SELC: Self-Ensemble Label Correction Improves Learning with Noisy Labels

Yangdi Lu, Wenbo He
Department of Computing and Software, McMaster University, Canada
{luy100, hew11}@mcmaster.ca

Abstract

Deep neural networks are prone to overfitting noisy labels, resulting in poor generalization performance. To overcome this problem, we present a simple and effective method self-ensemble label correction (SELC) to progressively correct noisy labels and refine the model. We look deeper into the memorization behavior in training with noisy labels and observe that the network outputs are reliable in the early stage. To retain this reliable knowledge, SELC uses ensemble predictions formed by an exponential moving average of network outputs to update the original noisy labels. We show that training with SELC refines the model by gradually reducing supervision from noisy labels and increasing supervision from ensemble predictions. Despite its simplicity, compared with many state-of-the-art methods, SELC obtains more promising and stable results in the presence of class-conditional, instance-dependent, and real-world label noise. The code is available at https://github.com/MacLLL/SELC.

1 Introduction

The recent success of deep neural networks (DNNs) for vision tasks owes much to the availability of large-scale, correctly annotated datasets. However, obtaining such high-quality datasets can be extremely expensive, and sometimes even impossible. The common approaches, such as web queries [Li et al., 2017] and crowdsourcing [Song et al., 2019], can easily provide extensive labeled data, but unavoidably introduce noisy labels. Existing studies [Arpit et al., 2017; Zhang et al., 2021] have demonstrated that DNNs can easily overfit noisy labels, which deteriorates the generalization performance. Thus, it is essential to develop noise-robust algorithms for learning with noisy labels.

Given a noisy training set consisting of clean samples and mislabeled samples, a common category of approaches [Reed et al., 2015; Arazo et al., 2019; Zhang et al., 2020] to mitigating the negative influence of noisy labels is to identify and correct the mislabeled samples. However, the correction procedure in these methods only updates the noisy labels using the model prediction from the most recent training epoch directly, thus it may suffer from the false correction as the model predictions for noisy samples tend to fluctuate. Take a bird image mislabeled as an airplane as an example. During the training, the clean bird samples would encourage the model to predict a given bird image as a bird, while the bird images with airplane labels regularly pull the model back to predict the bird as an airplane. Hence, the model prediction gathered in one training epoch may change back and forth between bird and airplane, resulting in false correction.

We investigate the reason for performance degradation by analyzing the memorization behavior of the DNNs models. We observe that there exists a turning point during training. Before the turning point, the model only learns from easy (clean) samples, and thus model prediction is likely to be consistent with clean samples. After the turning point, the model increasingly memorizes hard (mislabeled) samples. Hence model prediction oscillates strongly on clean samples. Triggered by this observation, we seek to make the model retain the early-learning memory for consistent predictions on clean samples even after the turning point.

In this paper, we propose self-ensemble label correction (SELC), which potentially corrects noisy labels during training thus preventing the model from being affected by the noisy labels. SELC leverages the knowledge provided in the model predictions over historical training epochs to form a consensus of prediction (ensemble prediction) before the turning point. We demonstrate that combining ensemble prediction with the original noisy label leads to a better target. Accordingly, the model is gradually refined as the targets become less noisy, resulting in improving performance. However, it is challenging to find the turning point. Existing works estimate the turning point based on a test set or noise information, which are unobservable in practice. We propose a metric to estimate the turning point only using training data, allowing us to select a suitable initial epoch to perform SELC. Overall, our contributions are summarized as follows:

- We propose a simple and effective label correction method SELC based on self-ensembling.
- We design an effective metric based on unsupervised loss modeling to detect the turning point without requiring the test set and noise information.
- SELC achieves superior results and can be integrated with other techniques such as mixup [Zhang et al., 2018] to further enhance the performance.
2 Related Work

We briefly discuss the existing methods that do not require a small set of clean data (as opposed to [Xiao et al., 2015]).

Robust loss functions. Some methods aim to develop loss functions that are robust to label noise, including GCE [Zhang and Sabuncu, 2018], L_{DMI} [Xu et al., 2019], SCE [Wang et al., 2019] and NCE [Ma et al., 2020]. Loss correction. [Patrini et al., 2017] focus on correcting the loss function explicitly by estimating the noise transition matrix. Sample selection. Co-training-based methods [Han et al., 2018; Lu et al., 2021] maintain two networks, and each network is trained on low-risk samples which are selected by its peer network based on the small-loss criterion. Regularization. These methods [Liu et al., 2020; Lu et al., 2021] prevent memorization of mislabeled samples by using a regularizer. Label filtering. SELF [Nguyen et al., 2020] filters the mislabeled samples by ensemble predictions to improve the performance. Label correction. Joint Opt [Tanaka et al., 2018] and PENCIL [Yi and Wu, 2019] replace the noisy labels with soft (i.e. model probability) or hard (i.e. to one-hot vector) pseudo-labels. Bootstrap [Reed et al., 2015] and M-correction [Arazo et al., 2019] correct the labels by using a convex combination of noisy labels and the model prediction. PLC [Zhang et al., 2020] updates the noisy labels of high confident samples with model predictions.

Our method is related to label correction. Compared with existing methods, we focus on using ensemble prediction based on historical model outputs to correct the noisy labels, rather than only using prediction from the most recent training epoch. Our approach is straightforward and yields superior performance. Furthermore, our technique can be employed as an add-on component to further enhance the other approaches in challenging cases.

3 Preliminaries

Supervised Classification. Considering a supervised classification problem with $C$ classes, suppose $\mathcal{X} \in \mathbb{R}^d$ be the input space, $\mathcal{Y} \in \{0, 1\}^C$ is the ground-truth label space in an one-hot manner. In practice, the joint distribution $\mathcal{P}$ over $\mathcal{X} \times \mathcal{Y}$ is unknown. We have a training set $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$ which are independently sampled from $\mathcal{P}$. Assume a mapping function class $\mathcal{F}$ wherein each $f: \mathcal{X} \rightarrow \mathbb{R}^C$ maps the input space to $C$-dimensional score space, we seek $f^* \in \mathcal{F}$ that minimizes an empirical risk $\frac{1}{N} \sum_{i=1}^N \ell(y_i, f(x_i))$ for a certain loss function $\ell$.

Learning with Noisy Labels. Our goal is to learn from a noisy training distribution $\mathcal{P}_n$ where the labels are corrupted, with probability $\eta$, from their true distribution $\mathcal{P}$. Given a noisy training set $\tilde{\mathcal{D}} = \{(\tilde{x}_i, \tilde{y}_i)\}_{i=1}^N$, the observable noisy label $\tilde{y}_i$ has a probability of $\eta$ to be incorrect. Suppose the mapping function $f$ is a deep neural network classifier parameterized by $\Theta$. $f$ maps an input $x_i$ to $C$-dimensional logits $z_i = f(x_i, \Theta)$. We obtain conditional probability of each class by using a softmax function $S(\cdot)$, thus $p_i = S(z_i)$. Then the empirical risk on $\tilde{\mathcal{D}}$ using cross-entropy loss is

$$\mathcal{L}_{ce} = \frac{1}{N} \sum_{i=1}^N \ell_{ce}(\tilde{y}_i, p_i) = -\frac{1}{N} \sum_{i=1}^N (\tilde{y}_i)^\top \log(p_i).$$

(1)

When optimizing $\mathcal{L}_{ce}$ by stochastic gradient descent (SGD), the DNNs have been observed to completely fit the training set including mislabeled samples eventually (see Figure 1 (a)), resulting in the test performance degradation in the later stage of training (see Figure 1 (b)).

Noise Models. The generation of real-world label noise is unpredictable, a common methodology to cope with noisy labels is to posit a noise model and design robust algorithms under this model. Then we evaluate the algorithms on the real-world datasets to see their effectiveness. A common noise model is class-conditional noise [Natarajan et al., 2013], wherein label noise is independent of input features and true label is corrupted by either a symmetric or asymmetric noise transition matrix (details are in Section 5.1). Recently, another label noise model, named instance-dependent noise [Zhang et al., 2020; Chen et al., 2021a], is proposed, in which the noise not only depends on the class but also the input feature.

4 Our Method

4.1 Memorization Behavior

Our motivation stems from the memorization behavior of DNNs when trained with noisy labels. In Figure 1 (c), we observe that for clean samples, the model predicts them correctly with the increase of epochs. For mislabeled samples in Figure 1 (d), the model predicts the true labels correctly for...
most mislabeled samples in the early stage (high blue line), even though the model begins making incorrect predictions because of the memorization of wrong labels (increasing red line). Since the model predictions are relatively correct for both mislabeled and clean samples in the early stage, can these reliable model predictions help correct the noisy labels?

4.2 Ensemble Prediction

To alleviate the impact of noisy labels, existing work Bootstrap [Reed et al., 2015] proposes to generate soft target by interpolating between the original noisy distributions and model predictions by \( \beta \hat{y} + (1 - \beta)p \), where \( \beta \) weights the degree of interpolation. Thus the cross-entropy loss using Bootstrap becomes

\[
L_{bs} = -\frac{1}{N} \sum_{i=1}^{N} (\beta \hat{y}_i + (1 - \beta)p_i) \top \log(p_i). \tag{2}
\]

However, applying a static weight (e.g. \( \beta = 0.8 \)) to the prediction limits the correction of a hypothetical noisy label. Although another work M-correction [Arazo et al., 2019] makes \( \beta \) dynamic for different samples, the one-step correction based solely on the model predictions at the most recent training epoch still easily incurs false correction.

Since the predictions gathered in a single training epoch for correction is sub-optimal, we generate the ensemble prediction \( \hat{p} \) for each sample, aggregating the predictions over multiple previous epochs by exponential moving average. Let’s denote the model prediction in epoch \( k \) as \( p_{[k-1]} \). In epoch \( k \), we have ensemble prediction

\[
\hat{p}_{[k]} = \begin{cases} 
0 & \text{if } k = 0 \\
\alpha \hat{p}_{[k-1]} + (1 - \alpha)p_{[k]} & \text{if } k > 0 
\end{cases} \tag{3}
\]

where \( 0 \leq \alpha < 1 \) is the momentum. Based on the Eq. (3), we can derive the ensemble prediction in \( k \)-th epoch as \( \hat{p}_{[k]} = \sum_{j=1}^{k} (1 - \alpha)\alpha^{k-j}p_{[j]} \). Although ensemble prediction requires a new hyperparameter \( \alpha \) and auxiliary memory to record, it maintains a more stable and accurate prediction, especially for mislabeled samples.

4.3 Self-Ensemble Label Correction

We seek to utilize the ensemble predictions to progressively enhance the targets in loss function. There are two options to be considered.

• **Option I.** Directly use ensemble prediction as the target.

• **Option II.** Preserve the original noisy label, and combine it with ensemble prediction as the target.

The first option is widely adopted in semi-supervised learning, as the ensemble prediction learned from the labeled inputs can be used as targets for the unlabeled inputs. However, in the noisy labels setting, the model needs supervisions from noisy labels as no extra clean samples are provided. We would compare these two options in Section 5.4. In SELC, we choose the second option. Specifically, for each training sample, we initialize the soft target \( t \) using original noisy label \( \hat{y} \). Then we update \( t \) in each training epoch \( k \) by

\[
t_{[k]} = \begin{cases} 
\hat{y} & \text{if } k = 0 \\
\alpha t_{[k-1]} + (1 - \alpha)p_{[k]} & \text{if } k > 0 
\end{cases} \tag{4}
\]

Based on the Eq. (4), we rewrite above equation as

\[
t_{[k]} = \alpha^k \hat{y} + \sum_{j=1}^{k} (1 - \alpha)\alpha^{k-j}p_{[j]} \tag{5}
\]

The first term preserves the original noisy labels with exponential decaying weights \( \alpha^k \). The second term is exactly the ensemble prediction at epoch \( k \). Therefore, in training epoch \( k \), the loss of SELC becomes

\[
L_{selc} = -\frac{1}{N} \sum_{i=1}^{N} \left( \alpha^k \hat{y}_i + \sum_{j=1}^{k} (1 - \alpha)\alpha^{k-j}p_{[j]} \right) \top \log(p_i) \\
= -\frac{1}{N} \sum_{i=1}^{N} \alpha^k \hat{y}_i \top \log(p_i) \\
- \frac{1}{N} \sum_{i=1}^{N} \left( \sum_{j=1}^{k} (1 - \alpha)\alpha^{k-j}p_{[j]} \right) \top \log(p_i) \tag{6}
\]

where \( p_{[j]} \) denotes the model prediction in epoch \( j \) for input \( x_i \). The first loss term is actually the cross-entropy loss \( L_{ce} \) but weighed by \( \alpha^k \). With the increase of training epoch \( k \), \( \alpha^k \) becomes smaller. Thus \( L_{selc} \) is less and less reliant on original noisy labels. The second loss term maintains an exponential moving average of historical prediction as target, and penalizes model predictions that are inconsistent with this target. As a consequence, SELC effectively prevents memorization of mislabeled samples (low red line in Figure 1 (f)) and attains superior performance.

4.4 Estimation of Turning Point

In semi-supervised learning, the initial epoch to perform ensemble prediction is not crucial as the supervision from the clean set guides the model to predict consistent prediction throughout the training. Comparatively, in our scenario, the model would overfit to noisy labels, causing the model predictions to deteriorate. Therefore, it is essential to select the initial epoch in SELC before the turning point \( T \), at which the model starts to memorize mislabeled samples.

We can clearly observe the occurrence of turning point by monitoring the test accuracy drop (Figure 1 (b)). However, the test set is unobservable in practice. The way to accurately identify the turning point without a test set and noise information remains challenging and underexplored.

In this paper, we propose three metrics and choose the optimal one to estimate the turning point by modeling training samples’ loss distribution without requiring a clean test set. Due to the memorization behavior of DNNs, the clean samples tend to have smaller loss values than the mislabeled samples in early stage. We analyze the normalized loss distribution over different training epochs in Figure 2 top row. Intriguingly, the two distributions are merged at the initialization, then start to separate, but resume merging after the turning point. Therefore, we propose to estimate the turning point by finding the epoch that has the largest distance between two distributions. To model these two distributions, we use two unsupervised learning approaches: Gaussian Mixture Model (GMM) [Permuter et al., 2006] and K-Means.
Figure 2: We train ResNet34 on the CIFAR-10 with 60% symmetric noise using CE loss and investigate the loss distribution. Top row: The normalized loss distribution over different training epochs. Bottom row: The corresponding mixture model after fitting a two-component GMM to loss distribution. Two components gradually separate at the beginning and start to merge after the turning point (red box).

Figure 3: Three metrics on CIFAR-10 with different ratios of noise.

**Metric 1 and Metric 2.** We fit a two-component GMM to loss distribution (in Figure 2 bottom row). The probability density function (pdf) of GMM with \( M \) components on the per sample loss value \( \ell \) can be defined as \( P(\ell) = \sum_{m=1}^{M} \pi_m \mathcal{N}(\ell \mid \mu_m, \sigma_m^2) \), \( \sum_{m=1}^{M} \pi_m = 1 \), where \( \pi_m \) is the coefficient for the linear convolution of each individual pdf \( \mathcal{N}(\ell \mid \mu_m, \sigma_m^2) \). We use the Expectation-Maximization (EM) algorithm to estimate the \( \pi_m \), \( \mu_m \) and \( \sigma_m^2 \). For Metric 1, we directly calculate the distance between two components by

\[
M_1 = | \mu_1 - \mu_2 | . \tag{7}
\]

For Metric 2, we calculate the Kullback–Leibler (KL) divergence of two components as distance.

\[
M_2 = \log \frac{\sigma_2^2}{\sigma_1^2} + \frac{\sigma_1^2 + (\mu_1 - \mu_2)^2}{2\sigma_1^2} - \frac{1}{2} . \tag{8}
\]

**Metric 3.** We fit two clusters by K-Means on the loss distribution. Then we calculate the distance between two cluster centroids \( S_1 \) and \( S_2 \) as the Metric 3.

\[
M_3 = | S_1 - S_2 | . \tag{9}
\]

When we train the DNNs with noisy labels, we monitor these three metrics. Once they achieve the maximum value, the corresponding epoch is likely to be the turning point \( T \). We compare three metrics on CIFAR-10 with label noise in Figure 3, \( M_1 \) is the most reliable and stable one since its corresponding epoch of maximum value precisely aligns with the epoch when test accuracy starts to drop in Figure 1 (b) in all noise cases. We put pseudocode of SELC in Algorithm 1.

5 Experiments

This section, first, investigates the effectiveness of the proposed SELC for classification with class-conditional noise (Section 5.1), instance-dependent noise (Section 5.2) and real-world noise (Section 5.3). This is followed by several empirical analyses (Section 5.4) to shed light on SELC.

5.1 Class-conditional Label Noise

Datasets and Networks. We conduct the experiments with class-conditional label noise on CIFAR-10 and CIFAR-100 [Krizhevsky et al., 2009]. Given these two datasets are initially clean, we follow [Patrini et al., 2017] to inject noise by label transition matrix \( Q \), where \( Q_{ij} = \Pr[y_i = j \mid y_i = i] \) denotes the probability that noisy label \( y_i \) is flipped from true label \( y_i \). We evaluate SELC in two types of noise: symmetric and asymmetric. Symmetric noise is generated by replacing the labels for a percentage of the training data with all possible labels uniformly. Asymmetric noise is designed to mimic the structure of real-world label noise, where the annotators are more likely to make mistakes only within very similar classes (e.g. deer \( \rightarrow \) horse and cat \( \leftrightarrow \) dog). We use the ResNet34 [He et al., 2016] as backbone for both datasets, and train the model using SGD with a momentum of 0.9, a weight decay of 0.001, and a batch size of 128. The network is trained for 200 epochs. We set the initial learning rate as 0.02, and reduce it by a factor of 10 after 40 and 80 epochs. We fix hyperparameter \( \alpha = 0.9 \). More discussions on \( \alpha \) are in Section 5.4. Note that we do not perform early stopping since we don’t assume the presence of clean validation data.

**Algorithm 1 SELC pseudocode.**

**Input:** DNNs \( f(\theta) \), training data \( D = \{(x_i, y_i)\}_{i=1}^N \), Estimated turning point \( T \), total epoch \( T_{max} \), hyperparameter \( \alpha \)

**Output:** Optimized DNN \( f(\theta^*) \)

1: Let \( t = \hat{y} \).
2: Select an initial epoch \( T_e < T \) (e.g. \( T_e = T - 10 \)).
3: while epoch \( e < T_{max} \) do
4: if epoch \( e < T_e \) then
5: Train \( f(\theta) \) by CE loss in Eq. (1) using SGD.
6: else
7: Update \( t \) by (5).
8: Train \( f(\theta) \) by SELC loss in Eq. (6) using SGD.
9: end if
10: end while
Datasets and Networks. We follow the recent works SEAL [Chen et al., 2021a] and PLC [Zhang et al., 2020] to inject instance-dependent label noise to CIFAR. SEAL generates the controllable label noise based on the assumption that ‘hard’ (low confidence) samples are more likely to be mislabeled. PLC introduces Polynomial Margin Diminishing (PMD) noise which allows arbitrary noise strength in a wide buffer near the decision boundary. For fair comparison with SEAL, we use the same network architecture WideResNet28×10. For PMD noise, we use the same network architecture PreAct ResNet34 as PLC.

Results. Table 2 shows the results on instance-dependent label noise from SEAL. Our approach consistently achieves the best generalization performance over different noise ratios. The larger the noise ratio is, the more improvement SELC obtains. Table 3 lists the performance of different methods under three types of PMD noise at noise level 35% and 70%. We observe that the proposed method outperforms baselines across different noise settings. When the noise level is high, performances of a few baselines deteriorate and become worse than the standard (CE) approach. In contrast, the improvement of SELC is substantial (∼10% in accuracy) for the more challenging CIFAR-100 with 70% label noise.

5.3 Real-world Label Noise

Datasets and Networks. We use ANIMAL-10N [Song et al., 2019], Clothing1M [Xiao et al., 2015] and Webvision [Li et al., 2017] to evaluate the performance of SELC under the real-world label noise settings. ANIMAL-10N contains human-labeled online images for 10 animals with confusing appearance. The estimated label noise rate is 8%. Clothing1M consists of 1 million images collected from online shopping websites with labels generated from surrounding texts. The estimated label noise rate is 38.5%. WebVision contains 2.4 million images crawled from the web using the 1,000 concepts in ImageNet ILSVRC12. The estimated label noise rate is 20%. For ANIMAL-10N, we use VGG-19 with batch normalization. For Clothing1M, we use ResNet50 pretrained on ImageNet. For Webvision, we use Inception-ResNetV2. Note that all the compared methods do not use mixup to boost the performance for fair comparison.

Results. Table 4, Table 5 and Table 6 show the results on ANIMAL-10N, Clothing1M and Webvision respectively. On ANIMAL-10N and Webvision, our approach outperforms the existing baselines. On Clothing1M, SELC achieves the comparable performance to PLC, despite its simplicity.

5.4 Empirical Analysis

Correlation Accuracy. The key idea of SELC is to correct the original noisy labels. We analyze the quality of

| Dataset | CIFAR-10 | | | CIFAR-100 |
|---------|---------|-----|-----|---------|
| Class-conditional noise type | symm | asymm | symm | asymm |
| Method/Noise ratio | 20% | 40% | 60% | 80% | 20% | 40% | 60% | 80% |
| Cross Entropy | 86.98 ± 0.12 | 81.88 ± 0.29 | 74.14 ± 0.56 | 53.82 ± 1.04 | 80.11 ± 1.44 | 58.72 ± 0.26 | 48.20 ± 0.65 | 37.41 ± 0.94 |
| Bootstrap [Reed et al., 2015] | 86.23 ± 0.23 | 82.23 ± 0.37 | 75.12 ± 0.56 | 54.12 ± 1.32 | 81.21 ± 1.47 | 58.27 ± 0.21 | 47.66 ± 0.55 | 34.84 ± 1.10 |
| Forward [Patrini et al., 2017] | 87.99 ± 0.36 | 83.25 ± 0.38 | 74.96 ± 0.65 | 54.64 ± 0.44 | 83.55 ± 0.58 | 39.19 ± 0.26 | 31.05 ± 0.44 | 19.12 ± 1.95 |
| GCE [Zhang and Sabuncu, 2018] | 89.83 ± 0.20 | 87.13 ± 0.22 | 82.54 ± 0.23 | 64.07 ± 1.38 | 76.74 ± 0.61 | 68.61 ± 0.42 | 61.77 ± 0.24 | 53.16 ± 0.78 |
| Mixup [Zhang et al., 2018] | 93.58 | 89.46 | 78.32 | 66.32 | 81.66 | 69.31 | 58.12 | 41.10 |
| Joint Opt [Tanaka et al., 2018] | 92.25 | 90.79 | 86.87 | 69.16 | - | 58.15 | 54.81 | 47.94 |
| PENCIL [Yu and Wu, 2019] | - | - | - | - | 91.01 | - | 69.12 ± 0.62 | 57.70 ± 3.86 |
| NLNL [Kim et al., 2019] | 94.23 | 92.43 | 88.32 | - | - | 71.52 | 66.39 | 56.51 |
| SCE [Wang et al., 2019] | 89.83 ± 0.20 | 87.13 ± 0.22 | 82.54 ± 0.23 | 64.07 ± 1.38 | 82.51 ± 0.45 | 70.38 ± 0.13 | 62.27 ± 0.22 | 54.82 ± 0.57 |
| M-correction [Arazo et al., 2019] | 92.30 | 86.10 | 74.10 | - | - | 70.10 | 59.50 | 39.50 |
| DAC [Thulasidasan et al., 2019] | 92.91 | 90.71 | 86.30 | 74.84 | - | 73.55 | 66.92 | 57.17 |
| SELC (Ours) | - | 91.13 | - | 63.59 | - | - | 66.71 | - |
| NCE+RCE [Ma et al., 2020] | 91.16 ± 0.08 | 86.02 ± 0.09 | 79.78 ± 0.50 | 52.71 ± 1.90 | 79.59 ± 0.40 | - | 59.48 ± 0.56 | 47.12 ± 0.62 |
| ELR [Liu et al., 2020] | 94.97 ± 0.04 | 93.12 ± 0.06 | 87.25 ± 0.09 | 74.13 ± 0.14 | 91.05 ± 0.11 | 73.66 ± 0.07 | 68.46 ± 0.10 | 59.41 ± 0.06 |
| SELC+ (Ours) | 93.09 ± 0.02 | 91.18 ± 0.06 | 87.25 ± 0.09 | 74.13 ± 0.14 | 91.05 ± 0.11 | 73.66 ± 0.07 | 68.46 ± 0.10 | 59.41 ± 0.06 |

Table 1: Test accuracy (%) on CIFAR-10/100 with various ratios of class-conditional label noise injected to the training set. All methods use the same backbone ResNet34. The average accuracy and standard deviation over 3 trials are reported. The best results are in bold.
Table 3: Test accuracy (%) on CIFAR under different types of PMD noise with various levels. The average accuracy and standard deviation over 3 trials are reported. All above methods do not use mixup to boost the performance for fair comparison. The best results are in bold.

| Noise Ratio | Type-I (35%) | Type-I (70%) | Type-II (35%) | Type-II (70%) | Type-III (35%) | Type-III (70%) |
|-------------|--------------|--------------|--------------|--------------|--------------|--------------|
| CIFAR-10    | 78.11 ± 0.74 | 79.97 ± 0.15 | 80.65 ± 0.59 | 79.76 ± 0.72 | 80.98 ± 0.80 | 82.80 ± 0.27 |
| CIFAR-100   | 66.41 ± 4.16 | 45.57 ± 8.77 | 49.85 ± 4.53 | 41.52 ± 4.53 | 42.74 ± 2.14 | 86.97 ± 0.15 |

Table 4: The accuracy (%) results on ANIMAL-10N.

| Method   | Accuracy   |
|----------|------------|
| Cross Entropy | 79.40 ± 0.14 |
| Nested [Chen et al., 2021b] | 81.30 ± 0.60 |
| SELFIE [Song et al., 2019] | 81.80 ± 0.09 |
| PLC [Zhang et al., 2020] | 83.40 ± 0.43 |
| SELC (ours) | 83.73 ± 0.06 |

Table 5: The accuracy (%) results on Clothing1M.

| Method   | Accuracy   |
|----------|------------|
| Cross Entropy | 68.94 |
| Forward [Patrini et al., 2017] | 69.84 |
| SEAL [Chen et al., 2021a] | 70.63 |
| SCE [Wang et al., 2019] | 71.02 |
| LRT [Zhang et al., 2020] | 71.74 |
| DMI [Xu et al., 2019] | 72.27 |
| ELR [Liu et al., 2020] | 72.87 |
| Nested [Chen et al., 2021b] | 73.10 |
| PENCIL [Yi and Wu, 2019] | 73.49 |
| PLC [Zhang et al., 2020] | 74.02 |
| SELC (ours) | 74.01 |

The new target \( t \) by calculating its correction accuracy: 
\[
\frac{1}{N} \sum_{i=1}^{N} \{ \text{arg max } y_{t} = \text{arg max } x_{t} \},
\]
where \( y_{t} \) is the true label of \( x_{t} \). Figure 4 (a) shows the correction accuracy of the Option I and Option II (SELC). We observe SELC achieves higher accuracy than Option I and correction accuracy is stable with the increase of training epochs. Figure 5 shows the confusion matrix of corrected labels w.r.t. the true labels on CIFAR-10 with 40% symmetric label noise. SELC corrects the noisy labels impressively well for all classes.
References

[Arazo et al., 2019] Eric Arazo, Diego Ortego, Paul Albert, Noel O’Connor, and Kevin Mcguinness. Unsupervised label noise modeling and loss correction. In ICML, pages 312–321, 2019.

[Arpit et al., 2017] Devansh Arpit, Stanislaw K Jastrzebski, Nicolas Ballas, David Krueger, Emmanuel Bengio, Maxinder S Kanwal, Tegan Maharaj, Asja Fischer, Aaron C Courville, Yoshua Bengio, et al. A closer look at memorization in deep networks. In ICML, 2017.

[Chen et al., 2019] Pengfei Chen, Ben Ben Liao, Guangyong Chen, and Shengyu Zhang. Understanding and utilizing deep neural networks trained with noisy labels. In ICML, pages 1062–1070, 2019.

[Chen et al., 2021a] Pengfei Chen, Junjie Ye, Guangyong Chen, Jingwei Zhao, and Pheng-Ann Heng. Beyond class-conditional assumption: A primary attempt to combat instance-dependent label noise. In AAAI, volume 35, pages 11442–11450, 2021.

[Chen et al., 2021b] Yingyi Chen, Xi Shen, Shell Xu Hu, and Johan Ak Suykens. Boosting co-teaching with compression regularization for label noise. In CVPR, pages 2688–2692, 2021.

[Gui et al., 2020] Xian-Jin Gui, Wei Wang, and Zhang-Hao Tian. Towards understanding deep learning from noisy labels with small-loss criterion. IJCAI, 2021.

[Han et al., 2018] Bo Han, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor Tsang, and Masashi Sugiyama. Co-teaching: Robust training of deep neural networks with extremely noisy labels. In NIPS, pages 8527–8537, 2018.

[He et al., 2016] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In CVPR, pages 770–778, 2016.

[Kim et al., 2019] Youngdong Kim, Junho Yim, Juseung Yun, and Junmo Kim. Nln: Negative learning for noisy labels. In ICCV, pages 101–110, 2019.

[Krizhevsky et al., 2009] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[Li et al., 2017] Wenhao Li, Minim Wang, Wei Li, Eirikur Agustsson, and Luc Van Gool. Webvision database: Visual learning and understanding from web data. arXiv, 2017.

[Liu et al., 2020] Sheng Liu, Jonathan Niles-Weed, Narges Razavian, and Carlos Fernandez-Granda. Early-learning regularization prevents memorization of noisy labels. NIPS, 33, 2020.

[Lu et al., 2021] Yangdi Lu, Yang Bo, and Wenbo He. Confidence adaptive regularization for deep learning with noisy labels. arXiv preprint arXiv:2108.08212, 2021.

[Lu et al., 2022] Yangdi Lu, Yang Bo, and Wenbo He. An ensemble model for combating label noise. In Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining, pages 608–617, 2022.

[Ma et al., 2020] Xingjun Ma, Hanxun Huang, Yisen Wang, Simone Romano, Sarah Erfani, and James Bailey. Normalized loss functions for deep learning with noisy labels. In In ICML, pages 6543–6553. PMLR, 2020.

[Mirzasoleiman et al., 2020] Baharan Mirzasoleiman, Kaidi Cao, and Jure Leskovec. Coresets for robust training of neural networks against noisy labels. NIPS, 2020.

[Natarajan et al., 2013] Nagarajan Natarajan, Inderjit S Dhillon, Pradeep K Ravikumar, and Ambuj Tewari. Learning with noisy labels. NIPS, 26:1196–1204, 2013.

[Nguyen et al., 2020] Tam Nguyen, C Mumme, T Ngo, L Beggel, and Thomas Brox. Self: learning to filter noisy labels with self-ensembling. In ICLR, 2020.

[Patrini et al., 2017] Giorgio Patrini, Alessandro Rozza, Aditya Krishna Menon, Richard Nock, and Lizhen Qu. Making deep neural networks robust to label noise: A loss correction approach. In CVPR, pages 1944–1952, 2017.

[Permuter et al., 2006] Haim Permuter, Joseph Francos, and Ian Jermyn. A study of gaussian mixture models of color and texture features for image classification and segmentation. Pattern recognition, 39(4):695–706, 2006.

[Reed et al., 2015] Scott E Reed, Honglak Lee, Dragomir Anguelov, Christian Szegedy, Dumitru Erhan, and Andrew Rabinovich. Training deep neural networks on noisy labels with bootstrapping. In ICLR (Workshop), 2015.

[Song et al., 2019] Hwanjun Song, Minseok Kim, and Jae-Gil Lee. Selfie: Refurbishing unclean samples for robust deep learning. In ICML, pages 5907–5915, 2019.

[Tanaka et al., 2018] Daiki Tanaka, Daiki Ikami, Toshihiko Yamasaki, and Kiyoharu Aizawa. Joint optimization framework for learning with noisy labels. In CVPR, pages 5552–5560, 2018.

[Thulasidasan et al., 2019] Sunil Thulasidasan, Tannoy Bhatcharya, Jeff Bilmes, Gopinath Chennupati, and Jamal Mohd-Yusof. Combating label noise in deep learning using abstinence. In ICLR, pages 6234–6243. PMLR, 2019.

[Wang et al., 2019] Yisen Wang, Xinjun Ma, Zaiyi Chen, Yuan Luo, Jinfeng Yi, and James Bailey. Symmetric cross entropy for robust learning with noisy labels. In ICCV, pages 322–330, 2019.

[Xiao et al., 2015] Tong Xiao, Tian Xia, Yi Yang, Chang Huang, and Xiaogang Wang. Learning from massive noisy labeled data for image classification. In CVPR, pages 2691–2699, 2015.

[Xu et al., 2019] Yilun Xu, Peng Cao, Yuqing Kong, and Yizhou Wang. L_dni: A novel information-theoretic loss function for training deep nets robust to label noise. Advances in Neural Information Processing Systems, 32:6225–6236, 2019.

[Yi and Wu, 2019] Kun Yi and Jianxin Wu. Probabilistic end-to-end noise correction for learning with noisy labels. In CVPR, pages 7017–7025, 2019.

[Yu et al., 2019] Xingrui Yu, Bo Han, Jiangchao Yao, Gang Niu, IW Tsang, and Masashi Sugiyama. How does disagreement help generalization against label corruption? In ICML, 2019.

[Zhang and Sabuncu, 2018] Zhilu Zhang and Mert Sabuncu. Generalized cross entropy loss for training deep neural networks with noisy labels. In NIPS, pages 8778–8788, 2018.

[Zhang et al., 2018] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In ICLR, 2018.

[Zhang et al., 2020] Yikai Zhang, Songzhu Zheng, Pengxiang Wu, Mayank Goswami, and Chao Chen. Learning with feature-dependent label noise: A progressive approach. In International Conference on Learning Representations, 2020.

[Zhang et al., 2021] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning (still) requires rethinking generalization. Communications of the ACM, 64(3):107–115, 2021.

[Zheng et al., 2020] Songzhu Zheng, Pengxiang Wu, Aman Goswami, Mayank Goswami, Dimitris Metaxas, and Chao Chen. Error-bounded correction of noisy labels. In ICML, pages 11447–11457. PMLR, 2020.