}$ denotes the corresponding soft label generated via Mixup. The complementary classification error of g under loss ℓ is defined as

$$\mathcal{R}_{hl}(g; \ell) = \frac{1}{N} \sum_{i=1}^{N} \ell\big(\bar{y}_i, g(\mathbf{x}_i)\big) = \mathbb{E}_{(\mathbf{x}, \bar{y}) \sim \bar{D}} \big[[\![\bar{y} \neq g(\mathbf{x})]\!] \big], \tag{1}$$

[1]

For Mixup-generated pairs $(\tilde{\mathbf{x}}_{i,j}, \tilde{y}_{i,j})$, the complementary classification risk under soft labels is defined as

$$\mathcal{R}_{sl}(g; \ell) = \frac{1}{N} \sum_{i=1}^{N} \ell\Big(\tilde{y}_{i,j}, g(\tilde{\mathbf{x}}_{i,j})\Big) = \mathbb{E}_{(\mathbf{x}, \bar{y}) \sim \bar{D}} [\![\bar{y}_i \neq g(\tilde{\mathbf{x}}_{i,j})]\!] + \mathbb{E}_{(\mathbf{x}, \bar{y}) \sim \bar{D}} [\![\bar{y}_j \neq g(\tilde{\mathbf{x}}_{i,j})]\!] . \tag{2}$$

Definition 2 (Error generated by label noise)**.** The error generated by label noise for the classifier g is defined as

$$\varepsilon(g) = \mathbb{E}_{(\mathbf{x}, \bar{y}) \sim \bar{D}} [\![\bar{y} = g(\mathbf{x})]\!] , \tag{3}$$

that is, $\varepsilon(g)$ is the probability that g predicts the complementary label itself.

Proposition 1 (Complementary error with Mixup)**.** For Mixup-generated pairs $(\tilde{\mathbf{x}}_{i,j}, \tilde{y}_{i,j})$, the complementary classification risk under Mixup is

$$\mathcal{R}'(g; \ell) = \frac{1}{N} \sum_{i=1}^{N} \ell\big(\tilde{y}_{i,j}, g(\tilde{\mathbf{x}}_{i,j})\big), \tag{4}$$

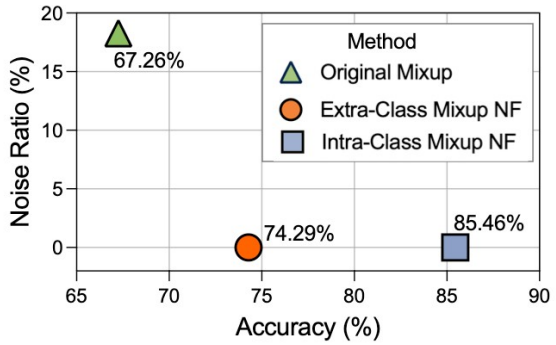
and admits the decomposition

$$\mathcal{R}'(g; \ell) = \lambda \mathbb{E}_{(\mathbf{x}, \bar{y}) \sim \bar{D}} [\![\bar{y}_i \neq g(\tilde{\mathbf{x}}_{i,j})]\!] + (1-\lambda) \mathbb{E}_{(\mathbf{x}, \bar{y}) \sim \bar{D}} [\![\bar{y}_j \neq g(\tilde{\mathbf{x}}_{i,j})]\!] + \lambda \varepsilon_i + (1-\lambda) \varepsilon_j . \tag{5}$$

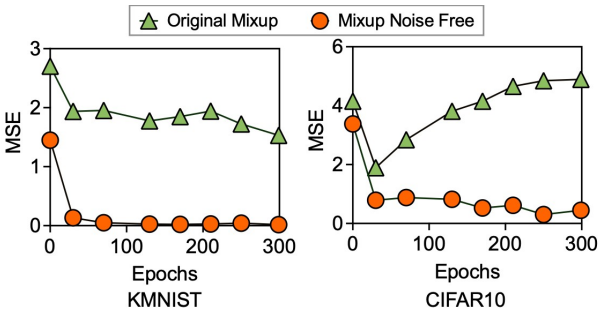
where ε_i and ε_j are the local noise errors defined in equation 3, and satisfy $\varepsilon(g) = \frac{1}{N} \sum_{i=1}^{N} \varepsilon_i$. Thus, the Mixup risk $\mathcal{R}'(g; \ell)$ consists of two classification-error terms weighted by λ and $(1-\lambda)$, plus the corresponding contributions from the local label-noise errors of the samples participating in the Mixup pair.

[1] Here, $[\![\cdot]\!]$ denotes the indicator function: for any condition A, $[\![A]\!] = 1$ if A holds and 0 otherwise.

Proof. Refer to Appendix A for the proof. □



(a) Relationship between noise ratio and model performance on CIFAR-10 with the SCL-NL loss and ResNet18 when applying original Mixup, Extra-Class Mixup Noise-Free (NF) and Intra-Class Mixup NF Increasing the noise ratio in original Mixup degrades model performance. Further ablation study reveals that same class Intra-Class Mixup NF can be more beneficial.



(b) Comparison of gradient estimation errors between original Mixup and *Mixup Noise-Free* on MNIST and CIFAR10, using the SCL-NL loss function and ResNet18 architecture. *Mixup Noise-Free* demonstrates lower gradient estimation error than the original Mixup on both datasets, attributed to reduced noise interference, which impacts classifier performance in CLL contexts.

Figure 2: Analysis of the impact of noise and Mixup Noise-Free (NF) on complementary-label learning performance.

The equation 5 emphasizes that minimizing complementary loss under Mixup requires careful control of the noise rate in the generated data. Specifically, it is critical to minimize instances where $\bar{y}_i = y_j \vee \bar{y}_j = y_i$, as such occurrences increase the error term ε in equation 5.

To further explore this effect, we evaluate gradient estimation error under noisy and noise-free setups. Let f denote the true gradient and c the complementary gradient estimator. The estimation error is defined as $\mathbb{E}_{(\mathbf{x},y,\bar{y})}[(f-c)^2]$. As shown in Figure 2b for MNIST and CIFAR10, the error associated with *Mixup Noise-Free* is consistently lower than that of original Mixup, reinforcing that label noise compromises optimization effectiveness.

Beyond quantifying noise, we investigate whether the original Mixup strategy used in ordinary learning transfers effectively to the CLL setting. In traditional supervised learning, original Mixup interpolates inputs and labels from different classes, which encourages smooth decision boundaries. In contrast, CLL imposes constraints that make such cross-class interpolation problematic. We hypothesize that mixing data within the same class, termed Intra-Class Mixup, preserves the CLL constraint $\bar{y}_i \in [K] \setminus y_i$ and reduces noise. To evaluate this, we synthetically generate intra-class and extra-class samples under a noise-free setup. Results in Figure 2a (highlighted by the *blue square* and *orange circle*) reveal that *Intra-Class Mixup Noise-Free*[2] outperforms *Extra-Class Mixup Noise-Free*[3] by 11%. This significant margin validates our hypothesis: intra-class mixing is more suitable in the CLL context and significantly improves performance.

The experimental results indicate that original Mixup can be effective under noise-free conditions, eliminating noise typically requires access to ordinary labels which contradicting the fundamental assumption of CLL. To address this challenge, we propose a dedicated framework for CLL, referred to as ICM. This framework is specifically designed to reduce synthetic label noise without requiring knowledge of the ordinary label. The following subsection provides a detailed explanation of the ICM framework.

3.2 Intra-Cluster Mixup in CLL

As discussed in above subsection, we introduce a specialized design for CLL called ICM, illustrated in Figure 1. Our proposed methodology, ICM, comprises two primary components. First, feature representations

[2]Intra-Class Mixup Noise-Free: is a proof-of-concept variant we designed prior to our proposed method. In this setting, Mixup is applied only between samples from the same class, so no additional label noise is introduced.

[3]Extra-Class Mixup Noise-Free: Mixup is applied between samples cross different class.

are extracted from the training data using a self-supervised learning model based on SimSiam (Chen & He, 2021). These embeddings are then clustered using k-means to group samples with similar feature characteristics, as shown in the *top* of Figure 1. This clustering step assigns cluster-based labels to sample and serves as a pre-processing phase. Second, synthetic complementary samples are generated by mixing inputs and labels within the same cluster, as illustrated in the *bottom right* of Figure 1. These augmented samples are then used to train the classifier. The procedure is defined as:

$$\tilde{\mathbf{x}}_{i,j} = \lambda \mathbf{x}_i + (1-\lambda)\mathbf{x}_j \tag{6}$$

$$\tilde{y}_{i,j} = \lambda \bar{y}_i + (1-\lambda)\bar{y}_j. \tag{7}$$

Integration of ICM into the training process is detailed in Algorithm 1. After clustering the dataset $\bar{D}$, ICM selects pairs within the same cluster to generate synthetic complementary samples using equation 6 and equation 7. Here, λ is sampled from a beta distribution $\beta(\alpha, \alpha)$, and the selected pairs $(\mathbf{x}_i, \bar{y}_i)$ and $(\mathbf{x}_j, \bar{y}_j)$ are drawn uniformly from the training data.

Algorithm 1: ICM training with cluster-consistent Mixup. Lines 1–3: extract SimSiam embeddings and assign k clusters. Lines 4–12: synthesize $(\tilde{\mathbf{x}}, \tilde{y})$ by interpolating pairs within the same cluster using Eq. (6)–(7). Lines 13–14: update θ on the synthetic batch.

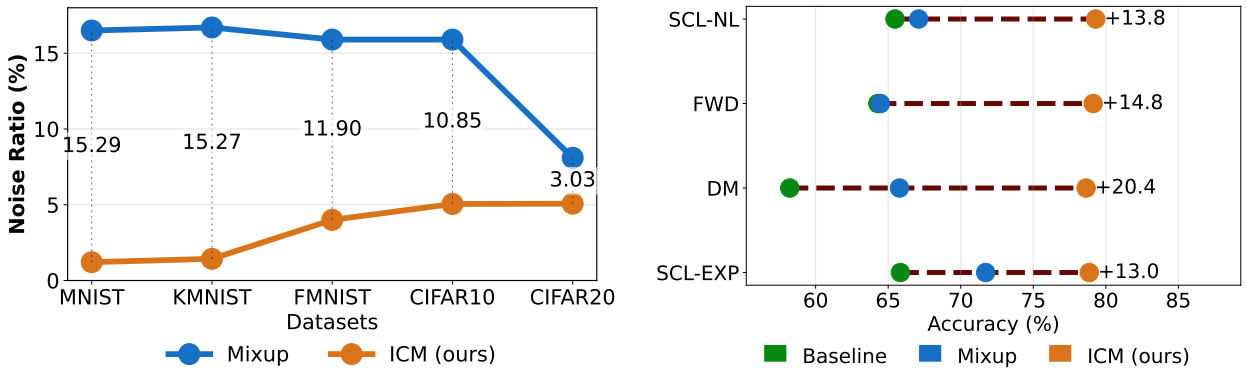
Input: Complementary-labeled dataset $\bar{\mathcal{D}} = \{(\mathbf{x}_i, \bar{y}_i)\}_{i=1}^N$, model f_θ.
Output: Trained parameters θ.

1 **(1)** Embedding: $\mathbf{z}_i \leftarrow \mathcal{F}_{\text{sim}}(\mathbf{x}_i)$ for $i = 1, \ldots, N$ `// $\mathcal{F}_{\text{sim}}$: pretrained SimSiam encoder`
2 **(2)** Clustering: run k-means on $\{\mathbf{z}_i\}$ to obtain cluster labels $c_i \in \{1, \ldots, k\}$.
3 **(3)** Augment data: $\bar{\mathcal{D}} \leftarrow \{(\mathbf{x}_i, \bar{y}_i, c_i)\}_{i=1}^N$.
4 **while** *not converged* **do**
5 **(4)** Sample a minibatch $\mathcal{B} \subset \bar{\mathcal{D}}$.
6 **(5)** For each cluster $u \in \{1, \ldots, k\}$, form $\mathcal{B}_u = \{(\mathbf{x}, \bar{y}, c) \in \mathcal{B} : c = u\}$.
7 **(6)** Initialize synthetic set $\tilde{\mathcal{B}} \leftarrow \emptyset$.
8 **foreach** u *with* $|\mathcal{B}_u| \geq 2$ **do**
9 **for** $m = 1$ **to** M_u **do**
10 (a) Draw two distinct pairs $(\mathbf{x}_i, \bar{y}_i, u), (\mathbf{x}_j, \bar{y}_j, u) \in \mathcal{B}_u$. `// sampling the samples in same cluster.`
11 (b) Sample $\lambda \sim \text{Beta}(\alpha, \alpha)$.
12 (c) Obtain ICM input $\tilde{\mathbf{x}}$ using Eq. (6).
13 (d) Compute label mixing coefficient $\lambda_{\bar{y}}$.
14 (e) Generate ICM label $\tilde{y}$ using Eq. (7).
15 (f) Append $(\tilde{\mathbf{x}}, \tilde{y})$ to $\tilde{\mathcal{B}}$.
16 **(7)** Compute loss: $\mathcal{L}(\theta) \leftarrow \frac{1}{|\tilde{\mathcal{B}}|} \sum_{(\tilde{\mathbf{x}}, \tilde{y}) \in \tilde{\mathcal{B}}} \mathcal{L}\big(g(\tilde{\mathbf{x}}), \tilde{y}\big)$. `// training model with new synthetic data.`
17 **(8)** Update: $\theta \leftarrow \theta - \eta \nabla_\theta \mathcal{L}(\theta)$.

Clustering plays a critical role by reducing label noise during data augmentation. Grouping samples within clusters encourages mixing between instances that are more likely to share the same true label. This increases the likelihood that the complementary label condition $\bar{y}_i \in [K] \setminus y_i$ holds across the cluster, thereby reducing the risk of introducing noise. To further investigate the effect of encoder choice in our method, we conduct an ablation study comparing SimSiam with other self-supervised encoders; detailed results are provided in Appendix D.7.

To evaluate ICM, we conduct an ablation study comparing it with original Mixup. As shown in Figure 3a, ICM significantly reduces the noise ratio. For instance, on MNIST, the ratio drops from 16.24% with Mixup to 0.95% with ICM. The effectiveness of noise reduction correlates with dataset complexity; simpler datasets such as MNIST, KMNIST, and FMNIST exhibit greater improvements than more complex datasets like CIFAR10 and CIFAR20. This reduction in noise ratio is mirrored by substantial performance improvements. As shown in Figure 3b, ICM consistently outperforms the original Mixup across all algorithms, with gains

ranging from 13% to 20%. These results validate the benefit of incorporating clustering into original Mixup for reducing label noise in CLL.



(a) Noise ratios of the Mixup and ICM methods across five datasets.

(b) Test accuracy of Mixup and ICM across different algorithms on CIFAR10

Figure 3: Comparison of the noise ratio across datasets (**left**) and the test accuracy of Mixup and ICM for different algorithms on CIFAR10 (**right**).

Surrogate Complementary Loss with ICM We propose a data augmentation for the existing loss-based CLL algorithms. In CLL, to minimize the non-convex $(0-1)$ loss in complementary learning, a common approach in statistical learning is to use a convex surrogate loss to approximate the target loss. In our work, we use ℓ to denote the *surrogate complementary loss (SCL)* loss functions and we combine our proposed data augmentation technique ICM with SCL during the training process. The main idea behind ICM data augmentation for complementary labels is to assign a new complementary label and incorporate additional information from samples within the same cluster, which share the same ordinary label. This approach enables the model to access not only more complementary labels but also new information about the samples. Additionally, it helps to reduce the noise that may arise when selecting pairs for data augmentation during training, thereby improving the overall learning process.

In loss-based complementary learning algorithms, a loss function $\ell\colon [K] \times \mathbb{R}^K \to \mathbb{R}$ is employed, which takes as input both the complementary label $\bar{y}_i$ and the prediction output of the model $g(\mathbf{x}_i)$. The objective of learning process is to minimize this loss function ℓ over the complementary dataset $\bar{D}$, which can be formulated as:

$$\mathcal{L}(g;\ell) = \frac{1}{N}\sum_{i=1}^{N} \ell\Big(\bar{y}_i, g(\mathbf{x}_i)\Big). \tag{8}$$

When incorporating the ICM data augmentation during training, the CLL loss function is updated as follows:

$$\mathcal{L}'(g;\ell) = \frac{1}{N}\sum_{i=1}^{N} \ell\Big(\tilde{y}_{i,j}, g(\tilde{\mathbf{x}}_{i,j})\Big) = \frac{1}{N}\sum_{i=1}^{N} \Big[\lambda\ell\Big(\bar{y}_i, g(\tilde{\mathbf{x}}_{i,j})\Big) + (1-\lambda)\ell\Big(\bar{y}_j, g(\tilde{\mathbf{x}}_{i,j})\Big)\Big], \tag{9}$$

where $\lambda \in [0,1] \sim \beta(\alpha,\alpha)$ (*beta distribution*), for $\alpha \in (0,\infty)$, $\tilde{\mathbf{x}}_{i,j}$ in equation 6, $\tilde{y}_{i,j}$ in equation 7, j is random sampling from the same cluster of i, and N is the size of training dataset. To better distinguish from $0-1$ based methods, we use a convex surrogate loss to approximate the target loss, denotes ℓ. In fact, previous research in complementary learning has revealed similar patterns focused on minimizing the predictions of label classes, including approaches such as:

- Negative learning loss (SCL-NL) in (Kim et al., 2019) a modified log loss specifically designed for negative learning with complementary labels:

$$\ell_{\mathrm{NL}}\big(\bar{y}, g(\mathbf{x})\big) = -\log(1 - \mathbf{p}_{\bar{y}} + \gamma), \text{where } 0 < \gamma < 1. \tag{10}$$

- Exponential loss (SCL-EXP) (Chou et al., 2020b):
$$\ell_{\mathrm{EXP}}\big(\bar{y}, g(\mathbf{x})\big) = \exp(\mathbf{p}_{\bar{y}}). \tag{11}$$
- Forward correction (FWD) in (Yu et al., 2018) is a method for correcting loss using a forward correction approach based on a given transition matrix T:
$$\ell_{\mathrm{FWD}}\big(\bar{y}, g(\mathbf{x})\big) = \ell(\bar{y}, T^T\mathbf{p}). \tag{12}$$

Here, γ is a constant added to the loss function to prevent the SCL-NL loss from approaching infinity when $\mathbf{p}_{\bar{y}}$ equals 1, $\mathbf{p} \in \Delta^{K-1}$ represents the probability output of learning model if g passes through a softmax layer, and Δ^{K-1} is the K-dimensional simplex.

4 Experiments

In this section, we evaluate ICM on synthetic and real-world datasets under both balanced and imbalanced conditions, comparing it with state-of-the-art CLL baselines. Our findings demonstrate that ICM significantly enhances performance and effectively addresses key CLL challenges.

4.1 Experiment Setup

Datasets. We assess the effectiveness of our proposed ICM framework across five synthetic labeled datasets: CIFAR10, CIFAR20, MNIST, KMNIST, and FMNIST. The synthetic labeled datasets consist of CIFAR10 (Krizhevsky et al., 2009) and CIFAR20, each containing 50,000 training samples and 10,000 testing samples. CIFAR10 encompasses 10 classes, whereas CIFAR20 comprises 20 superclasses derived from CIFAR100 (Krizhevsky et al., 2009). We do not benchmark on CIFAR100, as existing CLL algorithms have not demonstrated the ability to learn a meaningful classifier on this dataset when given only one complementary label per data instance. MNIST (Lecun et al., 1998), KMNIST (Clanuwat et al., 2018), and FMNIST (Xiao et al., 2017) each consist of 60,000 training samples and 10,000 testing samples, with all three datasets featuring ten classes.

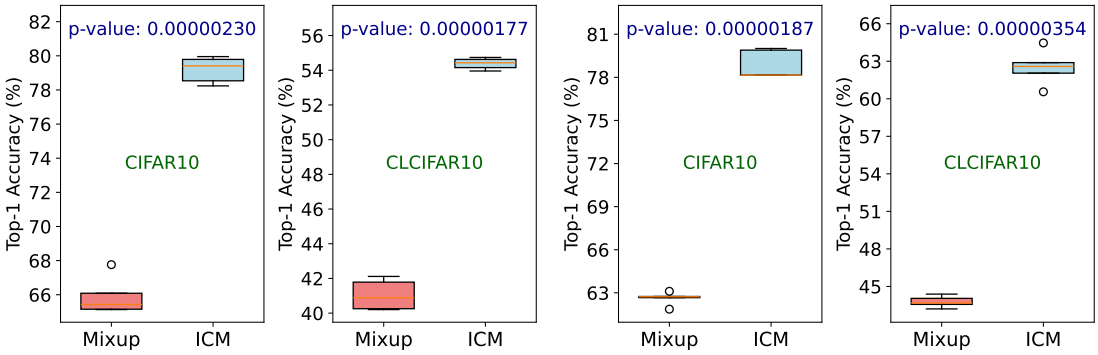
Additionally, we evaluate our framework on real-world labeled datasets, including CLCIFAR10 and CLCIFAR20 (Wang et al., 2025), which use the images from CIFAR10 and CIFAR20, respectively, with complementary labels annotated by humans. In CLL, MNIST and CIFAR are standard datasets. Researchers have not transitioned to large-scale datasets with numerous classes, such as TinyImageNet (Le & Yang, 2015) and ImageNet (Deng et al., 2009). Preliminary tests reveal that state-of-the-art CLL algorithms struggle to produce meaningful classifiers for 100 classes, even with uniformly and noiselessly generated synthetic complementary labels. This is why existing CLL algorithms are evaluated on datasets with 10, 20 classes.

Baseline Methods. Our framework can be applied with different methods, we choose SCL-EXP, SCL-NL (Chou et al., 2020b), FWD-INT (Yu et al., 2018), DM (Gao & Zhang, 2021) as our cooperators to validate the efficacy of our approach.

Implementation Details. For a fair comparison, we choose RestNet18 (He et al., 2016) as our backbone. We train the model with a batch size 512 for 300 epochs and an initial learning rate of 10^{-4}, weight decay 10^{-4}, and optimizer Adam. In the long-tailed imbalance setting (Cui et al., 2019; Cao et al., 2019), the difficulty of a dataset is commonly characterized by the *class imbalance ratio*, defined as $\rho = \frac{\max_i n_i}{\min_i n_i}$, where n_i denotes the number of samples in class i. A dataset is said to exhibit *long-tailed imbalance* with ratio ρ when the class sizes follow an exponentially decreasing sequence whose common ratio is $\rho^{1/(K-1)}$ across the K classes. This construction ensures that the ratio between the largest (head) class and the smallest (tail) class is exactly ρ. All the experiments were run with Tesla V100-SXM2, 32GB memories. The hyper-parameters can be appropriately tuned via the validation process. For each subtask, we run the experiments three times. Other implemented details, including hyper-parameter selection through validation process such as α, K cluster, can be found in our supplementary materials Appendix D. We experiment our proposed method across a wide spectrum of both balanced and imbalanced CLL settings. For imbalanced CLL, we follow (Cao et al., 2019) to generate a long-tailed distribution dataset with different imbalance ratios on ordinary datasets. Details of the different imbalanced setups can be found in the Appendix B.

Table 1: Top-1 validation accuracy (%) on balanced (*bal*) $\rho = 1$ and long-tailed imbalanced (*imb*) ratio $\rho = 10$, K cluster $= 50$ setups. The methods used are *SCL-NL* (S-NL), *FWD-INT* (FWD), *SCL-EXP* (S-EXP), *DM* losses, and ResNet18. Best performance is *highlighted in bold*.

	Imbalanced *(imb)*		Balanced *(bal)*	
Method	CLCIFAR10	CLCIFAR20	CLCIFAR10	CLCIFAR20
S-NL	$17.77_{0.20}$	$5.80_{0.03}$	$37.59_{0.40}$	$8.53_{0.24}$
S-NL+Mix	$21.28_{0.51}$	$6.64_{0.48}$	$42.96_{0.54}$	$9.13_{0.44}$
S-NL+ICM (ours)	$\mathbf{28.44}_{0.05}$	$\mathbf{7.55}_{0.08}$	$\mathbf{56.63}_{0.61}$	$\mathbf{11.26}_{0.24}$
DM	$15.19_{0.15}$	$5.76_{0.06}$	$38.20_{0.68}$	$8.34_{0.08}$
DM+Mix	$22.99_{0.22}$	$6.92_{0.21}$	$42.61_{0.48}$	$9.12_{0.32}$
DM+ICM (ours)	$\mathbf{27.88}_{0.94}$	$\mathbf{7.10}_{0.02}$	$\mathbf{53.04}_{0.40}$	$\mathbf{11.47}_{0.18}$
FWD	$12.07_{0.01}$	$5.98_{0.17}$	$42.98_{0.36}$	$21.10_{0.23}$
FWD+Mix	$17.06_{0.89}$	$6.10_{0.16}$	$42.38_{0.05}$	$21.48_{0.19}$
FWD+ICM (ours)	$\mathbf{18.23}_{0.08}$	$\mathbf{7.73}_{0.09}$	$\mathbf{58.97}_{0.21}$	$\mathbf{35.94}_{0.33}$
S-EXP	$17.37_{0.16}$	$5.99_{0.21}$	$41.42_{0.68}$	$8.56_{0.25}$
S-EXP+Mix	$20.38_{0.40}$	$6.84_{0.13}$	$43.56_{0.13}$	$9.04_{0.21}$
S-EXP+ICM (ours)	$\mathbf{27.52}_{0.06}$	$\mathbf{7.01}_{0.08}$	$\mathbf{56.26}_{0.15}$	$\mathbf{11.20}_{0.06}$



(a) S-NL algorithm.

(b) FWD algorithm.

Figure 4: Comparing the p-value of different between Mixup and ICM method on CIFAR10 and CLCIFAR10 with S-NL (**right**) and FWD (**left**) algorithms on both balanced and imbalanced ($\rho = 100$) scenarios.

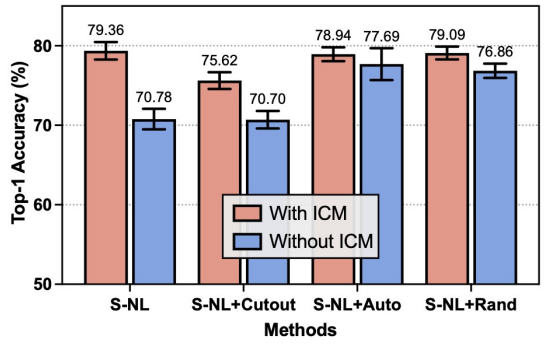
4.2 Results and Analysis

We compare our results with several baselines, including *without Mixup* and *with Mixup (Mix)*, to verify the efficacy of our proposed method. Additionally, our method can be integrated with various base algorithms such as SCL-NL, FWD-INT, DM, and SCL-EXP.

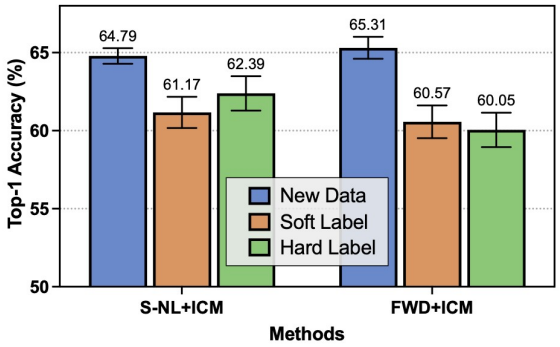
The results for the real-world labeled datasets, CLCIFAR10 and CLCIFAR20, both in balanced and imbalanced settings with different loss functions, are presented in Table 1. For the synthetic labeled datasets (CIFAR10, CIFAR20, MNIST, KMNIST, and FMNIST) with *setup 1*, spanning from balanced to various imbalance ratios, detailed results are shown in Tables 2. Additional experimental details for *setup 2* and *setup 3*, with varying imbalanced ratios, can be found in Appendix B. Our proposed method consistently outperforms the baselines across all setups, from balanced to imbalanced scenarios, and achieves significant performance improvements when integrated with different base algorithms. To assess whether the observed improvements are statistically significant, we compute p-values on the CIFAR-10 and CLCIFAR-10 datasets. The results show that the p-values are well below 0.001, providing strong evidence that our proposed method significantly outperforms the original Mixup. These results are summarized in Figure 4 and reported in more detail in Appendix D.5.

Table 2: Top-1 validation accuracy (%) for $\rho = 100$ (long-tailed imbalanced) and $\rho = 1$ (balanced) setups across CIFAR10, CIFAR20, MNIST, KMNIST, and FMNIST datasets using ResNet18 and different loss methods.

Method	CIFAR10		CIFAR20		MNIST		KMNIST		FMNIST	
	$\rho = 100$	$\rho = 1$	$\rho = 100$	$\rho = 1$	$\rho = 100$	$\rho = 1$	$\rho = 100$	$\rho = 1$	$\rho = 100$	$\rho = 1$
S-NL	$22.41_{0.31}$	$65.47_{0.05}$	$10.65_{0.28}$	$24.14_{0.33}$	$50.15_{0.52}$	$97.78_{0.21}$	$35.17_{0.17}$	$88.92_{0.14}$	$51.93_{0.33}$	$85.15_{0.16}$
S-NL+Mix	$31.46_{0.59}$	$67.10_{0.11}$	$13.47_{0.46}$	$26.45_{0.07}$	$54.23_{0.48}$	$96.64_{0.11}$	$37.25_{0.53}$	$79.04_{0.16}$	$56.07_{0.57}$	$84.35_{0.08}$
S-NL+ICM	$\mathbf{36.21}_{0.19}$	$\mathbf{79.13}_{0.04}$	$\mathbf{18.11}_{0.31}$	$\mathbf{39.17}_{0.15}$	$\mathbf{85.83}_{0.19}$	$\mathbf{98.20}_{0.10}$	$\mathbf{63.70}_{0.34}$	$\mathbf{89.09}_{0.08}$	$\mathbf{67.80}_{0.40}$	$\mathbf{85.25}_{0.99}$
FWD	$22.20_{0.40}$	$64.29_{0.33}$	$8.43_{0.29}$	$23.18_{0.34}$	$50.26_{0.57}$	$97.49_{0.08}$	$35.29_{0.37}$	$80.41_{0.36}$	$51.89_{0.63}$	$84.16_{0.07}$
FWD+Mix	$29.03_{0.39}$	$64.47_{0.26}$	$14.55_{0.46}$	$22.79_{0.06}$	$52.44_{0.49}$	$94.09_{0.37}$	$37.77_{0.47}$	$70.83_{0.11}$	$53.24_{0.48}$	$82.43_{0.59}$
FWD+ICM	$\mathbf{39.71}_{0.29}$	$\mathbf{79.22}_{0.03}$	$\mathbf{21.76}_{0.26}$	$\mathbf{42.20}_{0.09}$	$\mathbf{85.27}_{0.69}$	$\mathbf{98.18}_{0.10}$	$\mathbf{63.50}_{0.47}$	$\mathbf{88.92}_{0.04}$	$\mathbf{67.90}_{0.47}$	$\mathbf{84.75}_{0.11}$
DM	$20.91_{0.15}$	$58.22_{0.24}$	$10.16_{0.05}$	$21.43_{0.09}$	$51.28_{0.33}$	$95.10_{0.05}$	$32.60_{0.20}$	$73.98_{0.25}$	$49.46_{0.08}$	$82.68_{0.34}$
DM+Mix	$30.52_{0.31}$	$65.78_{0.22}$	$13.27_{0.12}$	$24.95_{0.66}$	$55.78_{2.36}$	$95.69_{0.10}$	$36.37_{1.19}$	$78.15_{0.35}$	$54.22_{0.56}$	$82.77_{0.37}$
DM+ICM	$\mathbf{36.37}_{0.17}$	$\mathbf{78.64}_{0.05}$	$\mathbf{17.42}_{0.52}$	$\mathbf{38.48}_{0.13}$	$\mathbf{85.91}_{0.15}$	$\mathbf{98.67}_{0.07}$	$\mathbf{66.98}_{4.79}$	$\mathbf{89.60}_{0.52}$	$\mathbf{66.46}_{1.19}$	$\mathbf{84.61}_{0.24}$
S-EXP	$22.96_{0.24}$	$65.84_{0.23}$	$10.13_{0.29}$	$24.07_{0.10}$	$50.29_{0.06}$	$98.68_{0.21}$	$35.62_{0.05}$	$90.38_{0.08}$	$51.30_{0.10}$	$85.23_{0.09}$
S-EXP+Mix	$31.51_{0.18}$	$71.72_{0.29}$	$13.77_{0.17}$	$27.90_{0.25}$	$53.68_{0.28}$	$98.27_{0.12}$	$37.52_{1.02}$	$88.40_{0.23}$	$53.50_{3.53}$	$84.42_{0.21}$
S-EXP+ICM	$\mathbf{36.70}_{0.10}$	$\mathbf{78.86}_{0.12}$	$\mathbf{16.75}_{0.15}$	$\mathbf{38.91}_{0.22}$	$\mathbf{86.06}_{0.37}$	$\mathbf{98.81}_{0.03}$	$\mathbf{64.46}_{0.04}$	$\mathbf{90.45}_{0.12}$	$\mathbf{64.03}_{3.31}$	$\mathbf{85.73}_{0.20}$



(a) Enhancing robustness by combining ICM with weak and strong data augmentation techniques on CIFAR10.



(b) Comparison of new, soft, and hard label sharing strategies under S-NL and FWD losses on CIFAR10.

Figure 5: Experimental results on CIFAR-10: (a) robustness gains from combining ICM with weak and strong data augmentations; (b) performance comparison of new, soft, and hard label-sharing strategies under S-NL and FWD losses.

Moreover, we conduct another analyses demonstrating that our proposed method, ICM, proves to be a competitive approach for enhancing CLL. This is evidenced by our assessment of the enhancing robustness of ICM when combining with various data augmentation techniques, ranging from weak (Flipflop, Cutout (DeVries & Taylor, 2017)) to strong (AutoAug (Cubuk et al., 2019), RandAug (Cubuk et al., 2020)). The results in Figure 5a illustrate the significant benefits of combining ICM with various data augmentation techniques, for instance, on the CIFAR10 dataset, the combination of ICM with these augmentations achieves accuracy levels approaching 80%, far surpassing the results of their counterparts without ICM. Interestingly, Cutout appears to hurt performance when used together with ICM. A plausible explanation is that applying ICM on top of Cutout may excessively remove informative regions of the input, leading to overly distorted synthetic samples.

Furthermore, we conduct a series of analyses demonstrating that our proposed method, ICM, proves to be a competitive approach for enhancing CLL. This is evidenced by our assessment of the *bias* and *variance* of the empirical *Gradient Analysis* in next section. It is also crucial to highlight that the benefits of sharing new synthetic data extend beyond merely sharing complementary labels in the CLL context. This assertion is supported by an ablation study where we share *new data*, *soft label*, and *hard label* during the model training process. The detailed results presented in Figure 5b. In addition, we investigate methods for mitigating class imbalance in CLL. We introduce *Multi Intra-Cluster Mixup (MICM)*, which extends intra-

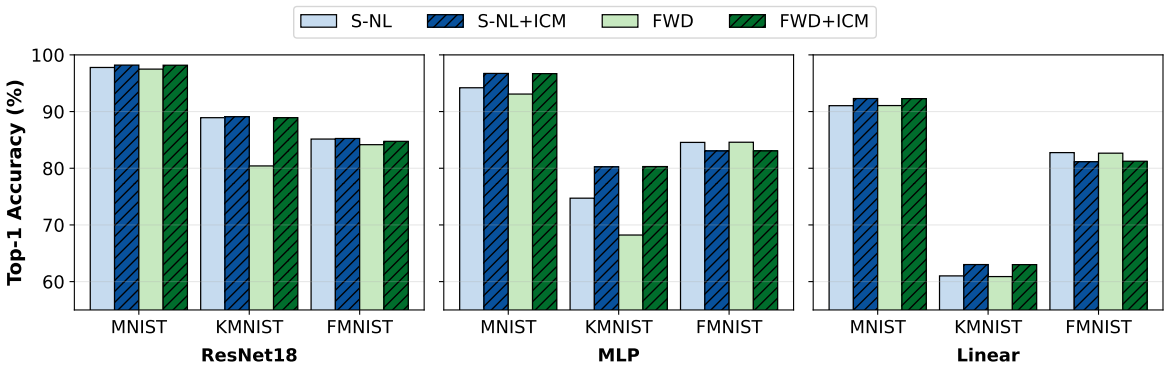


Figure 6: Comparing different model architecture with ICM method on MNIST family dataset under balanced setup.

cluster mixing to generate synthetic samples under imbalanced class distributions, thereby encouraging more effective complementary-label sharing for minority classes. Technical details of MICM and additional empirical results are provided in Appendices C and E.

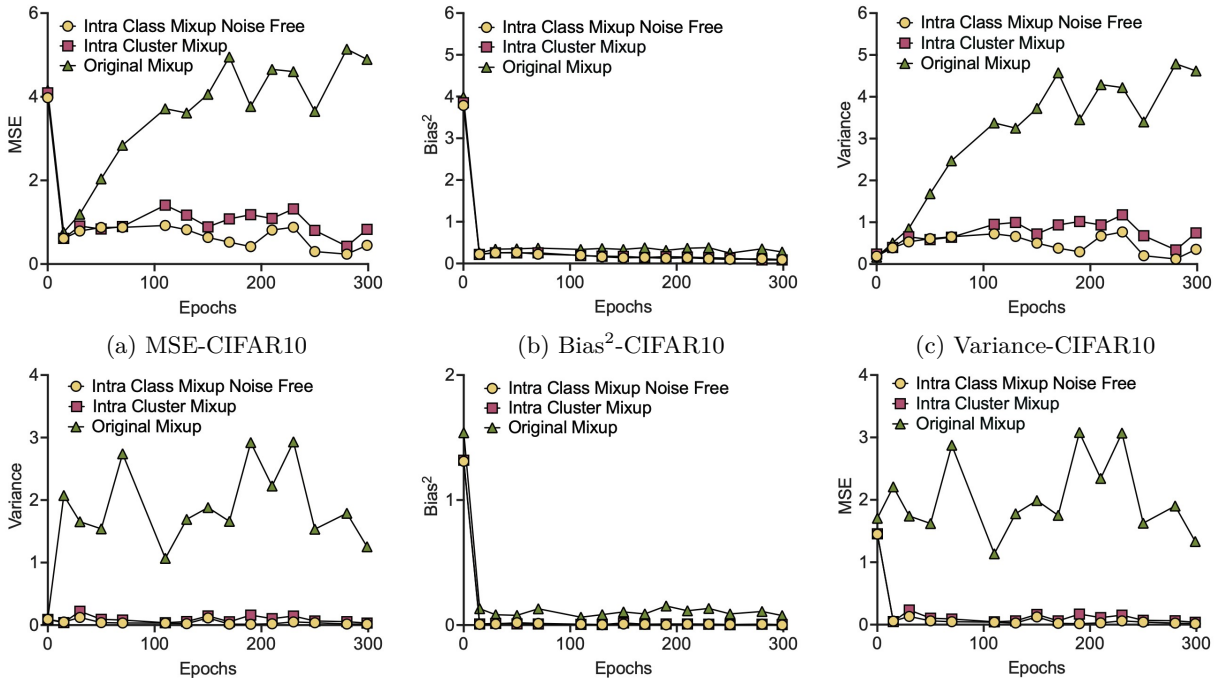
Additionally, we evaluate the effectiveness of our proposed method (ICM) across models of varying complexity, including linear classifiers, multilayer perceptrons (MLPs), and ResNet18, to examine how architectural capacity interacts with ICM. Our empirical results show that ICM consistently improves performance with ResNet18 on all three datasets (MNIST, KMNIST, and FMNIST). For MLPs and linear models, ICM yields clear gains on MNIST and KMNIST, but leads to degraded performance on FMNIST. Detailed results are reported in Figure 6 and Appendix D.6.

Taken together, these analyses motivate a broader perspective on the practicality of CLL. In recent years, the field has observed that CLL is still not fully practical, especially when moving from synthetic to real-world datasets and increasing the number of classes. We found that the current state-of-the-art algorithms struggle under these conditions, highlighting the need for further work to make CLL more applicable in practice (Wang et al., 2025; Ye et al., 2024). Our proposed method, ICM, introduces a novel data augmentation approach that aims to make CLL more realistic. Specifically, ICM is designed to mitigate the effects of complementary-label noise associated with synthetic complementary samples. Through rigorous empirical evaluation, ICM demonstrates the effectiveness of encouraging complementary-label sharing among nearby examples, leading to consistent performance improvements across a wide range of experimental setups. Our empirical results further show that ICM substantially improves the performance of learning models across various state-of-the-art algorithms. In particular, when we applied ICM to a real-world dataset, CLCIFAR10, the model performance increased by 10%. We hope that our work helps practitioners develop more accurate and reliable models in real-world scenarios characterized by complementary-label learning.

4.3 Gradient Analysis

We further discuss how the ICM framework gives such improvement by arranging the learning process via gradient analysis. This discussion centers on examining loss gradients within the experimental setup, particularly the *stochastic gradient (SGD)* employed in mini-batch optimization. Specifically, we evaluate the bias-variance tradeoff of the gradient estimation error involving complementary gradients with ICM and the Mixup method versus the ordinary gradient. To provide a more accurate assessment, we utilize the bias-variance decomposition technique. Traditionally used in statistical learning to assess algorithmic complexity, we extend this framework to evaluate the estimation error of the gradient, setting the ordinary gradient as the target. We will show that our proposed framework ICM has a lower *mean squared error (MSE)* than the original Mixup, caused by its slight variance and bias.

We represent the gradient step determined by ordinary labeled data $(\mathbf{x}, y)$ and ordinary loss ℓ as f. The complementary gradient step, considering complementary labeled data $(\mathbf{x}, \bar{y})$ and complementary loss $\bar{\ell}$ or (ϕ), is denoted as c. Additionally, b denotes the expected gradient step of $[K]\backslash\{y\}$, calculated as the average



(a) MSE-CIFAR10 (b) Bias^2-CIFAR10 (c) Variance-CIFAR10

(d) MSE-KMNIST (e) Bias^2-KMNIST (f) Variance-KMNIST

Figure 7: Comparison of gradient estimation errors between original Mixup and *Mixup Noise-Free* on MNIST and CIFAR10, using the SCL-NL loss function and ResNet18 architecture. *Mixup Noise-Free* demonstrates lower gradient estimation error than the original Mixup on both datasets, attributed to reduced noise interference, which impacts classifier performance in CLL contexts.

of c across all possible complementary labels. This can be formalized as follows:

$$f = \nabla \ell\big(y, g(\mathbf{x})\big), \tag{13}$$

$$c = \nabla \bar{\ell}\big(\bar{y}, g(\mathbf{x})\big), \tag{14}$$

$$b = \frac{1}{K-1} \sum_{y' \neq \bar{y}} \nabla \bar{\ell}\big(y', g(\mathbf{x})\big). \tag{15}$$

We designate f as the ground truth, representing the target complementary estimator c. We expect the MSE of the gradient estimation to be minimal.

$$\text{MSE} = \mathbb{E}_{\mathbf{x},y,\bar{y}}[(f-c)^2]. \tag{16}$$

We drive the bias-variance decomposition by introducing b and eliminating remaining terms:

$$\mathbb{E}\big[(f-c)^2\big] = \mathbb{E}[(f-b+b-c)^2], \tag{17}$$

$$= \underbrace{\mathbb{E}[(f-b)^2]}_{\texttt{Bias}^2} + \underbrace{\mathbb{E}[(b-c)^2]}_{\texttt{Variance}}. \tag{18}$$

We conduct experiments to assess how well the complementary gradient c approximates the ordinary gradient f and compare it with a baseline method (*original Mixup*). The training process is as follows:

In each epoch, we compute three gradients, namely the ordinary gradient f, the current method c, and b. We evaluate the MSE, the square bias term, and the variance term using equation 16 and equation 18. In each epoch, we update the model only with f to ensure a fair comparison of gradients. The optimizer used was SGD with a learning rate of 10^{-4}, and the training was conducted for 300 epochs.

The results presented in Figure 7 indicate that Mixup exhibits a higher MSE due to elevated levels of variance and bias. Conversely, ICM demonstrates significantly lower variance and bias when compared to Mixup, aligning more closely with the ideal case of *Intra Class Mixup Noise Free*. This supports our observation that our proposed method outperforms Mixup in CLL context by achieving lower variance and bias.

5 Conclusion

This paper presented a novel data augmentation approach, ICM, specifically designs to mitigate the effects of *complementary-label noise* associated with synthetic complementary samples by synthesizing augmented data only within the same cluster. Through rigorous empirical evaluations across diverse CLL settings, we have demonstrated the effectiveness of encouraging complementary label sharing of nearby examples, leading to consistent performance improvements across a wide spectrum of experimental setups, from synthetic to real-world labeled datasets, in both balanced and imbalanced CLL settings. Our empirical experiments reveal that ICM substantially enhances the performance of learning models across a variety of state-of-the-art algorithms. Additionally, our investigations highlight the heightened sensitivity of classifiers trained under CLL conditions to *complementary-label noise*, which leads to performance degradation of CLL models. These findings underscore the significant contribution of ICM to the field of CLL. By providing a data augmentation strategy that effectively tackles the issue of *complementary-label noise*, ICM empowers practitioners to develop more accurate and reliable models in real-world scenarios characterized by CLL.

6 Limitation and Future Works

Despite its contributions, this study has several limitations. First, when applied to simpler models such as linear classifiers and multilayer perceptrons (MLPs) on benchmark FMIST datasets, our augmentation technique yields reduced performance. This decline stems from the limited capacity of these models, which struggle to accommodate the added complexity and increased overlap in feature representations introduced by ICM. Second, we have not yet evaluated our approach in scenarios where instances carry multiple complementary labels. Investigating the benefits of ICM in a multi-complementary-label learning setting remains an important direction for future work.

Broader Impact Statement

We used publicly available benchmarks, including MNIST, KMNIST, FMNIST, CIFAR10, CIFAR20, CLCIFAR10, and CLCIFAR20, and identified no significant ethical concerns in their use. To tackle the limited supervision inherent in these datasets, we optimized our learning algorithm to efficiently extract insights from both balanced and imbalanced class distributions. This approach is especially valuable in scientific settings where true labels are sensitive or costly to obtain. Moreover, by relying on complementary labels, we preserve data privacy and reduce annotation costs without sacrificing model accuracy.

References

Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arechiga, and Tengyu Ma. Learning imbalanced datasets with label-distribution-aware margin loss. In *Advances in neural information processing systems*, volume 32, 2019. 8, 17, 19

Yuzhou Cao, Shuqi Liu, and Yitian Xu. Multi-complementary and unlabeled learning for arbitrary losses and models. *Pattern Recognition*, 124:108447, 2022. 1, 2, 3, 17

Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15750–15758, 2021. 6

Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. In *IEEE/CVF international conference on computer vision*, pp. 9640–9649, 2021. 22

Hsin-Ping Chou, Shih-Chieh Chang, Jia-Yu Pan, Wei Wei, and Da-Cheng Juan. Remix: rebalanced mixup. In *European Conference on Computer Vision*, pp. 95–110, 2020a. 2, 17

Yu-Ting Chou, Gang Niu, Hsuan-Tien Lin, and Masashi Sugiyama. Unbiased risk estimators can mislead: A case study of learning with complementary labels. In *International Conference on Machine Learning*, pp. 1929–1938, 2020b. 1, 2, 3, 8

Tarin Clanuwat, Mikel Bober-Irizar, Asanobu Kitamoto, Alex Lamb, Kazuaki Yamamoto, and David Ha. Deep learning for classical japanese literature, 2018. 8

Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. Autoaugment: Learning augmentation strategies from data. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 113–123, 2019. 10

Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 702–703, 2020. 10

Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-balanced loss based on effective number of samples. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9268–9277, 2019. 8, 17

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2009. 8

Terrance DeVries and Graham W. Taylor. Improved regularization of convolutional neural networks with cutout, 2017. URL https://arxiv.org/abs/1708.04552. 10

Lei Feng, Takuo Kaneko, Bo Han, Gang Niu, Bo An, and Masashi Sugiyama. Learning with multiple complementary labels. In *International Conference on Machine Learning*, pp. 3072–3081, 2020. 1, 2, 3, 17

Yi Gao and Min-Ling Zhang. Discriminative complementary-label learning with weighted loss. In *International Conference on Machine Learning*, pp. 3587–3597, 2021. 8

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016. 3, 8

Takashi Ishida, Gang Niu, Weihua Hu, and Masashi Sugiyama. Learning from complementary labels. In *Advances in Neural Information Processing Systems*, pp. 5639–5649, 2017. 1, 3

Takashi Ishida, Gang Niu, and Masashi Sugiyama. Binary classification from positive-confidence data. In *Advances in Neural Information Processing Systems*, volume 31, 2018. 1, 17

Takashi Ishida, Gang Niu, Aditya Menon, and Masashi Sugiyama. Complementary-label learning for arbitrary losses and models. In *International Conference on Machine Learning*, pp. 2971–2980, 2019. 3

Liming Jiang, Bo Dai, Wayne Wu, and Chen Change Loy. Deceive d: Adaptive pseudo augmentation for gan training with limited data. In *Advances in Neural Information Processing Systems*, pp. 21655–21667, 2021. 2

Rong Jin and Zoubin Ghahramani. Learning with multiple labels. In *Advances in Neural Information Processing Systems*, volume 15, 2002. 1

Yasuhiro Katsura and Masato Uchida. Bridging ordinary-label learning and complementary-label learning. In *Asian Conference on Machine Learning*, pp. 161–176, 2020. 3

Youngdong Kim, Junho Yim, Juseung Yun, and Junmo Kim. Nlnl: Negative learning for noisy labels. In *IEEE/CVF International Conference on Computer Vision*, pp. 101–110, 2019. 7

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. Computer Science University of Toronto, Canada, 2009. 8

Ya Le and Xuan Yang. Tiny imagenet visual recognition challenge. Report of CS231N: Deep Learning for Computer Vision Course, 2015. Stanford University. 8

Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86:2278–2324, 1998. 8

Zhixin Li and Yuheng Jia. Conmix: Contrastive mixup at representation level for long-tailed deep clustering. In *The Thirteenth International Conference on Learning Representations*, 2025. 2

Wei-I Lin and Hsuan-Tien Lin. Reduction from complementary-label learning to probability estimates. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pp. 469–481, 2023. winner of the best paper runner-up award. 2, 17

Agnieszka Mikołajczyk and Michał Grochowski. Data augmentation for improving deep learning in image classification problem. In *International Interdisciplinary PhD Workshop*, pp. 117–122, 2018. 2

Madeline Navarro and Santiago Segarra. Graphmad: Graph mixup for data augmentation using data-driven convex clustering. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5, 2023. 2

Giorgio Patrini, Alessandro Rozza, Aditya Menon, Richard Nock, and Lizhen Qu. Making deep neural networks robust to label noise: a loss correction approach. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1944–1952, 2017. 3

Sylvestre-Alvise Rebuffi, Sven Gowal, Dan Andrei Calian, Florian Stimberg, Olivia Wiles, and Timothy A Mann. Data augmentation can improve robustness. In *Advances in Neural Information Processing Systems*, pp. 29935–29948, 2021. 2

Jiawei Ren, Cunjun Yu, Xiao Ma, Haiyu Zhao, Shuai Yi, et al. Balanced meta-softmax for long-tailed visual recognition. In *Advances in Neural Information Processing Systems*, pp. 4175–4186, 2020. 17

Pierre H Richemond, Jean-Bastien Grill, Florent Altché, Corentin Tallec, Florian Strub, Andrew Brock, Samuel Smith, Soham De, Razvan Pascanu, Bilal Piot, et al. Byol works even without batch statistics. *arXiv preprint arXiv:2010.10241*, 2020. 22

Connor Shorten and Taghi M Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of Big Data*, 6:1–48, 2019. 2

Hsiu-Hsuan Wang, Mai Tan Ha, Nai-Xuan Ye, Wei-I Lin, and Hsuan-Tien Lin. CLImage: Human-annotated datasets for complementary-label learning. *Transactions on Machine Learning Research*, 2025. ISSN 2835-8856. 8, 11

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms, 2017. 8

Xiangjin Xie, Li Yangning, Wang Chen, Kai Ouyang, Zuotong Xie, and Hai-Tao Zheng. Global mixup: Eliminating ambiguity with clustering. In *AAAI Conference on Artificial Intelligence*, volume 37, pp. 13798–13806, 2023. 2

Nai-Xuan Ye, Tan-Ha Mai, Hsiu-Hsuan Wang, Wei-I Lin, and Hsuan-Tien Lin. libcll: an extendable python toolkit for complementary-label learning. *arXiv preprint arXiv:2411.12276*, 2024. 11

Xiyu Yu, Tongliang Liu, Mingming Gong, and Dacheng Tao. Learning with biased complementary labels. In *European Conference on Computer Vision*, pp. 68–83, 2018. 2, 3, 8, 17

Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*, 2018. 2

Zhi-Hua Zhou. A brief introduction to weakly supervised learning. *National Science Review*, 5:44–53, 08 2017. ISSN 2095-5138. 1