CDMAD: Class-Distribution-Mismatch-Aware Debiasing for Class-Imbalanced Semi-Supervised Learning

Hyuck Lee Heeyoung Kim
Department of Industrial and Systems Engineering, KAIST
Daejeon 34141, Republic of Korea
{dlgur0921, heeyoungkim}@kaist.ac.kr

Abstract

Pseudo-label-based semi-supervised learning (SSL) algorithms trained on a class-imbalanced set face two cascading challenges: 1) Classifiers tend to be biased towards majority classes, and 2) Biased pseudo-labels are used for training. It is difficult to appropriately re-balance the classifiers in SSL because the class distribution of an unlabeled set is often unknown and could be mismatched with that of a labeled set. We propose a novel class-imbalanced SSL algorithm called class-distribution-mismatch-aware debiasing (CDMAD). For each iteration of training, CDMAD first assesses the classifier's biased degree towards each class by calculating the logits on an image without any patterns (e.g., solid color image), which can be considered irrelevant to the training set. CDMAD then refines biased pseudo-labels of the base SSL algorithm by ensuring the classifier's neutrality. CDMAD uses these refined pseudo-labels during the training of the base SSL algorithm to improve the quality of the representations. In the test phase, CDMAD similarly refines biased class predictions on test samples. CDMAD can be seen as an extension of post-hoc logit adjustment to address a challenge of incorporating the unknown class distribution of the unlabeled set for re-balancing the biased classifier under class distribution mismatch. CDMAD ensures Fisher consistency for the balanced error. Extensive experiments verify the effectiveness of CDMAD.

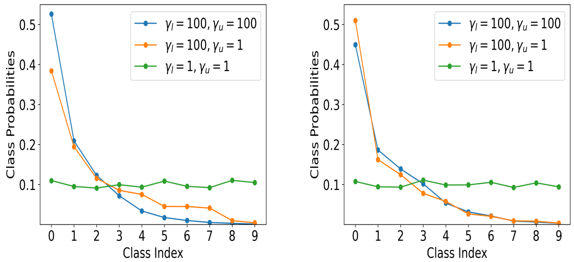
1. Introduction

Classifiers trained on a class-imbalanced set suffer from being biased toward the majority classes. Under semi-supervised learning (SSL) settings, classifiers of pseudo-label-based algorithms tend to be further biased because of the use of biased pseudo-labels for training. The use of biased pseudo-labels also decreases the quality of representations. This problem becomes more serious when the class distributions of the labeled and unlabeled sets differ significantly. In fact, recent SSL algorithms, such as ReMixMatch [1] and CoMatch [18], rely on the assumption that the class distribution of the unlabeled set is the same as that of the labeled set and cannot consider a potential class distribution mismatch between the labeled and unlabeled sets.

Recently, many class imbalanced SSL (CISSL) algorithms [7, 12, 15, 17, 21, 26] have been proposed. However, Fan et al. [7], Lee et al. [17], Wei et al. [26] assumed that the class distribution of the unlabeled set is known and the same as that of the labeled set, although the class distribution of the unlabeled set can be unknown in practice (e.g., STL-10 [2]) and training sets comprising labeled and unlabeled sets collected from different periods are likely to have a class distribution mismatch. Kim et al. [12], Lai et al. [15], Oh et al. [21] did not make an assumption of the same class distributions for labeled and unlabeled sets in the main training stage. However, after the main training stage, they additionally used the re-balancing technique of Classifier Re-training (cRT) [11] or post-hoc logit-adjustment (LA) [20], which were proposed for fully supervised class-imbalanced learning. When using cRT for CISSL, there are disadvantages that the classifier cannot be learned interactively with representations, and only the labeled set is used for training the classifier [17]. Using LA for CISSL may not re-balance the classifier to an appropriate degree when the class distribution of the unlabeled set is unknown and differs from that of the labeled set, because LA can not consider the unknown class distribution of the unlabeled set.

We propose a CISSL algorithm, class-distribution-mismatch-aware debiasing (CDMAD), which effectively mitigates class imbalance in SSL even under severe class distribution mismatch between labeled and unlabeled sets. The key idea of CDMAD is to consider the classifier's biased degree towards each class for refining both the biased pseudo-labels of the base SSL algorithm and class predictions on test samples. To measure the classifier's biased degree, we utilize the class prediction on an input that is reasonably assumed to be irrelevant to the training set.

In general, a trained classifier predicts a class of a new sample based on the learned features. Therefore, for an im-



(a) FixMatch (b) ReMixMatch

Figure 1. Class probabilities on an image without any patterns.

age irrelevant to the learned features, the predicted class probabilities are expected to be uniform across classes. However, this may not be true when the training set is class-imbalanced because the classifier tends to be biased towards the majority classes. Fig. 1 illustrates the class probabilities predicted on an image without any patterns (white image) using FixMatch [22] and ReMixMatch [1], base SSL algorithms of the recent CISSL studies, trained on CIFAR-10 under $\gamma_l = \gamma_u = 1$ (class-balanced set), $\gamma_l = 100$ and $\gamma_u = 1$ (class-imbalanced set), and $\gamma_l = \gamma_u = 100$ (class-imbalanced set), where γ_l and γ_u denote the class imbalanced ratios for the labeled and unlabeled sets (formally defined in Sec. 3.1), respectively. The classifiers trained on the class-imbalanced sets produced highly nonuniform class probabilities for the white image, whereas they produced nearly uniform class probabilities for the same input when trained on the class-balanced set. Here, it may be reasonable to assume that the solid color image does not have the features learned from the training set. Then, the class probabilities for the solid color image can be thought of as predicted based solely on the classifier's biased degree towards each class, regardless of the learned features.

Motivated by the above finding, CDMAD measures the classifier's biased degree by calculating logits on a solid color image for each iteration of training. Then, CDMAD refines the biased pseudo-labels of a base SSL algorithm by adjusting for the measured bias of the classifier in the logits for unlabeled samples. The refined pseudo-labels are used to train the base SSL algorithm, which leads to the mitigation of class imbalance and improved quality of the representations. After training is completed, CDMAD similarly refines the biased class predictions on test samples by adjusting for the measured bias of the classifier in the logits for test samples. CDMAD can appropriately re-balance the classifier even under severe class distribution mismatch between labeled and unlabeled sets because the class distributions of both labeled and unlabeled sets can be implicitly considered when measuring the classifier's biased degree. In Sec. 3.4, we analyze that CDMAD can be viewed as an extension of LA, incorporating awareness of class distribution mismatch. Similar to LA, CDMAD is Fisher consistent for minimizing the balanced error [19].

Experimental results on four benchmark datasets verify that CDMAD outperforms baseline CISSL algorithms in both scenarios where the class distributions of the labeled and unlabeled sets either match or mismatch. Furthermore, through qualitative analysis and an ablation study, we demonstrate the effectiveness of each component of CDMAD. Unlike previous CISSL studies, CDMAD does not require additional parameters or training stages in comparison to the base SSL algorithm. Additionally, it can be implemented by simply adding a few lines of code into the existing code of the base SSL algorithms as presented in Appendix A. The code for the CDMAD is available at https://github.com/LeeHyuck/CDMAD.

2. Related Works

CReST [26] uses unlabeled samples predicted as the minority classes more frequently than those predicted as the majority classes for iterative self-training. ABC [17] and CoSSL [7] use an auxiliary classifier and train the classifier to be balanced. CoSSL generates pseudo-labels for base SSL algorithms using the balanced classifier. These studies assume that the class distribution of the unlabeled set is known and same as that of the labeled set. DARP [12] and DASO [21] refine biased pseudo-labels by iteratively solving a convex optimization problem and blending semantic pseudo-labels and linear pseudo-labels, respectively. SAW [15] mitigates class imbalance using smoothed reweighting based on the number of pseudo-labels belonging to each class. These studies additionally use CIL techniques, such as cRT [11] and LA [20], after the main training stage. Adsh [8] and InPL [28] use pseudo-labels based on class-dependent confidence thresholds and energy score threshold, respectively. DebiasPL [24] debiases pseudo-labels by mitigating the classifier response bias based on counterfactual reasoning. UDAL [16] unifies distribution alignment technique [1] and logit-adjusted loss [20] to progressively mitigate class-imbalance. L2AC [23] trains a bias adaptive classifier composed of a bias attractor and a linear classifier with bi-level optimization. ACR [27] dynamically refines pseudo-labels using an adaptive consistency regularizer that estimates the true class distribution of unlabeled set.

3. Methodology

3.1. Problem setup

Suppose that we have a training set with labeled set $\mathcal{X} = \{(x_n, y_n) : n \in (1, \ldots, N)\}$ and unlabeled set $\mathcal{U} = \{(u_m) : m \in (1, \ldots, M)\}$, where $x_n \in \mathbb{R}^d$ and $y_n \in [C] = \{1, \ldots, C\}$ denote the nth labeled sample and corresponding label, respectively, and $u_m \in \mathbb{R}^d$ denotes the mth unlabeled sample. We denote the number of labeled and unlabeled samples of class c as N_c and M_c, respectively, i.e., $\sum_{c=1}^{C} N_c = N$ and $\sum_{c=1}^{C} M_c = M$, where M_c is challenging to know in a realistic scenario. The C classes are sorted in descending order according to the cardinality of labeled samples, i.e., $N_1 \geq \cdots \geq N_C$. The ratio of the class imbalance of labeled and unlabeled sets are $\gamma_l = \frac{N_1}{N_C}$ and $\gamma_u = \frac{M_1}{M_C}$, respectively, where $\gamma_l \gg 1$ or $\gamma_u \gg 1$ in the class-imbalanced training set. When M_c is unknown, γ_u will be also unknown and can differ from γ_l. That is, the class distribution of the unlabeled set can be mismatched with that of the labeled set. For each iteration of training, we sample minibatches $\mathcal{MX} = \{(x_b^m, y_b^m) : b \in (1, \ldots, B)\} \subset \mathcal{X}$ and $\mathcal{MU} = \{(u_b^m) : b \in (1, \ldots, \mu B)\} \subset \mathcal{U}$ from the training set, where B denotes the minibatch size and μ denotes the relative size of $\mathcal{MU}$ to $\mathcal{MX}$. Using $\mathcal{MX}$ and $\mathcal{MU}$ for training, we aim to learn a classifier $f_\theta : \mathbb{R}^d \rightarrow \{1, \ldots, C\}$ that effectively classifies samples in a test set $\mathcal{X}^{test} = \{(x_k^{test}, y_k^{test}) : k \in (1, \ldots, K)\}$, where θ denotes parameters of base SSL algorithm. We denote the output logits of f_θ on an input as $g_\theta(\cdot) \in \mathbb{R}^C$, i.e, $f_\theta(\cdot) = \arg\max_c g_\theta(\cdot)_c$, where $(\cdot)_c$ denotes the cth element.

3.2. Base SSL algorithms

The proposed algorithm uses FixMatch [22] or ReMixMatch [1] as its base SSL algorithm, following other CISSL studies. FixMatch and ReMixMatch use hard or sharpened pseudo-labels for entropy minimization and strong data augmentation techniques [4, 6] for consistency regularization. Specifically, FixMatch first predicts the class probability of weakly augmented unlabeled data point $\alpha(u_b^m)$ as $q_b = P_\theta(y|\alpha(u_b^m))$ and then generates hard pseudo-label $\hat{q}_b$=$\arg\max_c (q_{b,c})$, where $P_\theta(y|\cdot) = \phi(g_\theta(\cdot))$ for softmax activation function ϕ. For consistency regularization, FixMatch uses hard pseudo-label $\hat{q}_b$ only when $\max_c(q_{b,c}) \geq \tau$, where τ denotes a predefined confidence threshold, to improve the quality of the pseudo-labels used for training.

ReMixMatch similarly produces q_b and aligns the distribution of q_b to the class distribution of the labeled set $P_l(y)$ as $\tilde{q}_b = Normalize(q_b \times P_l(y)/q(y))$, where $Normalize(x)_i = x_i / \sum_j x_j$ and $q(y)$ denotes the moving average of the class probabilities predicted over the last 128 unlabeled minibatches. Then, ReMixMatch sharpens the pseudo-label as $\bar{q}_b = Normalize\left(\tilde{q}_b^{1/T}\right)$, where $1/T$ is the sharpening temperature, $0 < T < 1$. With the sharpened pseudo-label $\bar{q}_b$, ReMixMatch conducts consistency regularization by encouraging the class prediction on $\mathcal{A}(u_b^m)$ to be consistent with $\bar{q}_b$. ReMixMatch also conducts Mixup regularization and self-supervised learning by rotating unlabeled samples [13]. Data augmentation techniques $\alpha(\cdot)$ and $\mathcal{A}(\cdot)$ are described in detail in Appendix C. We express the training losses of FixMatch $loss_F$ and ReMixMatch $loss_R$ on $\mathcal{MX}$ and $\mathcal{MU}$ as:

$$loss_F(\mathcal{MX}, \mathcal{MU}, \hat{q}, \tau; \theta), \quad (1)$$

$$loss_R(\mathcal{MX}, \mathcal{MU}, \bar{q}; \theta), \quad (2)$$

where $\hat{q}$ and $\bar{q}$ are concatenations of $\hat{q}_b$ and $\bar{q}_b$, $b = 1, \ldots, \mu B$, respectively. The losses are detailed in Appendix D.

The proposed algorithm uses FixMatch or ReMixMatch as its base SSL algorithm with some modifications as follows: **1)** The proposed algorithm does not use hard or sharpened pseudo-labels because entropy minimization of class predictions may cause the classifier to be biased towards certain classes [17]. **2)** The proposed algorithm does not use confidence threshold τ for FixMatch, enabling the utilization of all unlabeled samples. A potential limitation of utilizing inaccurate pseudo-labels can be alleviated by refining them, as discussed in Sec. 3.3. **3)** The proposed algorithm does not employ the distribution alignment technique for ReMixMatch when the class distribution of the unlabeled set is unknown. This is because the labeled and unlabeled sets can potentially have different class distributions while the distribution alignment technique aligns the distribution of pseudo-labels with the class distribution of the labeled set. This modification helps prevent the generation of low-quality pseudo-labels in situations where there is a severe class distribution mismatch between the labeled and unlabeled sets, as discussed in Sec. 4.2.

3.3. CDMAD

Refinement of pseudo-labels during training

To refine a pseudo-label q_b generated by FixMatch or ReMixMatch, CDMAD first calculates the logits on a weakly augmented unlabeled sample, $g_\theta(\alpha(u_b^m))$, and the logits on an image without any patterns $g_\theta(\mathcal{I})$, where $\mathcal{I}$ denotes an image without any patterns (solid color image). The logits on a solid color image $g_\theta(I)$ is considered the classifier's biased degree towards each class regardless of the learned features, as discussed in Sec. 1. Then, CDMAD adjusts for the classifier's biased degree, $g_\theta(\mathcal{I})$, in the logits $g_\theta(\alpha(u_b^m))$, by simple subtraction as follows:

$$g_\theta^*(\alpha(u_b^m)) = g_\theta(\alpha(u_b^m)) - g_\theta(\mathcal{I}), \quad (3)$$

where $g_\theta^*(\cdot)$ denotes the refined logits, which are considered to be calculated based only on the learned features. With $g_\theta^*(\alpha(u_b^m))$, the refined pseudo-label q_b^* is obtained as:

$$q_b^* = \phi(g_\theta^*(\alpha(u_b^m))). \quad (4)$$

As noted in Sec. 3.2, CDMAD does not use the distribution alignment technique for ReMixMatch. Instead, CDMAD adds the supervised loss for weakly augmented labeled sample $Sup(\mathcal{MX}; \theta)$ into the training loss of ReMixMatch to enhance the classifier's familiarity with labeled samples. This can effectively improve the quality of pseudo-labels when the class distributions of labeled and unlabeled sets mismatch, as discussed in Sec. 4.2. The training losses for FixMatch and ReMixMatch with CDMAD, denoted by $loss_F^*$ and $loss_R^*$, respectively, are expressed as:

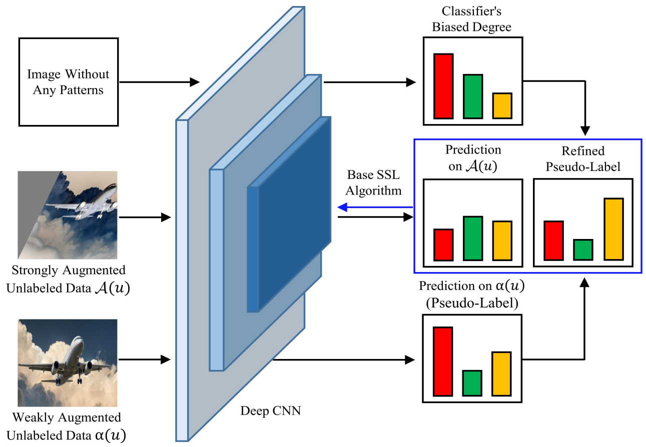


Figure 2. Pseudo-label refinement process using CDMAD.

$$loss_F^* = loss_F\left(\mathcal{MX}, \mathcal{MU}, q^*, 0; \theta\right), \tag{5}$$

$$loss_R^* = loss_R\left(\mathcal{MX}, \mathcal{MU}, q^*; \theta\right) + Sup(\mathcal{MX}; \theta), \tag{6}$$

where $loss_F$ and $loss_R$ are from Eq. (1) and Eq. (2), and q^* is the concatenation of the q_b^*. Fig. 2 illustrates the pseudo-label refinement process. By using the refined pseudo-labels during the training of the base SSL algorithm, the quality of representations is improved.

CDMAD can effectively refine the pseudo-labels even under severe class distribution mismatch between the labeled and unlabeled sets because classifier's biased degree $g_\theta(\mathcal{I})$ is affected by the class distributions of both sets. Unlike the previous CISSL studies, CDMAD does not require additional parameters for an auxiliary classifier or additional training stages. CDMAD can be implemented by adding a few lines of code as presented in Appendix A.

Refinement of biased class predictions during testing Even with all biased pseudo-labels perfectly refined during the training process, biased predictions may still be produced for the test samples because the training set is class-imbalanced. To refine biased class predictions on test samples, CDMAD also adjusts the biased logits on x_k^{test}, for $k = 1, \dots, K$. CDMAD first calculates the classifier's biased degree $g_\theta(\mathcal{I})$. Then, similar to Eq. (5), the logits for test samples, $g_\theta(x_k^{test})$, for $k = 1, \dots, K$, are adjusted as:

$$g_\theta^*\left(x_k^{test}\right) = g_\theta\left(x_k^{test}\right) - g_\theta\left(\mathcal{I}\right). \tag{7}$$

With the adjusted logits $g_\theta^*(x_k^{test})$, the refined class prediction $f_\theta^*(x_k^{test})$ is obtained as follows:

$$\begin{aligned} f_\theta^*\left(x_k^{test}\right) &= \arg\max_c g_\theta^*\left(x_k^{test}\right)_c \\ &= \arg\max_{y\in[C]} P_\theta\left(y|x_k^{test}\right)/P_\theta\left(y|\mathcal{I}\right). \end{aligned} \tag{8}$$

We illustrate the test process in Appendix E and pseudo code of the proposed algorithm in Appendix F.

3.4. CDMAD as a CISSL extension of post-hoc logit-adjustment (LA)

CDMAD can be viewed as a CISSL extension of LA [20] to take into account class distribution mismatch between labeled and unlabeled sets, where LA was originally introduced to re-balance a biased classifier in CIL. To re-balance the classifier, LA post-adjusts the logits on test samples $g_\theta(x_k^{test})$ by simply subtracting the log of the estimate of the underlying class prior $P(y)$, denoted by π (e.g., each class frequency on the training set), as follows:

$$g_\theta^*\left(x_k^{test}\right) = g_\theta\left(x_k^{test}\right) - \log \pi. \tag{9}$$

The adjustment in Eq. (9) was proven to be Fisher consistent for minimizing the balanced error rate (BER),

$$\text{BER}\left(f_\theta^*\right) = \frac{1}{C}\sum_{y\in[C]} P_{x|y}\left(y \neq f_\theta^*\left(x\right)\right). \tag{10}$$

In addition, empirical evidence has demonstrated that LA can effectively enhance classification performance across various class-imbalanced learning scenarios.

However, in CISSL, where the class distribution of the unlabeled set is unknown and may substantially differ from that of the labeled set, LA may result in the classifier being re-balanced to an inappropriate degree, thereby leading to a decrease in classification performance. This limitation arises from the challenge of incorporating the unknown class distribution of the unlabeled set for re-balancing. Specifically, the class prior $P(y)$, which is often approximated as the class distribution of the labeled set $P_l(y)$, cannot consider the class distribution of the unlabeled set.

By comparing Eq. (7) and Eq. (9), we can observe that CDMAD shares a similar form with LA. Specifically, given that $g_\theta(\mathcal{I}) + constant = \log P_\theta(y|\mathcal{I})$, CDMAD can be seen as replacing the class frequencies π in Eq. (9) by $P_\theta(y|\mathcal{I})$, which can be considered an estimate of the *classifier's prior* $P_\theta(y)$. (With this interpretation, CDMAD can be viewed as allowing the classifier to predict class probabilities based solely on the given input, without being affected by the classifier's prior $P_\theta(y)$.) This replacement allows CDMAD to implicitly incorporate the class distributions of both labeled and unlabeled sets, facilitating its awareness of class distribution mismatch between the two sets, as we will further discuss in Sec. 4.3. Furthermore, whereas LA is solely employed to refine biased class predictions on the test set, CDMAD is also employed to refine the biased pseudo-labels. Similar to LA, CDMAD ensures Fisher consistency for the balanced error.

Proposition 1. *Given a solid color image $\mathcal{I}$ independent of class labels y, the refinement by CDMAD in Eq. (8) is Fisher consistent for minimizing the BER in Eq. (10).*

Proof. Due to the universal approximation theorem [9, 29], $P_\theta(y)$ becomes $P(y)$, and $P_\theta(y|I)$ becomes $P(y|I)$, under the population setting. By the assumption of Proposition 1, i.e., $P(y) = P(y|I)$, it follows that $P_\theta(y) = P_\theta(y|I)$. The refined class prediction for input x, $f^*_\theta(x) = \arg\max_{y\in[C]} P_\theta(y|x) / P_\theta(y|\mathcal{I}) = \arg\max_{y\in[C]} P_\theta(y|x) / P_\theta(y)$. If the network is trained with the entire population, $P_\theta(y|x)$ becomes $P(y|x)$, and $P_\theta(y) = \int P_\theta(y|x) P(x)\, dx$ becomes $\int P(y|x) P(x)\, dx = P(y)$, due to the universal approximation theorem. Then, $f^*_\theta(x) = \arg\max_{y\in[C]} P_\theta(y|x) / P_\theta(y)$ becomes $\arg\max_{y\in[C]} P(y|x) / P(y) = \arg\max_{y\in[C]} P(x|y)$, which minimizes the BER [3, 20]. $\square$

Fisher consistency is a desirable property for an estimator and implies that in the entire population setting, optimizing the estimator yields the best result. In our case, we prove that in the population setting, CDMAD minimizes the BER. However, in practice, the entire population is not available, and thus the theory cannot be directly applied. Nevertheless, Proposition 1 makes us to be optimistic about that the algorithm is learned in the direction to minimize the BER.

4. Experiments

4.1. Experimental setup

We conducted experiments on CIFAR-10-LT, CIFAR-100-LT [5], STL-10-LT [12], and Small-ImageNet-127 [7] under the settings considered in previous studies [7, 15]. We used the balanced accuracy (bACC) [10] and geometric mean (GM) [14] to evaluate the classification performance on CIFAR-10-LT and STL-10-LT and only bACC to evaluate classification performance on CIFAR-100-LT and Small-ImageNet-127, following Fan et al. [7]. For CIFAR-10-LT, CIFAR-100-LT, and STL-10-LT, we repeated the experiments three times and report the average and standard error for the performance measures. We used a white image to measure the $g_\theta(y)$. Performance measures, description of datasets and experimental setup, and baseline algorithms are detailed in Appendices G, H, and I, respectively.

4.2. Experimental results

Tab. 1 summarizes bACC and GM of the baseline algorithms and proposed algorithm on CIFAR-10-LT when γ_u is assumed to be known and equal to γ_l. We can first observe that the vanilla algorithm (Deep CNN trained with cross-entropy loss) performed the worst. CIL (Re-sampling, LDAM-DRW, and cRT) mitigated class imbalance but did not significantly improve the classification performance compared to the vanilla algorithm. These results demonstrate the importance of using the unlabeled set. Compared to the vanilla algorithm, SSL algorithms (FixMatch and ReMixMatch) significantly improved the classification performance. However, their lower performance than that of the CISSL algorithms highlights the importance of mitigating class imbalance. By mitigating class imbalance and leveraging unlabeled data, CISSL algorithms achieved higher performance than the other algorithms. Overall, the proposed algorithm outperformed all other algorithms. This may be because CDMAD effectively refined biased pseudo-labels and class predictions on the test set by considering the classifier's biased degree.

Table 1. Comparison of bACC/GM on CIFAR-10-LT

CIFAR-10-LT ($\gamma = \gamma_l = \gamma_u$, γ_u is assumed to be known)			
Algorithm	$\gamma = 50$	$\gamma = 100$	$\gamma = 150$
Vanilla	65.2$_{\pm 0.05}$ / 61.1$_{\pm 0.09}$	58.8$_{\pm 0.13}$ / 58.2$_{\pm 0.11}$	55.6$_{\pm 0.43}$ / 44.0$_{\pm 0.98}$
Re-sampling	64.3$_{\pm 0.48}$ / 60.6$_{\pm 0.67}$	55.8$_{\pm 0.47}$ / 45.1$_{\pm 0.30}$	52.2$_{\pm 0.05}$ / 38.2$_{\pm 1.49}$
LDAM-DRW	68.9$_{\pm 0.07}$ / 67.0$_{\pm 0.08}$	62.8$_{\pm 0.17}$ / 58.9$_{\pm 0.60}$	57.9$_{\pm 0.20}$ / 50.4$_{\pm 0.30}$
cRT	67.8$_{\pm 0.13}$ / 66.3$_{\pm 0.15}$	63.2$_{\pm 0.45}$ / 59.9$_{\pm 0.40}$	59.3$_{\pm 0.10}$ / 54.6$_{\pm 0.72}$
FixMatch	79.2$_{\pm 0.33}$ / 77.8$_{\pm 0.36}$	71.5$_{\pm 0.72}$ / 66.8$_{\pm 1.51}$	68.4$_{\pm 0.15}$ / 59.9$_{\pm 0.43}$
/++DARP+cRT	85.8$_{\pm 0.43}$ / 85.6$_{\pm 0.56}$	82.4$_{\pm 0.26}$ / 81.8$_{\pm 0.17}$	79.6$_{\pm 0.42}$ / 78.9$_{\pm 0.35}$
/+CReST+LA	85.6$_{\pm 0.36}$ / 81.9$_{\pm 0.45}$	81.2$_{\pm 0.70}$ / 74.5$_{\pm 0.99}$	71.9$_{\pm 2.24}$ / 64.4$_{\pm 1.75}$
/+ABC	85.6$_{\pm 0.26}$ / 85.2$_{\pm 0.29}$	81.1$_{\pm 1.14}$ / 80.3$_{\pm 1.29}$	77.3$_{\pm 1.25}$ / 75.6$_{\pm 1.65}$
/+CoSSL	86.8$_{\pm 0.30}$ / 86.6$_{\pm 0.25}$	83.2$_{\pm 0.49}$ / 82.7$_{\pm 0.60}$	80.3$_{\pm 0.55}$ / 79.6$_{\pm 0.57}$
/+SAW+LA	86.2$_{\pm 0.15}$ / 83.9$_{\pm 0.35}$	80.7$_{\pm 0.15}$ / 77.5$_{\pm 0.21}$	73.7$_{\pm 0.06}$ / 71.2$_{\pm 0.17}$
/+Adsh	83.4$_{\pm 0.06}$ / -	76.5$_{\pm 0.35}$ / -	71.5$_{\pm 0.30}$ / -
/+DebiasPL	- / -	80.6$_{\pm 0.50}$ / -	- / -
/+UDAL	86.5$_{\pm 0.29}$ / -	81.4$_{\pm 0.39}$ / -	77.9$_{\pm 0.33}$ / -
/+L2AC	- / -	82.1$_{\pm 0.57}$ / 81.5$_{\pm 0.64}$	77.6$_{\pm 0.53}$ / 75.8$_{\pm 0.71}$
/+CDMAD	**87.3**$_{\pm 0.12}$ / **87.0**$_{\pm 0.15}$	**83.6**$_{\pm 0.46}$ / **83.1**$_{\pm 0.57}$	**80.8**$_{\pm 0.86}$ / **79.9**$_{\pm 1.07}$
ReMixMatch	81.5$_{\pm 0.26}$ / 80.2$_{\pm 0.32}$	73.8$_{\pm 0.38}$ / 69.5$_{\pm 0.84}$	69.9$_{\pm 0.47}$ / 62.5$_{\pm 0.35}$
/+DARP+cRT	87.3$_{\pm 0.61}$ / 87.0$_{\pm 0.11}$	83.5$_{\pm 0.07}$ / 83.1$_{\pm 0.09}$	79.7$_{\pm 0.54}$ / 78.9$_{\pm 0.49}$
/+CReST+LA	84.2$_{\pm 0.11}$ / -	81.3$_{\pm 0.34}$ / -	79.2$_{\pm 0.31}$ / -
/+ABC	87.9$_{\pm 0.47}$ / 87.6$_{\pm 0.51}$	84.5$_{\pm 0.32}$ / 84.1$_{\pm 0.36}$	80.5$_{\pm 1.18}$ / 79.5$_{\pm 1.36}$
/+CoSSL	87.7$_{\pm 0.21}$ / 87.6$_{\pm 0.25}$	84.1$_{\pm 0.56}$ / 83.7$_{\pm 0.66}$	81.3$_{\pm 0.83}$ / 80.5$_{\pm 0.76}$
/+SAW+cRT	87.6$_{\pm 0.21}$ / 87.4$_{\pm 0.26}$	85.4$_{\pm 0.32}$ / 83.9$_{\pm 0.21}$	79.9$_{\pm 0.15}$ / 79.9$_{\pm 0.12}$
/+CDMAD	**88.3**$_{\pm 0.35}$ / **88.1**$_{\pm 0.35}$	**85.5**$_{\pm 0.46}$ / **85.3**$_{\pm 0.44}$	**82.5**$_{\pm 0.23}$ / **82.0**$_{\pm 0.30}$

Tab. 2 summarizes bACC and GM of the baseline algorithms and proposed algorithm on CIFAR-10-LT and STL-10-LT when γ_u is unknown and different from γ_l. ReMixMatch performed poorly when γ_l and γ_u differed significantly, probably because the distribution alignment technique employed in ReMixMatch significantly degraded the quality of pseudo-labels. By aligning the class distribution of the pseudo-labels with that of the unlabeled set estimated as in Kim et al. [12], ReMixMatch* significantly improved the classification performance. However, the estimation of the class distribution of the unlabeled set becomes more time-consuming as the amount of unlabeled data increases. Furthermore, the estimation process requires more than 10 labeled samples for each class, making it unsuitable for datasets with a very small number of labeled samples, such as CIFAR-100-LT. In contrast, CDMAD does not rely on the estimated class distribution of the unlabeled set, making it more effective than baseline algorithms combined with ReMixMatch* for real-world scenarios. We can also observe that the LA decreased the performance of ReMixMatch*+DARP and ReMixMatch*+SAW when γ_l and γ_u differed significantly. This may be because the LA considers only the class distribution of the labeled set for rebalancing the classifier when the class distribution of the unlabeled set is unknown. In Appendix K, we present further comparisons of LA and CDMAD under the settings that the class distributions of labeled and unlabeled sets mismatch.

Table 2. Comparison of bACC/GM on CIFAR-10-LT and STL-10-LT under $\gamma_l \neq \gamma_u$ (γ_u is assumed to be unknown). ReMixMatch* denotes ReMixMatch with the estimated class distribution of the unlabeled set [12].

	CIFAR-10-LT ($\gamma_l = 100$, γ_u is assumed to be unknown)			STL-10-LT (γ_u =Unknown)	
Algorithm	$\gamma_u = 1$	$\gamma_u = 50$	$\gamma_u = 150$	$\gamma_l = 10$	$\gamma_l = 20$
FixMatch	68.9$_{\pm1.95}$/ 42.8$_{\pm8.11}$	73.9$_{\pm0.25}$/ 70.5$_{\pm0.52}$	69.6$_{\pm0.60}$/ 62.6$_{\pm1.11}$	72.9$_{\pm0.09}$/ 69.6$_{\pm0.01}$	63.4$_{\pm0.21}$/ 52.6$_{\pm0.09}$
FixMatch+DARP	85.4$_{\pm0.55}$/ 85.0$_{\pm0.65}$	77.3$_{\pm0.17}$/ 75.5$_{\pm0.21}$	72.9$_{\pm0.24}$/ 69.5$_{\pm0.18}$	77.8$_{\pm0.33}$/ 76.5$_{\pm0.40}$	69.9$_{\pm1.77}$/ 65.4$_{\pm3.07}$
FixMatch+DARP+LA	86.6$_{\pm1.11}$/ 86.2$_{\pm1.15}$	82.3$_{\pm0.32}$/ 81.5$_{\pm0.29}$	78.9$_{\pm0.23}$/ 77.7$_{\pm0.06}$	78.6$_{\pm0.30}$/ 77.4$_{\pm0.40}$	71.9$_{\pm0.49}$/ 68.7$_{\pm0.51}$
FixMatch+DARP+cRT	87.0$_{\pm0.70}$/ 86.8$_{\pm0.67}$	82.7$_{\pm0.21}$/ 82.3$_{\pm0.25}$	80.7$_{\pm0.44}$/ 80.2$_{\pm0.61}$	79.3$_{\pm0.23}$/ 78.7$_{\pm0.21}$	74.1$_{\pm0.61}$/ 73.1$_{\pm1.21}$
FixMatch+ABC	82.7$_{\pm0.49}$/ 81.9$_{\pm0.68}$	82.7$_{\pm0.64}$/ 82.0$_{\pm0.76}$	78.4$_{\pm0.87}$/ 77.2$_{\pm1.07}$	79.1$_{\pm0.46}$/ 78.1$_{\pm0.57}$	73.8$_{\pm0.15}$/ 72.1$_{\pm0.15}$
FixMatch+SAW	81.2$_{\pm0.68}$/ 80.2$_{\pm0.91}$	79.8$_{\pm0.25}$/ 79.1$_{\pm0.32}$	74.5$_{\pm0.97}$/ 72.5$_{\pm1.37}$	-/-	-/-
FixMatch+SAW+LA	84.5$_{\pm0.68}$/ 84.1$_{\pm0.78}$	82.9$_{\pm0.38}$/ 82.6$_{\pm0.38}$	79.1$_{\pm0.81}$/ 78.6$_{\pm0.91}$	-/-	-/-
FixMatch+SAW+cRT	84.6$_{\pm0.23}$/ 84.4$_{\pm0.26}$	81.6$_{\pm0.38}$/ 81.3$_{\pm0.32}$	77.6$_{\pm0.40}$/ 77.1$_{\pm0.41}$	-/-	-/-
FixMatch+CDMAD	**87.5**$_{\pm0.46}$/ **87.1**$_{\pm0.50}$	**85.7**$_{\pm0.36}$/ **85.3**$_{\pm0.38}$	**82.3**$_{\pm0.23}$/ **81.8**$_{\pm0.29}$	**79.9**$_{\pm0.23}$/ **78.9**$_{\pm0.38}$	**75.2**$_{\pm0.40}$/ **73.5**$_{\pm0.31}$
ReMixMatch	48.3$_{\pm0.14}$/ 19.5$_{\pm0.85}$	75.1$_{\pm0.43}$/ 71.9$_{\pm0.77}$	72.5$_{\pm0.10}$/ 68.2$_{\pm0.32}$	67.8$_{\pm0.45}$/ 61.1$_{\pm0.92}$	60.1$_{\pm1.18}$/ 44.9$_{\pm1.52}$
ReMixMatch*	85.0$_{\pm1.35}$/ 84.3$_{\pm1.55}$	77.0$_{\pm0.12}$/ 74.7$_{\pm0.04}$	72.8$_{\pm0.10}$/ 68.8$_{\pm0.21}$	76.7$_{\pm0.15}$/ 73.9$_{\pm0.32}$	67.7$_{\pm0.46}$/ 60.3$_{\pm0.76}$
ReMixMatch*+DARP	86.9$_{\pm0.10}$/ 86.4$_{\pm0.15}$	77.4$_{\pm0.22}$/ 75.0$_{\pm0.25}$	73.2$_{\pm0.11}$/ 69.2$_{\pm0.31}$	79.4$_{\pm0.07}$/ 78.2$_{\pm0.10}$	70.9$_{\pm0.44}$/ 67.0$_{\pm1.62}$
ReMixMatch*+DARP+LA	81.8$_{\pm0.45}$/ 80.9$_{\pm0.40}$	83.9$_{\pm0.42}$/ 83.4$_{\pm0.45}$	81.1$_{\pm0.20}$/ 80.3$_{\pm0.26}$	80.6$_{\pm0.45}$/ 79.6$_{\pm0.55}$	76.8$_{\pm0.60}$/ 74.8$_{\pm0.68}$
ReMixMatch*+DARP+cRT	88.7$_{\pm0.25}$/ 88.5$_{\pm0.25}$	83.5$_{\pm0.53}$/ 83.1$_{\pm0.51}$	80.9$_{\pm0.25}$/ 80.3$_{\pm0.31}$	80.9$_{\pm0.53}$/ 80.0$_{\pm0.46}$	76.7$_{\pm0.50}$/ 74.9$_{\pm0.70}$
ReMixMatch+ABC	76.4$_{\pm5.34}$/ 74.8$_{\pm6.05}$	85.2$_{\pm0.20}$/ 84.7$_{\pm0.25}$	80.4$_{\pm0.40}$/ 80.0$_{\pm0.44}$	76.8$_{\pm0.52}$/ 74.8$_{\pm0.64}$	71.2$_{\pm1.37}$/ 67.4$_{\pm1.89}$
ReMixMatch*+SAW	87.0$_{\pm0.75}$/ 86.4$_{\pm0.85}$	80.6$_{\pm1.57}$/ 79.2$_{\pm2.19}$	77.6$_{\pm0.76}$/ 76.0$_{\pm0.93}$	-/-	-/-
ReMixMatch*+SAW+LA	74.2$_{\pm1.49}$/ 65.1$_{\pm2.36}$	84.8$_{\pm1.07}$/ 82.4$_{\pm2.32}$	81.3$_{\pm2.42}$/ 80.9$_{\pm2.47}$	-/-	-/-
ReMixMatch*+SAW+cRT	88.8$_{\pm0.79}$/ 88.6$_{\pm0.83}$	84.5$_{\pm0.78}$/ 83.6$_{\pm1.27}$	82.4$_{\pm0.10}$/ 82.0$_{\pm0.10}$	-/-	-/-
ReMixMatch+CDMAD	**89.9**$_{\pm0.45}$/ **89.6**$_{\pm0.46}$	**86.9**$_{\pm0.21}$/ **86.7**$_{\pm0.17}$	**83.1**$_{\pm0.46}$/ **82.7**$_{\pm0.50}$	**83.0**$_{\pm0.38}$/ **82.1**$_{\pm0.35}$	**81.9**$_{\pm0.32}$/ **80.9**$_{\pm0.44}$

Table 3. Comparison of bACC/GM under $\gamma_l = \gamma_u = 100$ (reversed).

CIFAR-10-LT, $\gamma_l = 100$, $\gamma_u = 100$ (reversed)					
Algorithm	ABC	SAW	SAW+LA	SAW+cRT	**CDMAD**
FixMatch+	69.5/66.8	72.3/68.7	74.1/72.0	75.5/73.9	**77.1/75.4**
ReMixMatch+	63.6/60.5	79.5/78.5	50.2/14.8	80.8/79.9	**81.7/81.0**

Table 4. Comparison of bACC on CIFAR-100-LT.

CIFAR-100-LT ($\gamma = \gamma_l = \gamma_u$, γ_u is assumed to be known)			
Algorithm	$\gamma = 20$	$\gamma = 50$	$\gamma = 100$
FixMatch	49.6$_{\pm0.78}$	42.1$_{\pm0.33}$	37.6$_{\pm0.48}$
FixMatch+DARP	50.8$_{\pm0.77}$	43.1$_{\pm0.54}$	38.3$_{\pm0.47}$
FixMatch+DARP+cRT	51.4$_{\pm0.68}$	44.9$_{\pm0.54}$	40.4$_{\pm0.78}$
FixMatch+CReST	51.8$_{\pm0.12}$	44.9$_{\pm0.50}$	40.1$_{\pm0.65}$
FixMatch+CReST+LA	52.9$_{\pm0.07}$	47.3$_{\pm0.17}$	42.7$_{\pm0.70}$
FixMatch+ABC	53.3$_{\pm0.79}$	46.7$_{\pm0.26}$	41.2$_{\pm0.06}$
FixMatch+CoSSL	53.9$_{\pm0.78}$	47.6$_{\pm0.57}$	43.0$_{\pm0.61}$
FixMatch+UDAL	-	48.0$_{\pm0.56}$	43.7$_{\pm0.41}$
FixMatch+CDMAD	**54.3**$_{\pm0.44}$	**48.8**$_{\pm0.75}$	**44.1**$_{\pm0.29}$
ReMixMatch	51.6$_{\pm0.43}$	44.2$_{\pm0.59}$	39.3$_{\pm0.43}$
ReMixMatch+DARP	51.9$_{\pm0.35}$	44.7$_{\pm0.66}$	39.8$_{\pm0.53}$
ReMixMatch+DARP+cRT	54.5$_{\pm0.42}$	48.5$_{\pm0.91}$	43.7$_{\pm0.81}$
ReMixMatch+CReST	51.3$_{\pm0.34}$	45.5$_{\pm0.76}$	41.0$_{\pm0.78}$
ReMixMatch+CReST+LA	51.9$_{\pm0.60}$	46.6$_{\pm1.14}$	41.7$_{\pm0.69}$
ReMixMatch+ABC	55.6$_{\pm0.35}$	47.9$_{\pm0.10}$	42.2$_{\pm0.12}$
ReMixMatch+CoSSL	55.8$_{\pm0.62}$	48.9$_{\pm0.61}$	44.1$_{\pm0.59}$
ReMixMatch+CDMAD	**57.0**$_{\pm0.32}$	**51.1**$_{\pm0.46}$	**44.9**$_{\pm0.42}$

We also conducted experiments under the setting that the class distribution of the unlabeled set is imbalanced in the opposite direction to the labeled set. From Tab. 3, we can observe that CDMAD outperforms the baseline algorithms.

Tab. 4 summarizes bACC of the baseline algorithms and the proposed algorithm on CIFAR-100-LT. The proposed algorithm outperformed baseline algorithms. These results demonstrate that the proposed algorithm is well-suited for CISSL on datasets with a large number of classes. Moreover, given that several minority classes in the training set have only one labeled sample when $\gamma = 100$, the results indicate that the proposed algorithm may outperform the baseline CISSL algorithms when the number of labeled samples from minority classes is extremely limited. This may be because CDMAD effectively compensates for the lack of labeled samples by well refining the biased pseudo-labels compared to the baseline algorithms.

Table 5. Comparison of bACC on Small-ImageNet-127 (size 32×32 and 64×64, γ_u is assumed to be known)

Small-ImageNet-127 ($\gamma = \gamma_l = \gamma_u$, γ_u is assumed to be known)		
Algorithm	32×32	64×64
FixMatch	29.7	42.3
FixMatch+DARP	30.5	42.5
FixMatch+DARP+cRT	39.7	51.0
FixMatch+CReST	32.5	44.7
FixMatch+CReST+LA	40.9	55.9
FixMatch+ABC	46.9	56.1
FixMatch+CoSSL	43.7	53.8
FixMatch+CDMAD	**48.4**	**59.3**

Tab. 5 summarizes bACC of the baseline algorithms on Small-ImageNet-127. For both sizes of Small-ImageNet-127, CDMAD outperformed the baseline algorithms by a large margin. The effective use of unlabeled samples through appropriate refinement of the pseudo-labels may allow the proposed algorithm suitable for CISSL on large-scale datasets. Given that the test set of Small-ImageNet-127 is class-imbalanced, the results also show that CDMAD can be suitable for CISSL with a class-imbalanced test set.

To verify that CDMAD can be also effectively com-

Table 6. Comparison of bACC/GM on CIFAR-10-LT with FreeMatch as the base SSL algorithm

CIFAR-10-LT		
Algorithm	$\gamma_l = \gamma_u = 100$	$\gamma_l = 100, \gamma_u = 1$
FreeMatch	75.4/72.9	74.2/69.5
FreeMatch+SAW+cRT	82.8/82.3	86.4/86.2
FreeMatch+CDMAD	**84.8/84.4**	**89.0/88.7**

bined with recent SSL algorithms, we conducted experiments by setting FreeMatch [25] as the base SSL algorithm. From Tab. 6, we can observe that CDMAD outperforms FreeMatch and FreeMatch+SAW+cRT.

We also compared the classification performance of CDMAD with ACR [27], a recent CISSL algorithm. From Tab. 7, we can observe that CDMAD outperforms ACR.

Table 7. Comparison of bACC/GM on CIFAR-10-LT

Algorithm/ CIFAR-10-LT	$\gamma_l = \gamma_u = 100$	$\gamma_l = 100, \gamma_u = 1$
FixMatch+ACR	81.8/81.4	85.6/85.3
FixMatch+CDMAD	**83.6/83.1**	**87.5/87.1**

We present additional experimental results in Appendix. Specifically, fine grained results (many/medium/few group performance) are summarized in Appendix L. In Appendix M, we compare the classification performance of CDMAD with DASO [21] whose classification performance was measured under different settings with the settings of ours.

4.3. Qualitative analyses

We argue that the CDMAD can implicitly consider the class distributions of both labeled and unlabeled sets when measuring the classifier's biased degree. To verify this argument, in Fig. 3 (a) and (b), we analyze the class probabilities predicted on a white image, $P_\theta(y|\mathcal{I})$, using FixMatch+CDMAD and ReMixMatch+CDMAD trained on CIFAR-10-LT under the three settings: 1) $\gamma_l = \gamma_u = 100$, 2) $\gamma_l = 100$ and $\gamma_u = 1$, and 3) $\gamma_l = \gamma_u = 1$.

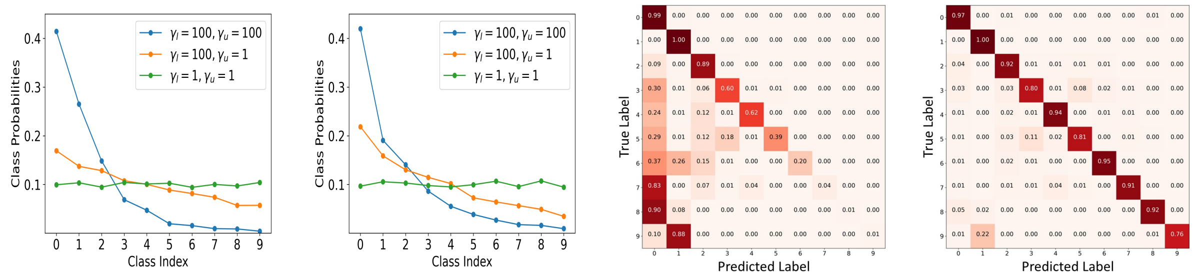
We can observe that both FixMatch+CDMAD and ReMixMatch+CDMAD produced highly nonuniform class probabilities when they were trained under $\gamma_l = 100$ and $\gamma_u = 100$. In contrast, when trained with $\gamma_l = 100$ and $\gamma_u = 1$, both algorithms produced significantly more balanced class probabilities. These results show that the classifier's biased degree, $g_\theta(\mathcal{I})$, depends on the class distribution of the unlabeled set. Moreover, the comparison of nearly uniform class probabilities produced under $\gamma_l = \gamma_u = 1$ and the results under $\gamma_l = 100$ and $\gamma_u = 1$ shows that $g_\theta(\mathcal{I})$ also depends on the class distribution of the labeled set. Based on the above findings, CDMAD can be considered as measuring the classifier's biased degree by implicitly incorporating the class distributions of both labeled and unlabeled sets. It is worth noting that under $\gamma_l = 100$ and $\gamma_u = 1$, FixMatch+CDMAD and ReMixMatch+CDMAD produced significantly more balanced class probabilities compared to FixMatch and ReMixMatch in Fig. 1. This may be because the use of biased pseudo-labels generated by FixMatch and ReMixMatch for training exacerbated class imbalance, whereas CDMAD effectively refined the biased pseudo-labels, as discussed in Figure 6 of Appendix J.

We also argue that the ability of CDMAD to implicitly incorporate the class distributions of both labeled and unlabeled sets enables it to effectively mitigate class imbalance even under severe class distribution mismatch. To verify this argument, we present the confusion matrices of the class predictions on the test set of CIFAR-10 using ReMixMatch and ReMixMatch+CDMAD trained on CIFAR-10-LT under $\gamma_l = 100$ and $\gamma_u = 1$ in Fig. 3 (c) and (d). The value in the ith row and jth column represents the proportion of the ith class samples classified as the jth class. We can observe that the class predictions of ReMixMatch in Fig. 3 (c) are biased towards the majority classes. Specifically, the data points in the minority classes (e.g., classes 7, 8 and 9) are often misclassified into the majority classes (e.g., classes 0 and 1). In contrast, ReMixMatch+CDMAD in Fig. 3 (d) made nearly balanced class predictions. Further qualitative analyses are presented in Appendix J.

4.4. Ablation study

To investigate the effectiveness of each element of CDMAD, we conducted an ablation study using CIFAR-10-LT ($\gamma_l = 100$ and $\gamma_u = 1$, γ_u is assumed to be unknown). Each row in Tab. 8 represents the proposed algorithm under the condition specified in that row. The results are as follows: 1) Without the refinement of the biased pseudo-labels using CDMAD in the training phase, the classification performance significantly decreased. 2) Without the refinement of the biased class predictions on the test set using CDMAD, the classification performance decreased. 3) With entropy minimization (using hard pseudo-labels and sharpened pseudo-labels) of the class predictions during training, the classification performance slightly decreased. 4) For FixMatch+CDMAD, the use of confidence threshold $\tau = 0.95$ slightly decreased the classification performance. 5) For ReMixMatch+CDMAD, the classification performance significantly decreased by using the distribution alignment technique instead of adding the supervised loss on $\alpha(x_b^m)$ for training. These results indicate that every element of CDMAD enhances performance.

To explore whether the classifier's biased degree can be measured using other images rather than the white image, we conducted experiments by replacing $\mathcal{I}$ with other solid color images or an image consisting of random pixel values that are generated from uniform, Bernoulli, and normal distributions. Experimental results are summarized in Tab. 9. From the table, we can observe that the classification performance of ReMixMatch+CDMAD decreased when $\mathcal{I}$ was replaced by images consisting of random pix-



(a) FixMatch+CDMAD (b) ReMixMatch+CDMAD (c) ReMixMatch (d) ReMixMatch+CDMAD

Figure 3. (a) and (b) present the class probabilities predicted on a white image using the proposed algorithm. (c) and (d) present the confusion matrices of the class predictions on test samples.

Table 8. Ablation study for the proposed algorithm on CIFAR-10-LT under $\gamma_l = 100$ and $\gamma_u = 1$

Ablation study ($\gamma_l = 100$, $\gamma_u = 1$)	**bACC/GM**		**bACC/GM**
FixMatch+CDMAD	**87.5/87.1**	**ReMixMatch+CDMAD**	**89.9/89.6**
Without CDMAD for refining pseudo-labels	78.2/75.8	Without CDMAD for refining pseudo-labels	72.3/65.9
Without CDMAD for test phase	84.9/84.1	Without CDMAD for test phase	88.2/87.7
With the use of hard pseudo-labels	86.7/86.3	With the use of sharpened pseudo-labels	88.9/88.6
With the use of confidence threshold $\tau = 0.95$	86.8/86.3	With the use of distribution alignment technique	80.4/78.5

Table 9. Experiments with the replacement of $\mathcal{I}$ by other inputs

ReMixMatch+CDMAD	CIFAR-10-LT	
Input	$\gamma_l = \gamma_u = 100$	$\gamma_l = 100$, $\gamma_u = 1$
Uniform	81.3/ 80.7	85.3/ 84.2
Bernoulli	82.5/ 82.0	83.6/ 82.8
Normal	78.4/ 77.5	84.0/ 83.2
Black	84.8/ 84.5	89.3/ 89.0
Red	84.8/ 84.6	90.1/ 89.9
Green	84.9/ 84.6	89.3/ 88.9
Blue	84.9/ 84.7	90.2/ 89.9
Gray	85.1/ 84.9	89.6/ 89.3
White	85.5/ 85.3	89.9/ 89.6

els. This may be because the parameters of the distributions (e.g., mean and standard deviation of a normal distribution) used to generate random pixels may be related to specific classes. In contrast, the classification performance of ReMixMatch+CDMAD did not significantly change when $\mathcal{I}$ was replaced by images of other solid color images. These results show that other solid color images can also be used to measure the classifier's biased degree.

Table 10. Experiments with replacing $\mathcal{I}$ by non-image input

Algorithm	CIFAR-10-LT	$\gamma_l = \gamma_u = 100$	$\gamma_l = 100$, $\gamma_u = 1$
FixMatch+CDMAD	White image	83.6/83.1	87.5/87.1
	Non-image	84.0/83.6	87.4/87.0
ReMixMatch+CDMAD	White image	85.5/85.3	89.9/89.6
	Non-image	85.6/85.4	89.8/89.6

However, the assumption that a solid color image is non-informative for the class labels may fail when the classification of images is related to their color. To address this concern, we additionally considered an image with pixel values that are outside the range [0, 255] (actually, *non-image input*) to replace the solid color image $\mathcal{I}$. For example, in the case of CIFAR-10-LT, the maximum value of each (R,G,B) channel becomes (2.06, 2.12, 2.11) after input normalization. In this case, if we generate an input with every pixel's (R,G,B) values set to (3, 3, 3), the input would not be associated with a specific class because it is actually not an image. By replacing $\mathcal{I}$ with this non-image input, we conducted experiments and presented the results in Tab. 10. We observe that CDMAD with the non-image input performs comparably with the white image. These results verify that the non-image input can be effectively used for measuring the classifier's biased degree, overcoming the challenge of finding data that are non-informative for the class labels.

5. Conclusion

We proposed CDMAD, which considers the classifier's biased degree towards each class to appropriately mitigate the class imbalance in SSL even under severe class distribution mismatch between the labeled and unlabeled sets. Using CDMAD, we refined biased pseudo-labels as well as biased class predictions on test samples. Experiments on four benchmark datasets show that the proposed algorithm outperforms the existing CISSL algorithms. Moreover, the qualitative analysis and ablation study on the proposed algorithm demonstrate the effectiveness of each component of CDMAD. In this paper, we used a solid color image to measure the classifier's biased degree, which lacks a firm theoretical basis. In future research, we plan to establish a theoretical foundation for utilizing the solid color image to measure the classifier's biased degree.

Acknowledgements This research was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (2023R1A2C2005453, RS-2023-00218913).

References

[1] David Berthelot, Nicholas Carlini, Ekin D Cubuk, Alex Kurakin, Kihyuk Sohn, Han Zhang, and Colin Raffel. Remixmatch: Semi-supervised learning with distribution matching and augmentation anchoring. In *International Conference on Learning Representations*, 2020. 1, 2, 3

[2] Adam Coates, Andrew Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 215–223. JMLR Workshop and Conference Proceedings, 2011. 1

[3] Guillem Collell, Drazen Prelec, and Kaustubh Patil. Reviving threshold-moving: a simple plug-in bagging ensemble for binary and multiclass imbalanced data. *arXiv preprint arXiv:1606.08698*, 2016. 5

[4] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 702–703, 2020. 3

[5] Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-balanced loss based on effective number of samples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9268–9277, 2019. 5

[6] Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017. 3

[7] Yue Fan, Dengxin Dai, Anna Kukleva, and Bernt Schiele. Cossl: Co-learning of representation and classifier for imbalanced semi-supervised learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14574–14584, 2022. 1, 2, 5

[8] Lan-Zhe Guo and Yu-Feng Li. Class-imbalanced semi-supervised learning with adaptive thresholding. In *International Conference on Machine Learning*, pages 8082–8094. PMLR, 2022. 2

[9] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366, 1989. 4

[10] Chen Huang, Yining Li, Chen Change Loy, and Xiaoou Tang. Learning deep representation for imbalanced classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5375–5384, 2016. 5

[11] Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, and Yannis Kalantidis. Decoupling representation and classifier for long-tailed recognition. In *International Conference on Learning Representations*, 2020. 1, 2

[12] Jaehyung Kim, Youngbum Hur, Sejun Park, Eunho Yang, Sung Ju Hwang, and Jinwoo Shin. Distribution aligning refinery of pseudo-label for imbalanced semi-supervised learning. In *Advances in Neural Information Processing Systems*, pages 14567–14579. Curran Associates, Inc., 2020. 1, 2, 5, 6

[13] Nikos Komodakis and Spyros Gidaris. Unsupervised representation learning by predicting image rotations. In *International conference on learning representations (ICLR)*, 2018. 3

[14] Miroslav Kubat, Stan Matwin, et al. Addressing the curse of imbalanced training sets: one-sided selection. In *Icml*, page 179. Citeseer, 1997. 5

[15] Zhengfeng Lai, Chao Wang, Henrry Gunawan, Sen-Ching S Cheung, and Chen-Nee Chuah. Smoothed adaptive weighting for imbalanced semi-supervised learning: Improve reliability against unknown distribution data. In *International Conference on Machine Learning*, pages 11828–11843. PMLR, 2022. 1, 2, 5

[16] Justin Lazarow, Kihyuk Sohn, Chen-Yu Lee, Chun-Liang Li, Zizhao Zhang, and Tomas Pfister. Unifying distribution alignment as a loss for imbalanced semi-supervised learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 5644–5653, 2023. 2

[17] Hyuck Lee, Seungjae Shin, and Heeyoung Kim. Abc: Auxiliary balanced classifier for class-imbalanced semi-supervised learning. *Advances in Neural Information Processing Systems*, 34:7082–7094, 2021. 1, 2, 3

[18] Junnan Li, Caiming Xiong, and Steven CH Hoi. Comatch: Semi-supervised learning with contrastive graph regularization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9475–9484, 2021. 1

[19] Aditya Menon, Harikrishna Narasimhan, Shivani Agarwal, and Sanjay Chawla. On the statistical consistency of algorithms for binary classification under class imbalance. In *International Conference on Machine Learning*, pages 603–611. PMLR, 2013. 2

[20] Aditya Krishna Menon, Sadeep Jayasumana, Ankit Singh Rawat, Himanshu Jain, Andreas Veit, and Sanjiv Kumar. Long-tail learning via logit adjustment. In *International Conference on Learning Representations*, 2020. 1, 2, 4, 5

[21] Youngtaek Oh, Dong-Jin Kim, and In So Kweon. Daso: Distribution-aware semantics-oriented pseudo-label for imbalanced semi-supervised learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9786–9796, 2022. 1, 2, 7

[22] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *Advances in Neural Information Processing Systems*, 33, 2020. 2, 3

[23] Renzhen Wang, Xixi Jia, Quanziang Wang, Yichen Wu, and Deyu Meng. Imbalanced semi-supervised learning with bias adaptive classifier. In *The Eleventh International Conference on Learning Representations*, 2023. 2

[24] Xudong Wang, Zhirong Wu, Long Lian, and Stella X Yu. Debiased learning from naturally imbalanced pseudo-labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14647–14657, 2022. 2

[25] Yidong Wang, Hao Chen, Qiang Heng, Wenxin Hou, Yue Fan, Zhen Wu, Jindong Wang, Marios Savvides, Takahiro Shinozaki, Bhiksha Raj, Bernt Schiele, and Xing Xie. Freematch: Self-adaptive thresholding for semi-supervised

learning. In *The Eleventh International Conference on Learning Representations*, 2023. 7

[26] Chen Wei, Kihyuk Sohn, Clayton Mellina, Alan Yuille, and Fan Yang. Crest: A class-rebalancing self-training framework for imbalanced semi-supervised learning. *arXiv preprint arXiv:2102.09559*, 2021. 1, 2

[27] Tong Wei and Kai Gan. Towards realistic long-tailed semi-supervised learning: Consistency is all you need. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3469–3478, 2023. 2, 7

[28] Zhuoran Yu, Yin Li, and Yong Jae Lee. InPL: Pseudo-labeling the inliers first for imbalanced semi-supervised learning. In *The Eleventh International Conference on Learning Representations*, 2023. 2

[29] Ding-Xuan Zhou. Universality of deep convolutional neural networks. *Applied and computational harmonic analysis*, 48 (2):787–794, 2020. 4