Debiased Sample Selection for Learning with Noisy Labels

Supplementary Material

1. Additional method details

1.1. Details of CCS

In this section, we present the details of the Mann-Kendall trend test [7, 11] used in CCS. Mann-Kendall trend test is a non-parametric method to detect trends in a series of data. It is based on the ranks of the data rather than their actual values, making it robust against extreme values. In CCS, we first collect model predictions for each sample during training. Without loss of generality, let $\{x_1, x_2, ..., x_n\}$ denote the collected model prediction confidence in a certain class of a certain sample where the subscript indicates the training epoch. Our goal is to judge whether this sequence presents an upward trend statistically. To achieve this, we first calculate the test statistic S as follows:

$$S = \sum_{j=2}^{n} \sum_{k=1}^{j-1} \operatorname{sgn}(x_j - x_k), \tag{1}$$

where n is length of the series and $\operatorname{sgn}(x_j - x_k)$ is the sign function:

$$\operatorname{sgn}(x_j - x_k) = \begin{cases} 1 & \text{if } x_j > x_k \\ 0 & \text{if } x_j = x_k \\ -1 & \text{if } x_j < x_k \end{cases} \tag{2}$$

S analyzes the sign differences between later and earlier data points in the sequence. If the sequence is monotonically increasing, S will be positive, while a monotonically decreasing series gives a negative S. The null hypothesis of the test is that there is no trend, which means that the observations are randomly ordered in the series. If S is significantly different from zero, we reject the null hypothesis and conclude that there is a trend. The variance of S under the null hypothesis is given by:

$$\operatorname{Var}(S) = \frac{1}{18}\left(n(n-1)(2n+5)\right), \tag{3}$$

where n is the number of data points. To facilitate implementation, we do not consider the tied ranks here. Finally, the standardized test statistic Z is calculated as:

$$Z = \begin{cases} \frac{S-1}{\sqrt{\operatorname{Var}(S)}} & \text{if } S > 0 \\ 0 & \text{if } S = 0 \\ \frac{S+1}{\sqrt{\operatorname{Var}(S)}} & \text{if } S < 0 \end{cases} \tag{4}$$

Our alternative hypothesis is the sequence has an upward trend. If $Z > Z_{1-\alpha}$ (α is the chosen significance level and $Z_{1-\alpha}$ is the $100\times(1-\alpha)$th percentile of the standard normal distribution) the probability of $\{x_1, x_2, .., x_n\}$ has no trend is less than α so we accept the alternative hypothesis. For simplicity, let MK-Test$(\cdot)$ denote the Mann-Kendall testing process, which takes a series of data as input and outputs the standardized test statistic Z. Let C_i^j represent the collected model prediction confidence sequence in class j for sample x_i during training. Formally, the construction of likely true label set for sample $(x_i, \tilde{y}_i)$ is as follows:

$$\mathcal{I}_i = \{j \mid \text{MK-Test}(C_i^j) > Z_{1-\alpha}\}. \tag{5}$$

Note we collect model predictions and update the test statistic S on the fly during training, so the inclusion of CCS does not incur significant computation overhead.

1.2. Further discussion on CCS's reliability

An essential question regarding CCS is whether the Mann-Kendall trend test can effectively identify the true label. To evaluate this, we cast the true label identification as binary classification: true labels as positives, others as negatives. Given a dataset with N samples and k candidate classes, this results in N positive samples and $N \times (k-1)$ negative samples. The Mann-Kendall test generates a statistic z for each label, indicating its correctness probability. We evaluate CCS using AUPRC and AUROC, computed separately for correctly-labeled and mislabeled samples.

As shown in Fig. 1, AUROC and AUPRC are tracked from epoch 30 (end of warm-up) to epoch 150 (end of training) under different noise conditions. Across all four noise settings, AUROC remains consistently above 0.95. AUPRC on Sym 50% and IDN 50% exhibits a rapid rise and remain above 0.95 till the end of training. Under Asym 40% and CIFAR-10N-Worst, AUPRC also increases to around 0.9 than slight declines at the end of training. This suggests that analyzing the confidence change trend is an effective way to identify the true label in the presence of label noise.

For the late-stage decline in AUROC and AUPRC under Asym 40% and CIFAR-10N-Worst, we attribute it to the highly feature-dependent nature of these noise patterns. In both settings, samples are mislabeled as visually similar categories (*e.g.* mislabel Dog to Cat), making it easier for the model to fit these incorrect labels. To the best of our knowledge, no method for learning with noisy labels can entirely eliminate such memorization. So the overall high metrics throughout most of training confirm the effectiveness of the Mann-Kendall trend test for correct label identification. Additionally, the fact that some incorrect labels can eventually attain a high statistic Z further supports the rationale behind

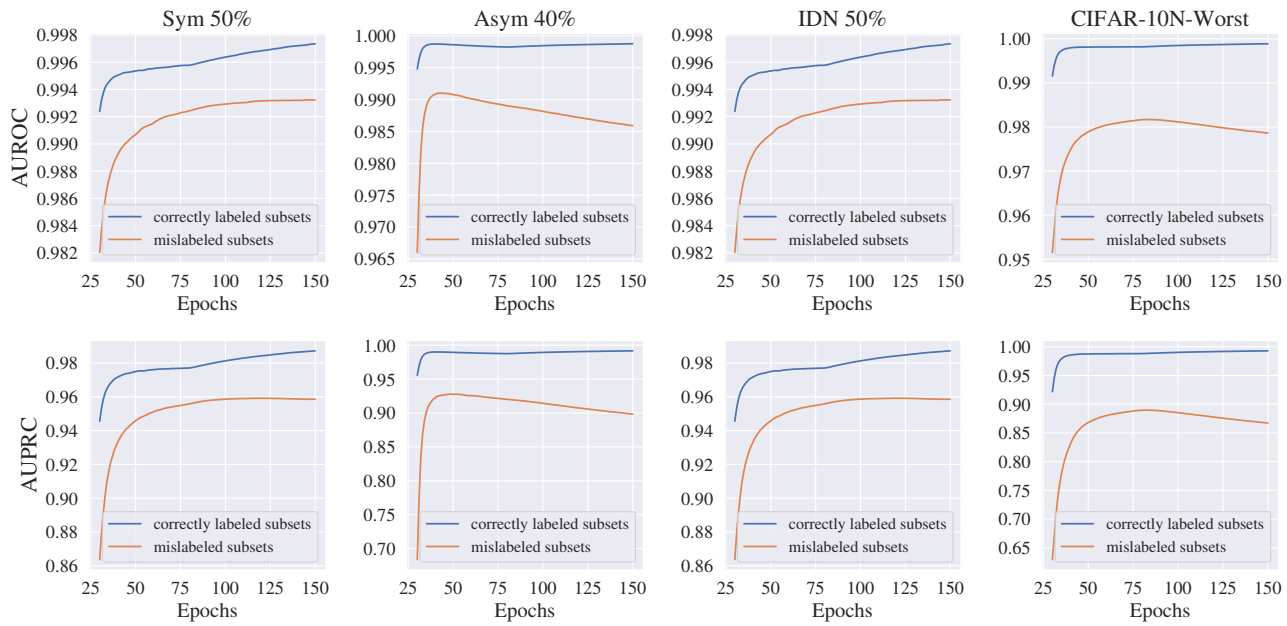

Figure 1. AUROC (top) and AUPRC (bottom) curves for distinguishing correct from incorrect labels using the Mann-Kendall trend test under four noise settings in CIFAR-10: symmetric noise (Sym 50%), asymmetric noise (Asym 40%), instance-dependent noise (IDN 50%), and real-world noisy labels (CIFAR-10N-Worst). The metrics are recorded from epoch 30 (end of warmup) to epoch 150 (end of training) and plotted separately for correctly labeled (blue) and mislabeled (orange) subsets. Higher curves indicate stronger separation between correct and incorrect labels, supporting the reliability of CCS.

Candidate Class Selection (CCS). Rather than directly relabeling, CCS employs dynamic exclusion: if a class exhibits a consistent upward trend in confidence, it is temporarily excluded from the candidate set. This ensures that even if the noisy label is mistakenly treated as a potential ground truth, it does not provide direct erroneous supervision to the model. Therefore, CCS remains stable even in the presence of incorrect label memorization.

In conclusion, CCS effectively identifies correct labels by analyzing confidence trend in model predictions throughout the training process, and its dynamic exclusion strategy further enhances the reliability of CCS.

1.3. Details of DSS+

In this section, we present the details of the cross-selection [4] and weak-strong consistency regularization [13] used in DSS+. Algorithm 1 provides the pseudo code of DSS+.

Cross-selection. In DSS, the selected sample set $\mathcal{C}$ and likely true labels for each sample $\{\mathcal{I}_i\}_{i=1}^N$ are generated by the model itself. This self-training procedure would accumulate errors in sample selection. To avoid this, we simultaneously train two networks to select correctly labeled samples $\mathcal{C}$ and likely true labels $\{\mathcal{I}_i\}_{i=1}^N$ for each other.

Weak-strong consistency regularization. We encourage the network to output consistent predictions on a weakly augmented image and its strongly augmented counterpart. We use the model's predictions on weakly augmented images as pseudo labels for the corresponding strongly augmented images. Following MixMatch [2], we apply temperature sharpening on model prediction to generate the pseudo label. Formally, for sample x_i, we generate its pseudo label y_i^p as follows:

$$y_i^p = \text{SumNorm}(p_\theta(\omega(x_i))^{1/T}), \tag{6}$$

where $\omega(\cdot)$ denote the weak augmentation, $p_\theta(x_i)$ is the model prediction for sample x_i and T is a hyperparameter which controls the temperature. We further enhance performance by conducting MixUp augmentation [16]. Specifically, when processing the ith example$(x_i, \tilde{y}_i)$ in a mini-batch, we randomly sample another example $(x_j, \tilde{y}_j)$, then apply MixUp augmentation and compute the consistency regularization term as follows:

$$\begin{aligned}
&\ell \sim \text{Beta}(\beta, \beta), \\
&\ell' = \max(\ell, 1-\ell), \\
&x_i' = \ell' \Omega(x_i) + (1-\ell')\Omega(x_j), \\
&y_i' = \ell' y_i^p + (1-\ell') y_j^p, \\
&\ell_{\text{reg}}(x_i, \tilde{y}_i) = \mathcal{H}(y_i', p_\theta(x_i')),
\end{aligned} \tag{7}$$

Algorithm 1: DSS+.

Input: noisy training dataset $\mathcal{D} = \{x_i, \tilde{y}_i\}_{i=1}^{N}$; two neural networks $f_{\theta_1}(\cdot)$, $f_{\theta_2}(\cdot)$ parameterized by θ_1 and θ_2; momentum coefficient λ used in MDA; significant level α used in Mann-Kendall test; sharpening temperature T; Beta distribution parameter β for mixup; regularization coefficient λ_{reg}; number of classes K; number of training epochs T_{max}; number of warm-up epochs T_w;

$p_1, p_2 = \mathbf{0}_{[N,K,T_{max}]}, \mathbf{0}_{[N,K,T_{max}]}$

$p'(y)_1, p'(y)_2 = \frac{1}{k}, \frac{1}{k}\ \forall y \in \mathcal{Y}$

$\mathcal{I}_{i,1}, \mathcal{I}_{i,2} = \emptyset, \emptyset\ \ \forall i \in [1, N]$

$\mathcal{C}_1, \mathcal{C}_2 = \mathcal{D}, \mathcal{D}$

for e *in* $1, ..., T_{max}$ **do**

  for j *in* $1, 2$ **do**

    foreach *minibatch*=$\{idx_i, x_i, \tilde{y}_i\}_{i=1}^{|b|}$ **do**

      # Marginal distribution adjustment.

      $p'(y)_j = \lambda p'(y)_j + (1-\lambda) \cdot \frac{1}{|b|} \sum_{x_i \in b} p_{\theta_j}(x_i)$

      $p_{bal}(y|x_i)_j = \mathrm{SumNorm}(\frac{p_{\theta_j}(x_i)}{p'(y)_j})$, for i in $1, ..., |b|$

      $p_j[idx_i, :, e] = p_{bal}(y|x_i)_j$, for i in $1, ..., |b|$

      # Candidate class selection, excluding likely true labels from candidate class set.

      $\mathcal{L}_x = \frac{1}{b} \sum_{i=1}^{b} \mathbb{I}((x_i, \tilde{y}_i) \in \mathcal{C}_j) \ell_{\mathrm{ccs}}(\tilde{y}_i, f_\theta(x_i), \mathcal{I}_{i,j})$,

      $\mathcal{L}_{reg} = \frac{1}{b} \sum_{i=1}^{b} \ell_{\mathrm{reg}}(x_i, \tilde{y}_i)$

      $\mathcal{L} = \mathcal{L}_x + \lambda_{reg} \mathcal{L}_{reg}$

      $\theta_j = \mathrm{SGD}(\mathcal{L}, \theta_j)$

    end

  end

  if $e \geq T_w$ **then**

    $\mathcal{C}_1 = \{(x_i, \tilde{y}_i) \mid \underset{y}{\mathrm{argmax}}\, p_{bal}(y|x_i)_2 = \tilde{y}_i\}$

    $\mathcal{C}_2 = \{(x_i, \tilde{y}_i) \mid \underset{y}{\mathrm{argmax}}\, p_{bal}(y|x_i)_1 = \tilde{y}_i\}$

    $\mathcal{I}_{i,1} = \{c \mid \text{MK-Test}(p_2[i, c]) > Z_{1-\alpha}\}$ for i in $1, ..., N$

    $\mathcal{I}_{i,2} = \{c \mid \text{MK-Test}(p_1[i, c]) > Z_{1-\alpha}\}$ for i in $1, ..., N$

  end

end

where β is the hyperparameter, $\Omega(\cdot)$ denote the strong augmentation, and $\mathcal{H}(\cdot, \cdot)$ denote the cross entropy between two inputs. The loss function used in DSS+ is as follows:

$$\mathcal{L} = \mathcal{L}_x + \lambda_{reg} * \frac{1}{n} \sum_{i=1}^{n} \ell_{\mathrm{reg}}(x_i, \tilde{y}_i), \tag{8}$$

where $\mathcal{L}_x$ is the loss on the selected subsets, λ_{reg} is the hyperparameter to control the strength of regularization. In our experiment, the consistency regularization term is calculated on all examples in the mini-batch. The only exception is the Clothing1M dataset, where we found that performing this regularization on noisy examples harms performance. This is because some classes being extremely noisy in Clothing1M, resulting in inaccurate pseudo labels. Hence, the MixUp augmentation and consistency regularization is only performed on the selected subsets in Clothing1M.

2. Implementation Details

All experiments are conducted on a system equipped with an Intel(R) Xeon(R) Platinum 8176 CPU @ 2.10GHz, 256GB RAM, and eight NVIDIA GeForce RTX 4090 graphics cards. We use PyTorch 2.1.0 with CUDA 12.8 and Python 3.8.16 on an Ubuntu 22.04.4 LTS operating system. For DSS, we follow the hyperparameter configuration in previous work CT [12]. We use PreActResNet-18 [5] as the backbone and train it for 150 epochs on both CIFAR-10 and CIFAR-100. The optimizer is SGD with a momentum of 0.9, a weight decay of 0.001, and a batch size of 128. The initial learning rate is set as 0.02 and reduced to 0.002 after 80 epochs. The momentum coefficient λ used in MDA and significant level α used in CCS are set to 0.99 and 0.10, respectively. For DSS+, we follow the hyperparameter configuration in previous works [3, 8, 10]. Table 1 summarizes the hyperparameter configurations on different

Table 1. The hyper-parameters for DSS+ on different benchmarks. T_{max} and T_w represent the total epochs and warm-up epochs, respectively. λ is the momentum coefficient when estimating marginal distribution in model prediction. α is the significant level used in the Mann-Kendall test. T is the sharpening temperature. β is the Beta distribution parameter for mixup. λ_{reg} is the coefficient of weak-strong consistency regularization. lr, bs, and wd are the abbreviations of the learning rate, batch size, and weight decay, respectively.

Dataset	Backbone	T_{max}	T_w	λ	α	T	β	λ_{reg}	lr	bs	wd	learning rate schedule
CIFAR-10N-Worst	PreActResNet-18	300	20	0.99	0.10	2	4.0	2	0.02	128	5e-4	cosine annealing to 4e-4
CIFAR-100N-Noisy	PreActResNet-18	300	30	0.99	0.10	2	4.0	1	0.02	128	5e-4	cosine annealing to 4e-4
Clothing1M	ResNet-50	50	1	0.99	0.50	2	0.5	1	2e-3	64	5e-4	divide 10 at epoch 15
miniWebVision	InceptionResNetV2	130	30	0.99	0.10	2	0.5	1	0.02	32	1e-4	divide 10 at epoch 50, 100

datasets. For experiments using CLIP as the visual encoder, we follow the training setting in previous work DeFT [14]. Specifically, we use the pre-trained ViT-B/16 in CLIP as the backbone, and train it for 10 epochs with 1 epoch warm-up. The optimizer is SGD with a momentum of 0.9, a weight decay of 5e-4, and a batch size of 64. The initial learning rate is set as 5e-4 and cosine annealing to 2e-4.

3. Additional Experiment and Analysis

In this section, we provide additional analysis and experiment results to further demonstrate the effectiveness of our bias-mitigating methods MDA and CCS:

- Section 3.1 shows how to extend MDA to handle Intrinsically long-tailed benchmarks.
- Section 3.2 intuitively demonstrates how MDA improves performance by rebalancing sample selection across different classes.
- Section 3.3 demonstrates that applying CCS can reduce the number of wrong labels memorized by the model.
- Section 3.4 shows that MDA and CCS can enhance various sample selection methods beyond BASE.
- Section 3.5 provides more intuitive examples showing how CCS prevents the model from memorizing mislabeled samples incorrectly selected by the sample selection module.
- Section 3.6 demonstrates that our method remains effective under high-level symmetric noise.
- Section 3.7 conducts additional ablation studies to further confirm the effectiveness of MDA and CCS on real noisy datasets.
- Section 3.8 conducts hyperparameter sensitivity experiments to demonstrate that our method is robust to the choice of momentum coefficient λ used in MDA and significant level α used in CCS.

3.1. Intrinsically long-tailed benchmarks

Real-world datasets often exhibit both label noise and a long-tailed distribution of true labels. To simulate this practical scenario, we adopt the setting introduced by Lu et al. [10], which first construct long-tailed versions of CIFAR-10/100 with an imbalance factor of 10 (largest / smallest class ratio) and then replace the clean labels with the noisy labels of CIFAR-10N-Worst and CIFAR-100N-Noisy, obtaining two benchmarks dubbed C10N-LT and C100N-LT.

Table 2. Test accuracy (%) on noisy and long-tailed benchmarks.

Method	prior π	C10N-LT	C100N-LT
CE	-	68.13	42.93
BASE	-	76.36	47.80
BASE+MDA	uniform	77.78	48.04
BASE+MDA+CCS	uniform	79.28	49.88
BASE+MDA	estimated	77.96	48.19
BASE+MDA+CCS	estimated	79.89	49.94

For such intrinsically unbalanced data, we extend MDA to adjust the marginal distribution in model prediction towards user-specified prior distribution π. The debiased model prediction $\hat{p}(y|x)$ in **??** is modified to:

$$\hat{p}(y|x) = \frac{p(y|x)\pi(y)}{p'(y)} \Big/ \sum_{c\in\mathcal{Y}} \frac{p(c|x)\pi(c)}{p'(c)}. \tag{9}$$

We evaluate two choices of prior distribution π:

- Uniform: $\pi(c) = 1/k, \forall c \in \mathcal{Y}$, which is identical to the MDA in main paper.
- Estimated: $\pi(c) = \sum_{(x,y)\in val} \mathbb{I}(y = c)/|val|$, which is obtained from a small clean validation set.

Table 2 reports the test accuracy of different configuration. Simply plugging MDA (uniform) into BASE improves performance by +0.2–1.4%, and further equipping CCS brings the total improvement to +2.0–2.9%, verifying that the two modules remain effective under noisy and unbalanced data. Switching from uniform to estimated prior gives an extra +0.1–0.6%, but this marginal benefit is much smaller than the gain produced by MDA and CCS themselves, indicating that the uniform prior is already a strong default when clean validation data are unavailable. In short, MDA and CCS consistently boost performance on both balanced and long-tailed noisy datasets, demonstrating their broad applicability.

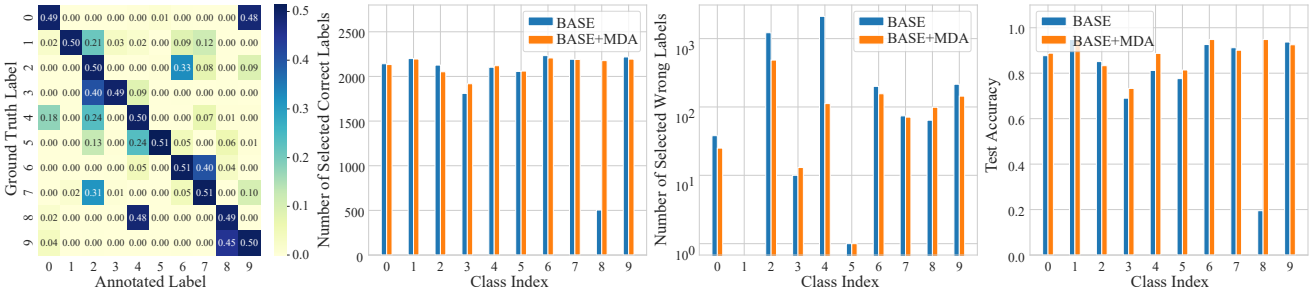

Figure 2. comparison of BASE and BASE+MDA on CIFAR-10 dataset with 50% instance-dependent noise. BASE+MDA yields fairer sample selection across classes, rescuing minority class (*i.e.* class 8) that is ignored by the BASE, boosting test accuracy on class 8.

3.2. Effectiveness of MDA

Figure 2 compares BASE and BASE+MDA on the CIFAR-10 dataset with 50% instance-dependent noise, showing that the performance improvement brought by MDA comes from its fairer sample selection. Specifically, in CIFAR-10 with 50% instance-dependent noise, 48% of samples from class 8 are mislabeled as class 4, making it difficult for the model to distinguish between them. Moreover, class 4 labels appear more frequently in the noisy dataset. Therefore, samples from the majority class 4 generally have lower loss than samples from minority class 8. Consequently, BASE over-selects class 4 while ignoring class 8. This introduces many mislabeled samples from class 4 while overlooking a large portion of correctly labeled samples from class 8. In contrast, after applying MDA, sample selection becomes more balanced. The number of correctly labeled samples from class 8 rises significantly from just over 500 to more than 2000, while the selection of mislabeled class 4 samples is greatly reduced. This fairer selection directly improves test accuracy for both class 4 and class 8. Overall, MDA proves to be an effective strategy for fairer sample selection, reducing class-level confirmation bias and improving model performance.

3.3. Effectiveness of CCS

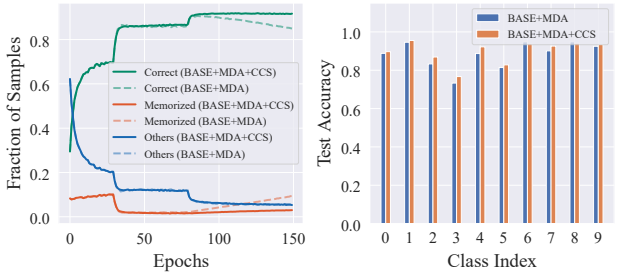

Figure 3. Fraction of different samples in each epoch and test accuracy in each class on CIFAR-10 with 50% IDN.

To demonstrate CCS's role in mitigating instance-level confirmation bias, Figure 3 analyzes the model's behavior on the training set. We categorize training samples into three groups: (1) Correct: the model's prediction matches the true label; (2) Memorized: the model's prediction matches an incorrect, noisy label; (3) Others: all remaining samples. Without CCS, the number of correct samples decreases while memorized samples increase, indicating that the model progressively memorizes incorrect labels, suffering from instance-level confirmation bias. By incorporating CCS, this issue is effectively mitigated. The fraction of "correct" samples remains stable, while "memorized" samples are minimized. This leads to higher test accuracy across nearly all classes, demonstrating CCS's ability to reduce the impact of instance-level confirmation bias and enhance model robustness.

3.4. Integrate MDA and CCS into other selector

As pluggable components, MDA and CCS can be combined with other sample selection methods beyond BASE. Tabel 3 demonstrates MDA and CCS can integrate into GMM, DIST and DIST+CT, yielding consistent gains, demonstrating the broad applicability of our approach.

Table 3. Test accuracy of different methods under CIFAR-10N-Worst (C10N) and CIFAR-100N-Noisy (C100N).

Dataset	C10N	C100N
GMM [1]	86.6±0.3	62.1±0.3
GMM+MDA	86.8±0.4	63.1±0.3
GMM+MDA+CCS	**87.2±0.3**	**64.4±0.3**
DIST [9]	86.4±0.5	61.0±0.2
DIST+MDA	86.6±0.5	63.3±0.2
DIST+MDA+CCS	**88.0±0.2**	**63.7±0.3**
DIST+CT [12]	87.0±0.4	62.2±0.2
DIST+CT+MDA	87.7±0.4	63.8±0.3
DIST+CT+MDA+CCS	**88.2±0.2**	**64.3±0.1**

Figure 4. Confidence trajectories when DSS+ with and without CCS on correctly and incorrectly labeled samples, illustrating CCS's crucial role in preventing memorization of noisy labels while preserving the model's ability to effectively learn from hard samples.

3.5. Case study

The trajectories in Fig 4 illustrate how CCS influences the model's handling of mislabeled and hard samples. For mislabeled samples, the model initially assigns the highest probability to the incorrect labels. So sample selector mistakenly treats these samples as correctly labeled and uses them in training. Without CCS, the model continues to train on these mislabeled samples with standard cross-entropy loss, which suppresses confidence in true labels and eventually causes the model to memorize incorrect labels. With CCS, true labels are excluded from the classification task, allowing the model's confidence in true labels to progressively increase until the correct prediction is retained, effectively mitigating the impact of noisy labels. For hard but correctly labeled samples, the model may initially show a rise in confidence for a similar but incorrect class. CCS temporarily excludes such incorrect classes from the classification task. However, as training progresses and the model's discrimination ability improves, confidence in those incorrect classes eventually decreases, allowing them to re-enter the classification task and let the model learn to distinguish similar classes. In general, these examples emphasize the crucial role of CCS in preventing the memorization of noisy labels while preserving the model's ability to effectively learn from hard samples.

3.6. High-level symmetric noise experiments

To demonstrate the performance of DSS+ under extreme noisy conditions, we conducted experiments on the CIFAR-100 dataset with 90%, 92%, and 95% symmetric noise. To make a fair comparison, for experiments under high-level symmetric noise, we adopt the same contrastive learning term as UNICON [6] and DULC [15]. Table 4 demonstrates that DSS+ outperforms the recent baseline DULC under high-level symmetric noise. Moreover, removing MDA or CCS causes significant performance degradation, confirm-

ing their effectiveness under high-level noise condition.

Table 4. Test accuracy (%) of different methods under CIFAR-100 with high-level symmetric noise.

Method\Noise type	Sym 90%	Sym 92%	Sym 95%
UNICON (2022)	44.80	32.24	19.37
DULC (2025)	52.80	45.32	25.61
DSS+	**57.16**	**47.05**	**26.84**
DSS+ w/o MDA	22.62	24.64	11.44
DSS+ w/o CCS	48.03	43.11	25.97

3.7. More ablation study

Table 5 shows that removing MDA or CCS leads to a marked drop in the real-world noisy datasets Clothing1M, WebVision and ILSVRC12, further confirming that MDA and CCS play a key role in the success of DSS+.

Table 5. Test accuracy (%) on Clothing1M, WebVision and ILSCVRC12 when remove MDA or CCS from DSS+.

Method\Dataset	Clothing1M	WebVision	ILSVRC12
DSS+	**75.13**	**82.40**	**78.48**
DSS+ w/o MDA	74.71	81.28	77.92
DSS+ w/o CCS	74.60	82.12	78.04

3.8. Hyper-parameter sensitive study

Our method involves two hyperparameters: the momentum coefficient λ in Marginal Distribution Adjustment (MDA) and the significance level α used in Mann-Kendall test in Candidate Class Selection (CCS). To evaluate the sensitivity of the proposed method to these hyperparameters, we analyze DSS's performance across different values of λ and α. As shown in Tab. 6, DSS remains stable performance across a range of λ and α, indicating that our method is robust to the selection of these hyperparameters.

Table 6. Test accuracy (%) of DSS under different hyperparameter configurations.

Hyperparameter		CIFAR-10N	CIFAR-100N
λ	α	Worst	Noisy
0.90	0.10	88.03±0.33	64.46±0.47
0.95	0.10	88.00±0.22	64.23±0.32
0.99	0.10	87.93±0.19	64.45±0.24
0.99	0.05	87.61±0.68	64.41±0.30
0.99	0.01	87.69±0.30	64.10±0.39

References

[1] Eric Arazo, Diego Ortego, Paul Albert, Noel E. O'Connor, and Kevin McGuinness. Unsupervised label noise modeling and loss correction. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, pages 312–321. PMLR, 2019. 5

[2] David Berthelot, Nicholas Carlini, Ian J. Goodfellow, Nicolas Papernot, Avital Oliver, and Colin Raffel. Mixmatch: A holistic approach to semi-supervised learning. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 5050–5060, 2019. 2

[3] Chen Feng, Georgios Tzimiropoulos, and Ioannis Patras. SSR: an efficient and robust framework for learning with unknown label noise. In *33rd British Machine Vision Conference 2022, BMVC 2022, London, UK, November 21-24, 2022*, page 372. BMVA Press, 2022. 3

[4] Bo Han, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor W. Tsang, and Masashi Sugiyama. Co-teaching: Robust training of deep neural networks with extremely noisy labels. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 8536–8546, 2018. 2

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part IV*, pages 630–645. Springer, 2016. 3

[6] Nazmul Karim, Mamshad Nayeem Rizve, Nazanin Rahnavard, Ajmal Mian, and Mubarak Shah. UNICON: combating label noise through uniform selection and contrastive learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 9666–9676. IEEE, 2022. 6, 7

[7] M.G. Kendall. *Rank Correlation Methods*. Charles Griffin, 4th edition, 1975. 1

[8] Noo-Ri Kim, Jin-Seop Lee, and Jee-Hyong Lee. Learning with structural labels for learning with noisy labels. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pages 27600–27610. IEEE, 2024. 3

[9] Yifan Li, Hu Han, Shiguang Shan, and Xilin Chen. DISC: learning from noisy labels via dynamic instance-specific selection and correction. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pages 24070–24079. IEEE, 2023. 5

[10] Yang Lu, Yiliang Zhang, Bo Han, Yiu-Ming Cheung, and Hanzi Wang. Label-noise learning with intrinsically long-tailed data. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023*, pages 1369–1378. IEEE, 2023. 3, 4

[11] Henry B. Mann. Nonparametric tests against trend. *Econometrica*, 13(3):245–259, 1945. 1

[12] Weiran Pan, Wei Wei, Feida Zhu, and Yong Deng. Enhanced sample selection with confidence tracking: Identifying correctly labeled yet hard-to-learn samples in noisy data. In *AAAI-25, Sponsored by the Association for the Advancement of Artificial Intelligence, February 25 - March 4, 2025, Philadelphia, PA, USA*, pages 19795–19803. AAAI Press, 2025. 3, 5

[13] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. 2

[14] Tong Wei, Hao-Tian Li, Chun-Shu Li, Jiang-Xin Shi, Yu-Feng Li, and Min-Ling Zhang. Vision-language models are strong noisy label detectors. In *Advances in Neural Information Processing Systems 37*, 2024. 4

[15] Yuanzhuo Xu, Xiaoguang Niu, Jie Yang, Ruiyi Su, Jian Zhang, Shubo Liu, and Steve Drew. Revisiting interpolation for noisy label correction. In *AAAI-25, Sponsored by the Association for the Advancement of Artificial Intelligence, February 25 - March 4, 2025, Philadelphia, PA, USA*, pages 21833–21841. AAAI Press, 2025. 6, 7

[16] Hongyi Zhang, Moustapha Cissé, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. 2