Rethinking Label Refurbishment: Model Robustness under Label Noise

Yangdi Lu, Zhiwei Xu, Wenbo He

McMaster University, Department of Computing and Software, Canada
{luy100, xuz131, hew11}@mcmaster.ca

Abstract

A family of methods that generate soft labels by mixing the hard labels with a certain distribution, namely *label refurbishment*, are widely used to train deep neural networks. However, some of these methods are still poorly understood in the presence of label noise. In this paper, we revisit four label refurbishment methods and reveal the strong connection between them. We find that they affect the neural network models in different manners. Two of them smooth the estimated posterior for regularization effects, and the other two force the model to produce high-confidence predictions. We conduct extensive experiments to evaluate related methods and observe that both effects improve the model generalization under label noise. Furthermore, we theoretically show that both effects lead to generalization guarantees on the clean distribution despite being trained with noisy labels.

Introduction

In supervised learning tasks, we always expect and assume a large amount of correct-annotated training data. However, noisy labels are inevitably introduced in real-world datasets collected from crowdsourcing or automatic labeling systems. Zhang et al. (2018) have demonstrated that deep neural networks end up memorizing noisy labels and lead to poor generalization. Therefore, it is essential to develop techniques for learning with noisy labels.

Numerous approaches have been proposed to improve robustness of deep neural networks, wherein a family of label refurbishment methods generate soft targets by mixing the hard labels with a certain distribution, are of great attraction. Intuitively, label refurbishment methods can mitigate the influence of label noise, as a wrong label after refurbishing is likely to be corrected or less "poisonous" to the networks. For example, label smoothing (LS) (Szegedy et al. 2016) utilizes soft labels by taking a positively weighted average between the hard training labels and the uniform distribution. Take image classification as an example. Suppose we have three classes (i.e. dog, cat, monkey) and a dog image mislabeled as cat with one-hot noisy label $[0, 1, 0]$. After applying LS with smoothing rate $\alpha \in [0, 1)$, the refurbished label $[\alpha/3, 1-2\alpha/3, \alpha/3]$ contributes less misleading information but still retains the maximal probability on assigned label. LS has been widely studied for improving the model performance (Szegedy et al. 2016; Vaswani et al. 2017), improving calibration (Müller, Kornblith, and Hinton 2019), and its close relationship with knowledge distillation (Yuan et al. 2020; Shen et al. 2020). Empirical studies (Lukasik et al. 2020; Zhang et al. 2021; Wei et al. 2022) have demonstrated the effectiveness of LS in improving the model performance in the presence of noisy labels.

Although label refurbishment methods are commonly adopted as "add-on tricks" to boost the classification performance in the label noise literature (Ma et al. 2018; Arazo et al. 2019; Li, Socher, and Hoi 2020; Zhou, Wang, and Bilmes 2020), the individual performance gains of these methods have hardly been evaluated, and their corresponding theoretical guarantees remain underexplored.

In this paper, we conduct an in-depth study of four label refurbishment methods, including bootstrapping loss (BL) (Reed et al. 2015), label smoothing (LS) (Szegedy et al. 2016), back correction (BC) (Patrini et al. 2017), and energy regularized loss (ERL) (Pereyra et al. 2017). Specifically, we first generalize these methods by introducing a transformation matrix, which helps to reveal the intrinsic relationship between them. By investigating their effects on loss functions, we discover that two of them (BC and BL), proposed to improve model robustness against label noise, force the model to produce high-confidence predictions, while the other two (LS and ERL), proposed for regularization under clean labels, have the opposite effect. We wonder if we can reverse the effect of LS and ERL by using a negative hyperparameter (i.e. α), so that LS and ERL will yield high-confidence predictions as BC and BL. By evaluating their classification performance under label noise, we observe two interesting phenomena.

1. *The LS achieves superior performance when the smoothing rate α is close to 1.0 (e.g. $\alpha = 0.9$), referred to in this paper as "extreme" label smoothing (ELS).*
2. *When the methods restrict the model to produce high-confidence predictions, their classification performance under label noise is substantially improved, including BL, BC, reversed LS and reversed ERL.*

To explain the first phenomenon, we find that training with ELS restricts the network parameters close to its initialization. Based on the theory of neural tangent kernel (Jacot,

Gabriel, and Hongler 2018), we theoretically prove that ELS is robust to label noise when using wide neural networks as classifiers. For the second phenomenon, we prove that an extremely confident model implicitly makes the loss function robust to label noise, which bridges the gap between the label refurbishment methods and noise-robust loss functions (Ghosh, Kumar, and Sastry 2017). We also conduct extensive experiments to support our findings, shedding light upon the robustness of deep neural networks in the presence of label noise.

Preliminaries

Consider the C-class classification problem, we denote clean training set by $D = \{(\boldsymbol{x}^{[i]}, y^{[i]})\}_{i=1}^{N} \backsim \mathbb{P}(\boldsymbol{x}, y)$, where $\boldsymbol{x}^{[i]}$ is an input and $y^{[i]} \in [C] = \{1, \ldots, C\}$ is label. In the noisy label scenario, the clean training set is unobservable. We only have a noisy training set $\hat{D} = \{(\boldsymbol{x}^{[i]}, \hat{y}^{[i]})\}_{i=1}^{N}$ from noisy distribution $\hat{\mathbb{P}}(\boldsymbol{x}, \hat{y})$, i.e., the observed labels are not reflective of the ground truth labels. As the generation of real-world label noise is unknown, a common methodology to cope with noisy labels is to posit noise assumptions. A typical noise assumption is class-conditional noise (Natarajan et al. 2013), wherein the true label is corrupted by a noise transition matrix $\mathbf{T} \in [0,1]^{C\times C}$ and $\mathbf{T}_{ij} = \Pr(\hat{y} = j | y = i)$ is the probability of the true label i being flipped into a noisy label j. Given the noise rate η, *symmetric* noise further assumes that flip probability to other labels is constant, i.e., $\forall_{i=j}\mathbf{T}_{ij} = 1-\eta \wedge \forall_{i\neq j}\mathbf{T}_{ij} = \frac{\eta}{C-1}$. In contrast, *asymmetric* noise assumes the flip probability can be various for different classes, i.e., $\forall_{i=j}\mathbf{T}_{ij} = 1-\eta \wedge \exists_{i\neq j, i\neq k, j\neq k}\mathbf{T}_{ij} > \mathbf{T}_{ik}$. A classifier f maps an input $\boldsymbol{x}$ to a softmaxed C-dimensional prediction probability distribution $f(\boldsymbol{\Theta}, \boldsymbol{x})$. Let $\boldsymbol{\Theta}$ denotes the parameters and $\ell(y, f(\boldsymbol{\Theta}, \boldsymbol{x})) : [C] \times \mathbb{R}^C \to \mathbb{R}^+$ denotes the loss function. Suppose we train with cross entropy (CE) loss on the noisy data $\hat{D}$, and let $\boldsymbol{p} = f(\boldsymbol{\Theta}, \boldsymbol{x})$, we have

$$\begin{aligned}\mathcal{L}_{\text{ce}}(\boldsymbol{\Theta}) &= \frac{1}{N}\sum_{i=1}^{N} \ell_{\text{ce}}(\hat{y}^{[i]}, f(\boldsymbol{\Theta}, \boldsymbol{x}^{[i]})) \\ &= -\frac{1}{N}\sum_{i=1}^{N} (\hat{\boldsymbol{y}}^{[i]})^\top \log(\boldsymbol{p}^{[i]}), \end{aligned} \tag{1}$$

where $\hat{\boldsymbol{y}}^{[i]} \in \{0,1\}^C$ is one-hot vector of noisy label $\hat{y}^{[i]}$. When directly optimizing Eq. (1) by gradient descent, it was observed that the deep neural networks fit the training data including the samples with wrong labels, resulting in performance degradation (Zhang et al. 2018).

Generalization of Label Refurbishment

Many methods have been proposed to prevent the model from becoming over-confident or to improve the model generalization, including a few approaches are specifically designed to combat noisy labels. We can simply introduce a transformation matrix $\mathbf{M} \in \mathbb{R}^{C\times C}$ to scale the hard label vector $\hat{\boldsymbol{y}}$ to refurbished soft label $\boldsymbol{t}$. Then we have $\boldsymbol{t} = \mathbf{M}\cdot\hat{\boldsymbol{y}}$ and different $\mathbf{M}$ leads to different forms of soft label $\boldsymbol{t}$. Now the refurbished cross entropy loss becomes

Type	Method	Matrix $\mathbf{M}$	Prediction confidence
i	LS	$(1-\alpha)\mathbf{I} + \frac{\alpha}{C}\mathbf{J}$	$\searrow$
	BC	$\frac{1}{1-\alpha}(\mathbf{I} - \frac{\alpha}{C}\cdot\mathbf{J})$	$\nearrow$
ii	BL	$(1-\alpha)\mathbf{I} + \alpha\boldsymbol{p}\cdot\boldsymbol{j}^\top$	$\nearrow$
	ERL	$\mathbf{I} - \alpha\boldsymbol{p}\cdot\boldsymbol{j}^\top$	$\searrow$

Table 1: Comparison of related methods. Let bold I denotes the $C\times C$ identity matrix, bold J denotes the $C\times C$ all-ones matrix and $\boldsymbol{j}$ denotes the $C\times 1$ all-ones vector. $\nearrow$ means increase and $\searrow$ means decrease.

$$\mathcal{L}_{\text{ce}}^{\mathbf{M}}(\boldsymbol{\Theta}) = -\frac{1}{N}\sum_{i=1}^{N} (\mathbf{M}\cdot\hat{\boldsymbol{y}}^{[i]})^\top \log(\boldsymbol{p}^{[i]}). \tag{2}$$

Compared to the loss in Eq. (1), we now potentially involve all entries in noisy labels, scaled appropriately by the transformation matrix $\mathbf{M}$.

Existing Label Refurbishment Methods

Baseline. As a baseline method, it directly trains the deep neural networks with cross entropy loss using noisy labels. Thus, the transformation matrix $\mathbf{M} = \mathbf{I}$, where $\mathbf{I}$ denotes a $C\times C$ identity matrix.

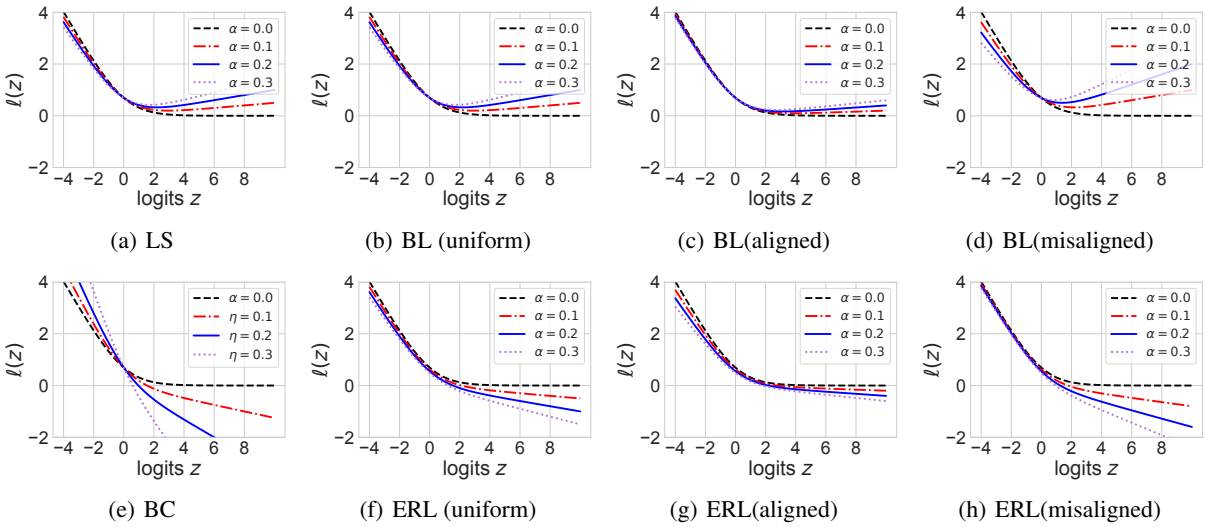
Bootstrapping Loss (BL). Reed et al. (2015) propose the perceptually-consistent training by adding a perceptual term (i.e. current prediction of the model) to noisy labels. So the refurbished label $\boldsymbol{t} = (1-\alpha)\hat{\boldsymbol{y}} + \alpha\boldsymbol{p}$, where $\alpha \in [0,1)$ is hyperparameter.

Label Smoothing (LS). Label smoothing (Szegedy et al. 2016) mixes the hard labels with a uniform distribution over all possible labels by given smooth rate $\alpha \in [0,1)$. The soft label $\boldsymbol{t} = (1-\alpha)\hat{\boldsymbol{y}} + \frac{\alpha}{C}$. It was empirically shown that LS prevents the network from producing over-confident predictions (Müller, Kornblith, and Hinton 2019).

Backward Correction (BC). Suppose the label transition matrix $\mathbf{T}$ is either known or being estimated, backward correction (Patrini et al. 2017) transforms the noisy labels by an estimated $\mathbf{T}^{-1}$. It theoretically recovers the noisy class probability back to the clean one. The main drawback of them is that they need to know the noise type and noise ratio, which makes it impractical in real life. For class-conditional label noise, the estimated $\mathbf{T}^{-1} = \frac{1}{1-\alpha}(\mathbf{I} - \frac{\alpha}{C}\cdot\mathbf{J})$, where $\alpha = \frac{C}{C-1}\cdot\eta$ is theoretical optimal choice in original paper.

Energy Regularized Loss (ERL). Pereyra et al. (2017) aim to prevent the neural networks from being over-confident by penalizing low-entropy (confident) distributions. The entropy of a prediction (conditional distribution) is calculated by $\mathcal{H}(\boldsymbol{p}) = -\boldsymbol{p}^\top\log(\boldsymbol{p})$. ERL adds the negative entropy to the loss. So the refurbished soft label is $\boldsymbol{t} = \hat{\boldsymbol{y}} - \alpha\boldsymbol{p}$ for cross entropy loss, where $\alpha \geq 0$ controls the strength of the confidence penalty.

In this paper, we focus on the above four methods. Note that there are variants based on label refurbishment (Arazo et al. 2019; Li, Dasarathy, and Berisha 2020; Lu and He 2022), but their core ideas are similar. We summarize the transformation matrix $\mathbf{M}$ of four methods in Table 1. They



(a) LS (b) BL (uniform) (c) BL(aligned) (d) BL(misaligned)

(e) BC (f) ERL (uniform) (g) ERL(aligned) (h) ERL(misaligned)

Figure 1: Effect of LS, BL, BC and ERL on logistic loss ($C = 2$). (a) LS introduces a finite positive minima. (b) BL with uniform prediction (i.e. $p = \frac{1}{C}\boldsymbol{j}$) performs exactly the same effect as LS. (c-d) BL weakens/reinforces its effect if prediction is aligned/misaligned with noisy label. (e) BC makes the loss negative for large positive logits. (f) ERL performs the similar effect as BC with uniform prediction. (g-h) ERL weakens/reinforces its effect if prediction is aligned/misaligned with noisy label.

can be divided into two categories. The first category, including LS and BC, linearly combines an identity matrix $\mathbf{I}$ with an all-ones matrix $\mathbf{J}$ and treats all samples equally. Thus the label with maximal probability is preserved as long as $\alpha < 1$. Another category combines an identity matrix $\mathbf{I}$ with all-ones vector $\boldsymbol{j}$ scaled by current prediction $\boldsymbol{p}$. In this case, the label with maximal probability can be potentially modified if the current prediction $\boldsymbol{p}$ is not aligned with the given label, resulting in a label correction effect.

The Effect on Loss and Model Confidence

Let $\ell_{\text{ce}}(y, \boldsymbol{p})$ denotes the cross entropy loss for a sample $(\boldsymbol{x}, y)$ and $\mathcal{H}(\boldsymbol{p})$ denotes the entropy of a prediction $\boldsymbol{p}$. Based on the different $\mathbf{M}$, we have the refurbished loss functions as follows:

$$\ell_{\text{ce}}^{\text{BL}}(y, \boldsymbol{p}) \propto \ell_{\text{ce}}(y, \boldsymbol{p}) + \frac{\alpha}{1-\alpha} \cdot \mathcal{H}(\boldsymbol{p}); \tag{3}$$

$$\ell_{\text{ce}}^{\text{BC}}(y, \boldsymbol{p}) \propto \ell_{\text{ce}}(y, \boldsymbol{p}) - \frac{\alpha}{C} \cdot \sum_{c=1}^{C} \ell_{\text{ce}}(c, \boldsymbol{p}); \tag{4}$$

$$\ell_{\text{ce}}^{\text{LS}}(y, \boldsymbol{p}) \propto \ell_{\text{ce}}(y, \boldsymbol{p}) + \frac{\alpha}{(1-\alpha) \cdot C} \cdot \sum_{c=1}^{C} \ell_{\text{ce}}(c, \boldsymbol{p}); \tag{5}$$

$$\ell_{\text{ce}}^{\text{ERL}}(y, \boldsymbol{p}) \propto \ell_{\text{ce}}(y, \boldsymbol{p}) - \alpha \cdot \mathcal{H}(\boldsymbol{p}). \tag{6}$$

Figure 1 depicts the effect of these methods on the logistic loss when $C = 2$. The standard logistic loss ($\alpha = 0.0$) vanishes for large positive logits while performing almost linearly for large negative logits. Both LS and BL introduce finite positive minima, while BC and ERL guarantee an unbiased risk estimate, allowing for a negative loss on positive samples that are correctly predicted.

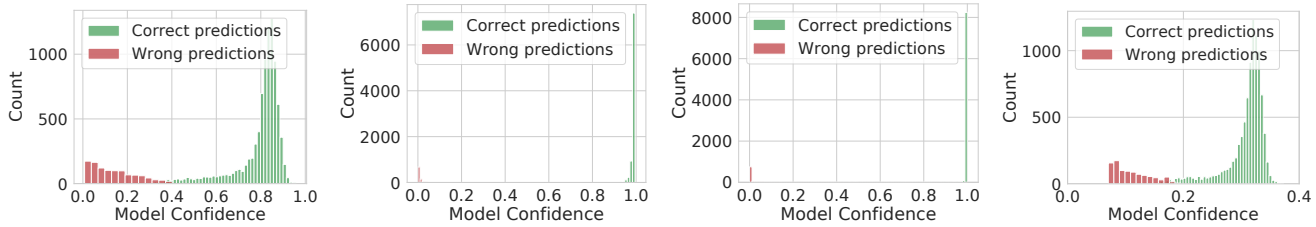
Base on the derived loss functions, we observe that the effects of these methods on model confidence are fundamentally different: BL aims to minimize the entropy of prediction while ERL seeks to maximize it. LS aims to minimize the average per-class loss $\frac{1}{C}\sum_{c=1}^{C} \ell(c, \boldsymbol{p})$, while BC seeks to maximize it. In other words, both BL and BC tend to push the model become overly confident early in the training (see Figure 2(b) and 2(c)), while LS and ERL seek to penalize confident predictions (see Figure 2(d) and 2(e)).

Since BL and BC are proposed to improve robustness under label noise, we now return to our original question: can we reverse the effect of LS and ERL? To achieve it, we extend the hyperparameter α of LS and ERL to negative values. As shown in Figure 2(f) and 2(g), both of the reversed LS and ERL (with $\alpha = -0.7$) make the model become over-confident on correct predictions and under-confident on wrong predictions. In the next section, we empirically assess whether this effect improves model robustness to label noise.

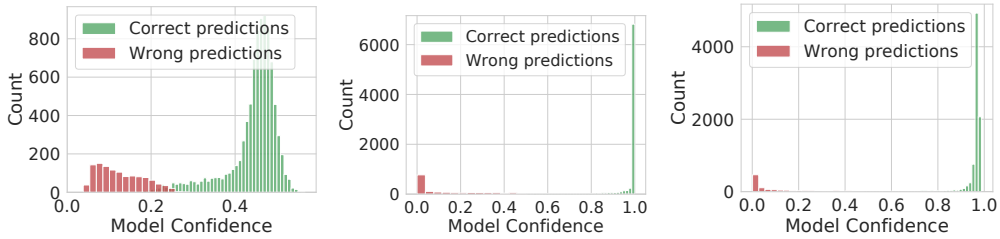
Classification Performance

In this section, we re-evaluate the label refurbishment methods to see the classification performance in the noisy label scenario. We test these methods on two benchmark datasets MNIST (LeCun et al. 1998) and CIFAR-10 (Krizhevsky et al. 2009) with class-conditional label noise.

Experimental Setups. We corrupt MNIST and CIFAR-10 by noise transition matrix $\mathbf{T}$. As mentioned in Preliminaries, $\mathbf{T}$ has two representative noise assumptions (Patrini et al. 2017). 1) Symmetric label noise is generated by uniformly flipping the label to one of the other class label; 2) Asymmetric label noise is a simulation of fine-grained classification with noisy labels in the real world, where the mistakes only occur within very similar classes (e.g. dog $\leftrightarrow$ cat). We implement all methods using Pytorch and train them on a NVIDIA A100 GPU. We use a 4-layer CNN for MNIST and

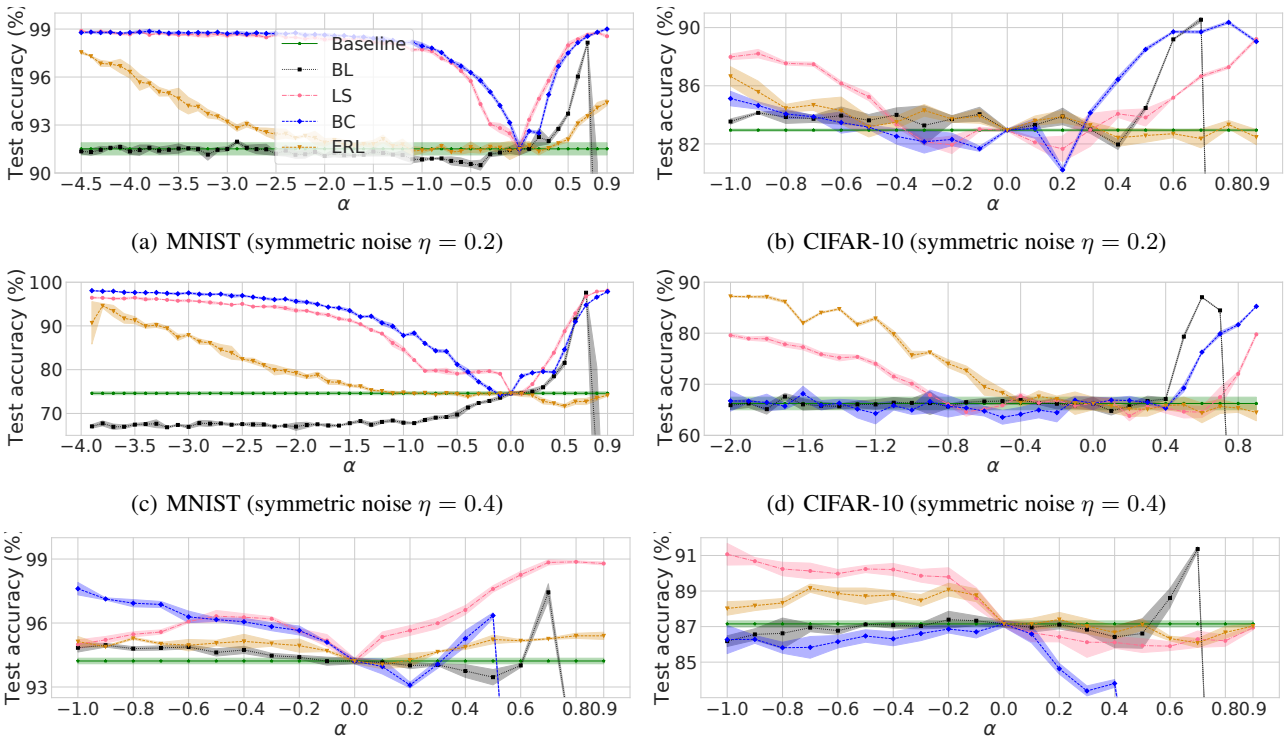


(a) CE (b) BL ($\alpha = 0.7$) (c) BC ($\alpha = 0.7$) (d) LS ($\alpha = 0.7$)



(e) ERL ($\alpha = 0.7$) (f) Reversed LS ($\alpha = -0.7$) (g) Reversed ERL ($\alpha = -0.7$)

Figure 2: Model confidence distribution of correct and wrong predictions on CIFAR-10 test data. All models are trained on CIFAR-10 with symmetric noise $\eta = 0.2$.



(a) MNIST (symmetric noise $\eta = 0.2$) (b) CIFAR-10 (symmetric noise $\eta = 0.2$)

(c) MNIST (symmetric noise $\eta = 0.4$) (d) CIFAR-10 (symmetric noise $\eta = 0.4$)

(e) MNIST (asymmetric noise $\eta = 0.2$) (f) CIFAR-10 (asymmetric noise $\eta = 0.2$)

Figure 3: Test accuracies of compared methods with different α on MNIST and CIFAR-10. For each noise case, we run 3 times with random seeds to get mean accuracy and the corresponding standard deviations (denoted by the shaped regions).

ResNet34 (He et al. 2016) for CIFAR-10 as backbones. For hyperparameter, all compared methods contain a parameter α to adjust the strength of refurbishment. Although in the label noise literature, it is customary to estimate the optimal α. For example, the theoretical optimal α for BC equals $\frac{C}{C-1} \cdot \eta$. However, here we simply treat all α as a tuning parameter. For reversed LS and reversed ELR, we use negative α to test their performance.

Results. Figure 3 reports the results on MNIST and CIFAR-10 with symmetric and asymmetric noise. We observe that

label refurbishment methods can significantly improve performance over the baseline with an appropriate α. Both BL and BC achieve impressive performance under symmetric label noise, while excessive correction of BL on noisy labels (e.g. $\alpha > 0.7$) leads to performance degradation as soft targets end with a delusional agent, resulting in underfitting. Besides, the original BL paper (Reed et al. 2015) suggests that the optimal choice of α should be 0.05, while we find that the optimal α is from 0.6 to 0.7 in our setting.

Our second observation is that, the optimal α for LS is extremely large (i.e. 0.8, 0.9) on MNIST and CIFAR-10 with symmetric label noise. This phenomenon is considered counterintuitive. Here we call LS with α close to 1.0 *extreme label smoothing* (ELS) and deeply investigate this phenomenon theoretically in next section. Besides, choosing $\alpha \gg \eta$ improves performance for BC. It is totally in contrast to the theoretically optimal choice as noted in (Patrini et al. 2017). Therefore, we suggest that it is valuable and more practical to treat α as a tuning hyperparameter.

For reversed LS and reversed ERL (with negative α), we observe that they also significantly outperform baseline with most α, which empirically validates our hypothesis that forcing the models to produce high-confidence predictions helps improve robustness under label noise.

Theoretical Analysis

In this section, we provide theoretical analyses to answer the following two questions: (1) How does ELS specifically improve the classification performance under symmetric label noise? (2) How does a model with high-confidence predictions achieve robustness to label noise?

Regularization Effect of Extreme Label Smoothing

Prior work (Lukasik et al. 2020) has demonstrated that LS has a similar effect to an explicit L2 regularization in a linear model. Given a linear model $f(\mathbf{\Theta}, \boldsymbol{x}) = \mathbf{\Theta}\boldsymbol{x}$, trained on features $\mathbf{X} \in \mathbb{R}^{N\times d}$ and one-hot labels $\mathbf{Y} \in \{0,1\}^{N\times C}$ using the l_2 loss, i.e., $\min_{\mathbf{\Theta}} ||\mathbf{X}\mathbf{\Theta} - \mathbf{Y}||_2^2$. Then LS at level α transforms the optimal solution $\mathbf{\Theta}^*$ to $\hat{\mathbf{\Theta}}^* = (1-\alpha)\cdot \mathbf{\Theta}^* + \frac{\alpha}{C}\cdot(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{J}$. LS encourages shrinkage of the parameters towards zero, which is similar to L2 regularization. However, it requires a strong assumption that the matrix $\mathbf{X}^\top\mathbf{X}$ has an inverse (i.e. data is centered). Otherwise, we may use gradient-based strategy, gradually updating the parameters to reach the optimum. Intuitively, $\alpha \to 1$ makes the smoothed distribution close to a uniform distribution, which dramatically slows down the learning as the smoothed labels are less informative. Consequently, the network parameters would stay close to the initialization. In contrast, $\alpha = 0$ recovers the label smoothing to regular training. Therefore, we speculate that ELS has an effect to constrain the network parameters close to the initialization, rather than zero.

To verify our hypothesis, we adopt a special regularization called distance to initialization (DTI) (Hu, Li, and Yu 2019) for comparison. We denote ℓ_2 loss [1] by $\mathcal{L}(\mathbf{\Theta}) = \frac{1}{2}\sum_{i=1}^{N}(f(\mathbf{\Theta}, \boldsymbol{x}^{[i]}) - \hat{\boldsymbol{y}}^{[i]})^2$, initial parameters by $\mathbf{\Theta}_0$ and parameters movement by $\mathbf{\Theta}_\Delta = \mathbf{\Theta} - \mathbf{\Theta}_0$. Then the ELS and DTI regularization minimize the following objectives:

$$\mathcal{L}^{\text{ELS}}(\mathbf{\Theta}) = \frac{1}{2}\sum_{i=1}^{N}\big(f(\mathbf{\Theta}, \boldsymbol{x}^{[i]}) - (1-\alpha)\hat{\boldsymbol{y}}^{[i]} - \frac{\alpha}{C}\big)^2.$$

$$\mathcal{L}^{\text{DTI}}(\mathbf{\Theta}) = \frac{1}{2}\sum_{i=1}^{N}\big(f(\mathbf{\Theta}, \boldsymbol{x}^{[i]}) - \hat{\boldsymbol{y}}^{[i]}\big)^2 + \frac{\lambda^2}{2}\|\mathbf{\Theta}_\Delta\|^2.$$

Then, we conduct experiments to investigate the network parameters by measuring three metrics, including Frobenius norm of current parameters $\|\mathbf{\Theta}\|_F$, the parameters movement $\|\mathbf{\Theta}_\Delta\|_F$ and its corresponding logarithm term $\lg(\|\mathbf{\Theta}_\Delta\|_F)$. In Figure 4(a), we observe that by using L2 regularization, the corresponding $\|\mathbf{\Theta}\|_F$ shrinks fast as expected. In contrast, $\|\mathbf{\Theta}\|_F$ of both ELS and DTI regularization hardly change. In Figure 4(b), we find that a larger α reinforces the restriction on parameters movement. In Figure 4(c), we observe that ELS has the similar $\lg(\|\mathbf{\Theta}_\Delta\|_F)$ with DTI regularization during training. Therefore, similar to DTI regularization, ELS has the effect of constraining network parameters to be close to the initialization.

Relation to Kernel Ridge Regression To establish the connection of ELS with kernel ridge regression in wide neural networks, we briefly recap the theory of neural tangent kernel (NTK) (Jacot, Gabriel, and Hongler 2018; Arora et al. 2019), which builds the equivalence between training a wide neural network and a kernel method.

Suppose a neural network is trained by minimizing the l_2 loss over training set. It was shown that if the network is sufficiently wide and the parameters $\mathbf{\Theta}$ stay close to the initialization $\mathbf{\Theta}_0$ during training, the network can be effectively approximated by its first-order Taylor expansion with respect to its parameters at initialization. Thus, we have the following approximation accurate in NTK regime.

$$f(\mathbf{\Theta}, \boldsymbol{x}) \approx f(\mathbf{\Theta}_0, \boldsymbol{x}) + \langle \nabla_{\mathbf{\Theta}} f(\mathbf{\Theta}_0, \boldsymbol{x}), \mathbf{\Theta}_\Delta\rangle. \quad (7)$$

This approximation is exact in the infinite width limit, but can also be demonstrated when the width is sufficiently large. We have $\phi(\boldsymbol{x}) = \nabla_{\mathbf{\Theta}} f(\mathbf{\Theta}_0, \boldsymbol{x})$ which induces the NTK $k(\boldsymbol{x}, \boldsymbol{x}') = \langle \phi(\boldsymbol{x}), \phi(\boldsymbol{x}')\rangle$. Then we obtain approximation $f(\mathbf{\Theta}, \boldsymbol{x}) \approx f(\mathbf{\Theta}_0, \boldsymbol{x}) + \phi(\boldsymbol{x})^\top\mathbf{\Theta}_\Delta$. Assume we have near-zero initial output: $f(\mathbf{\Theta}_0, \boldsymbol{x}) \approx 0$ [2], then

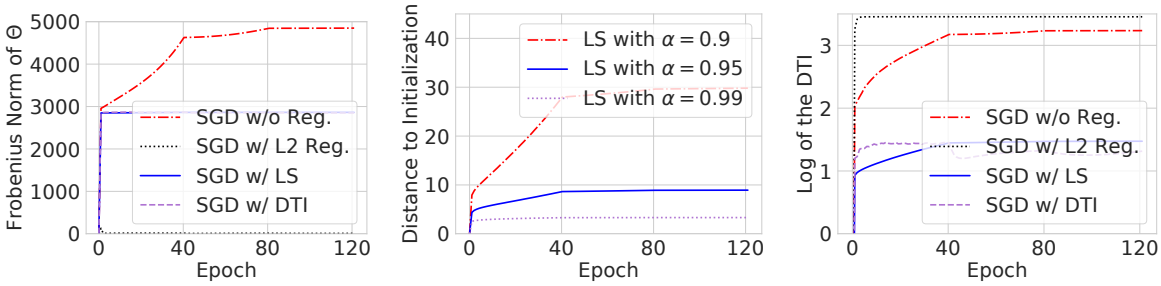
$$f(\mathbf{\Theta}, \boldsymbol{x}) \approx \phi(\boldsymbol{x})^\top\mathbf{\Theta}_\Delta. \quad (8)$$

At the end of training, each output of network leads to the kernel regression solution (Arora et al. 2019). Using the corresponding dimension in the targets $\hat{\boldsymbol{y}}^{[i]}$, the h-th output of the network at the end of training approximately computes the following function

$$f^{(h)}(\boldsymbol{x}) = k(\boldsymbol{x}, \mathbf{X})^\top(k(\mathbf{X}, \mathbf{X}))^{-1}\hat{\mathbf{Y}}^{(h)}, h \in [C], \quad (9)$$

[1] For the theoretical results we use ℓ_2 loss, while we present experimental results of both cross entropy and ℓ_2 loss in Figure 5.

[2] We can ensure small or even zero output at initialization by multiplying a small factor (Arora et al. 2019).



(a) $\|\boldsymbol{\Theta}\|_F$ (b) $\|\boldsymbol{\Theta}_\Delta\|_F$ (c) $\lg(\|\boldsymbol{\Theta}_\Delta\|_F)$

Figure 4: $\|\boldsymbol{\Theta}\|_F$, $\|\boldsymbol{\Theta}_\Delta\|_F$, and $\lg(\|\boldsymbol{\Theta}_\Delta\|_F)$ during training on CIFAR-10 with 40% symmetric noise. We set $\alpha = 0.9$ for LS and coefficient $\lambda = 0.02$ for both L2 regularization and DTI regularization. The backbone network is ResNet34.

where $\mathbf{X} = (\boldsymbol{x}^{[1]}, \ldots, \boldsymbol{x}^{[N]})$ denotes the inputs, $\hat{\mathbf{Y}} = (\hat{\boldsymbol{y}}^{[1]}, \ldots, \hat{\boldsymbol{y}}^{[N]})^\top \in \mathbb{R}^{C\times N}$ denotes the training targets, $k(\boldsymbol{x}, \mathbf{X}) = (k(\boldsymbol{x}, \boldsymbol{x}^{[1]}), k(\boldsymbol{x}, \boldsymbol{x}^{[2]}), \ldots, k(\boldsymbol{x}, \boldsymbol{x}^{[N]}))^\top \in \mathbb{R}^N$, and $k(\mathbf{X}, \mathbf{X}) \in \mathbb{R}^{N\times N}$ with (i, j)-th entry being $k(\boldsymbol{x}_i, \boldsymbol{x}_j)$. $\hat{\mathbf{Y}}^{(h)} \in \mathbb{R}^N$ is the h-th row of $\hat{\mathbf{Y}}$. Since the effect of ELS on network parameters is akin to the condition in NTK regime. Under the approximation in Eq. (8), it suffices to consider gradient descent on the objectives of ELS and DTI using the linearized model instead:

$$\tilde{\mathcal{L}}^{\text{ELS}}(\boldsymbol{\Theta}) = \frac{1}{2}\sum_{i=1}^{N}\left(\phi(\boldsymbol{x}^{[i]})^\top\boldsymbol{\Theta}_\Delta - (1-\alpha)\hat{\boldsymbol{y}}^{[i]} - \frac{\alpha}{C}\right)^2.$$

$$\tilde{\mathcal{L}}^{\text{DTI}}(\boldsymbol{\Theta}) = \frac{1}{2}\sum_{i=1}^{N}\left(\phi(\boldsymbol{x}^{[i]})^\top\boldsymbol{\Theta}_\Delta - \hat{\boldsymbol{y}}^{[i]}\right)^2 + \frac{\lambda^2}{2}\|\boldsymbol{\Theta}_\Delta\|^2.$$

We then drive the kernel approximations as follows:

Theorem 1 (Kernel Approximations) *Consider gradient descent on $\tilde{\mathcal{L}}(\boldsymbol{\Theta})$ with initialization $\boldsymbol{\Theta}_0$ and fixed learning rate $\gamma > 0$:*

$$\boldsymbol{\Theta}_{t+1} = \boldsymbol{\Theta}_t - \gamma\nabla_{\boldsymbol{\Theta}}\tilde{\mathcal{L}}(\boldsymbol{\Theta}_t), \quad t = 0, 1, 2, \ldots$$

For ELS, if the learning rate satisfies $\gamma \le \frac{1}{\|k(\mathbf{X},\mathbf{X})\|}$, the h-th output of the network learns the following kernel function:

$$f^{(h)}_{ELS}(\boldsymbol{x}) = k(\boldsymbol{x}, \mathbf{X})^\top(k(\mathbf{X}, \mathbf{X}))^{-1}\left[(1-\alpha)\hat{\mathbf{Y}}^{(h)} + \frac{\alpha}{C}\right],$$

where $h \in [C]$. For DTI, if the learning rate satisfies $\gamma \le \frac{1}{\|k(\mathbf{X},\mathbf{X})\|+\lambda^2}$, the h-th output of the network learns the kernel function:

$$f^{(h)}_{DTI}(\boldsymbol{x}) = k(\boldsymbol{x}, \mathbf{X})^\top\left(k(\mathbf{X}, \mathbf{X}) + \lambda^2\mathbf{I}\right)^{-1}\hat{\mathbf{Y}}^{(h)}.$$

Theorem 1 indicates that using gradient descent on ELS and DTI leads to the similar dynamics and converges to the kernel ridge regression solution using the NTK. Compared to regular training, the effect of ELS is to scale the function but preserves the maximal output in original entry. The effect of DTI is to add $\lambda^2\mathbf{I}$ to the kernel matrix.

Next, we show that gradient descent training on noisy data with ELS and DTI leads to a generalization guarantee on the clean data distribution.

Theorem 2 (Generalization Guarantee) *Consider the C-classification with noisy labels, let $\{(\boldsymbol{x}^{[i]}, y^{[i]}, \hat{y}^{[i]})\}_{i=1}^N$ be i.i.d. samples from the symmetric label noise, where the transition probabilities from a matrix $\mathbf{T}_{ij} = \Pr(\hat{y} = j \mid y = i)(\forall_{i,j} \in [C])$. Let $\mathbf{X} = (\boldsymbol{x}^{[1]}, \boldsymbol{x}^{[2]}, \ldots, \boldsymbol{x}^{[N]})$, and $\boldsymbol{y} = e^{(y)} \in \mathbb{R}^C$, $\hat{\boldsymbol{y}} = e^{(\hat{y})} \in \mathbb{R}^C$ be the one-hot label vectors. Denote $\mathbf{Y} = (\boldsymbol{y}^{[1]}, \boldsymbol{y}^{[2]}, \ldots, \boldsymbol{y}^{[N]}) \in \mathbb{R}^{C\times N}$, $\hat{\mathbf{Y}} = (\hat{\boldsymbol{y}}^{[1]}, \hat{\boldsymbol{y}}^{[2]}, \ldots, \hat{\boldsymbol{y}}^{[N]}) \in \mathbb{R}^{C\times N}$, and let $\hat{\mathbf{Y}}^{(h)} \in \mathbb{R}^N$ be the h-row of $\hat{\mathbf{Y}}$. Consider the kernel ridge regression solution in Theorem 1. Suppose the kernel matrix satisfies $tr[k(\mathbf{X}, \mathbf{X})] = O(N)$. Then with probability at least $1-\delta$, the classification error of $f^{(h)}_{ELS}(\boldsymbol{x})$ on the clean data distribution D is bounded as*

$$\Pr_{(\boldsymbol{x},y)\sim D}\left[y \notin \arg\max_{h\in[C]} f^{(h)}_{ELS}(\boldsymbol{x})\right] \le \frac{1}{1-\alpha}(O(\Upsilon) + \Omega),$$

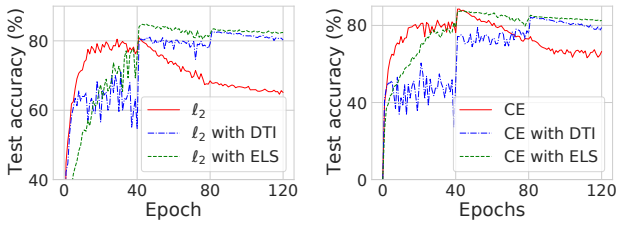
where $\Omega = C\cdot O\left(\sqrt{\frac{\log\frac{1}{\delta}}{N}} + \frac{1}{\sqrt{N}} + \sqrt{\frac{\log\frac{N}{\delta}}{N}}\right)$ and $\Upsilon = \sum_{h=1}^{C}\sqrt{\frac{(\mathbf{A}^{(h)})^\top(k(\mathbf{X},\mathbf{X}))^{-1}\mathbf{A}^{(h)}}{N}}$ and the classification error of $f^{(h)}_{DTI}(\boldsymbol{x})$ on D is bounded as

$$\Pr_{(\boldsymbol{x},y)\sim D}\left[y \notin \arg\max_{h\in[C]} f^{(h)}_{DTI}(\boldsymbol{x})\right] \le \frac{1}{\Psi}\left(\frac{\lambda + O(1)}{2}\Upsilon + \Omega'\right),$$

where $\Omega' = C\cdot O\left(\frac{1}{\lambda} + \sqrt{\frac{\log\frac{1}{\delta}}{N}} + \sqrt{\frac{\log\frac{N}{\delta\lambda}}{N}}\right)$, $\Psi = \min_{i,j\in[C],i\neq j}(\mathbf{T}_{i,i} - \mathbf{T}_{i,j})$, $\mathbf{A} = \mathbf{T}\cdot\mathbf{Y} \in \mathbb{R}^{C\times N}$ and $\mathbf{A}^{(h)}$ is the h-th row of $\mathbf{A}$.

Regarding the two generalization bounds in Theorem 2, when the number of samples $N \to \infty$ and hyperparameter λ in DTI grows with N [3], we have $\Omega \to 0$ and $\Omega' \to 0$. Therefore, the dominating terms in two bounds are $\frac{1}{1-\alpha}O(\Upsilon)$ and $\frac{\lambda}{2}O(\Upsilon)$. Notice that these two terms depend on the (unobserved) clean labels $\mathbf{Y}$, rather than the noisy labels $\hat{\mathbf{Y}}$. $O(\Upsilon)$ can be viewed as the complexity measure of data. One can easily derive that regular training on clean data leads to a population loss bound $O(\Upsilon)$. In comparison, the dominating term for ELS only has an extra factor $\frac{1}{1-\alpha}$, the dominating term for DTI has an extra factor λ. If

[3] This can be achieved by set $\lambda = N^q$ using a small constant q.



(a) ℓ_2 (b) CE

Figure 5: Test accuracy on CIFAR-10 with 40% symmetric noise. We evaluate the classification performance of CE, ℓ_2, and with ELS ($\alpha = 0.9$) and DTI ($\lambda = 0.02$) on them.

$(\mathbf{A}^{(h)})^\top (k(\mathbf{X}, \mathbf{X}))^{-1} \mathbf{A}^{(h)}$ grows much slower than N, by choosing an appropriate α and λ, these two generalization bounds on the noisy data distribution are comparable to the bound when trained with clean data, indicating that the underlying clean distribution is still learnable in the presence of label noise. Therefore, ELS improves the classification performance under symmetric label noise.

As can be observed in Figure 5(a), training with DTI and ELS achieve more robust test accuracy compared to using ℓ_2 loss alone, which verifies our theoretical results. Figure 5(b) shows the similar results on CE loss.

Noise Robustness by Model Confidence

We define the risk of a classifier f under clean distribution as $R_\ell(f) = \mathbb{E}_D[\ell(y, f(\boldsymbol{\Theta}, \boldsymbol{x}))]$, and under noisy distribution as $R_\ell^\eta(f) = \mathbb{E}_{\hat{D}}[\ell(\hat{y}, f(\boldsymbol{\Theta}, \boldsymbol{x}))]$. Let f^* and f_η^* be the global minimizers of $R_\ell(f)$ and $R_\ell^\eta(f)$ respectively. (Note that the achievable global minimization is a strong assumption, we use it because it is widely accepted in existing works (Ghosh, Kumar, and Sastry 2017; Wang et al. 2019; Ma et al. 2020)). We have demonstrated that the loss minimization of BL, BC, reversed LS and reversed ERL pushes the model to produce high-confidence predictions towards any vertex $\boldsymbol{v}$ of $(C-1)$-simplex, i.e., the resultant model f outputs the prediction $f(\boldsymbol{\Theta}, \boldsymbol{x}) \in V$, where V denotes the vertices set of $(C-1)$-simplex. However, it causes the optimization to fail as this discrete mapping produces many zero gradients when using gradient-based strategies. Therefore, we relax the restriction on confident predictions with an error ϵ in the following assumption.

Assumption 1 (Confident Prediction with error ϵ) *When optimizing loss functions in BL, BC, reversed LS and reversed ERL, the trained classifier f produces the confident prediction $f(\boldsymbol{\Theta}, \boldsymbol{x})$ such that $|f(\boldsymbol{\Theta}, \boldsymbol{x}) - \boldsymbol{v}| \leq \epsilon$, where $\boldsymbol{v} \in V$ and ϵ denotes the restriction error.*

According to the Assumption 1, we have the following bounded risk guarantee.

Theorem 3 (Bounded Risk Guarantee with error ϵ) *In the C-class classification problem, for any two predictions $f(\boldsymbol{\Theta}, \boldsymbol{x})_1$ and $f(\boldsymbol{\Theta}, \boldsymbol{x})_2$ given classifier f, suppose the loss ℓ satisfies $|\sum_{c=1}^{C}(\ell(f(\boldsymbol{\Theta}, \boldsymbol{x})_1, c) - \ell(f(\boldsymbol{\Theta}, \boldsymbol{x})_2, c))| \leq \delta$ when $|f(\boldsymbol{\Theta}, \boldsymbol{x})_1 - f(\boldsymbol{\Theta}, \boldsymbol{x})_2| \leq \epsilon$, and $\delta \to 0$ when $\epsilon \to 0$. Then for symmetric label noise satisfying $\eta < \frac{C-1}{C}$, the two risks can be expressed as*

$$R_\ell(f_\eta^*) - R_\ell(f^*) \leq \frac{2\eta\delta}{C(1-\eta) - 1} \tag{10}$$

where f_η^* and f^* denote the minimizer of $R_\ell^\eta(f)$ and $R_\ell(f)$, respectively.

Theorem 3 indicates that when the classifier produces high-confidence predictions following Assumption 1, the difference of the risks caused by the derived f_η^* and f^* under noisy labels and clean labels are always bounded. The bound is related to the error ϵ and the noise rate η. When $\epsilon \to 0$ or $\eta \to 0$, the bound tends to 0. Therefore, BL, BC, reversed LS and reversed ERL achieve robustness under label noise.

Related Work

Learning with hard (one-hot) labels is prone to over-fitting, thus using refurbished (soft) labels naturally attracts more attention. In addition to the four label refurbishment methods discussed in this paper, label distribution learning (Geng 2016) provides instances with description degrees of all the possible labels. Empirical studies have demonstrated that LS boosts performance on different tasks (Vaswani et al. 2017; Chorowski and Jaitly 2017) and also improves model calibration (Müller, Kornblith, and Hinton 2019). Later, more advanced forms of LS were proposed, such as structural LS (Li, Dasarathy, and Berisha 2020) and non-uniform LS (Chen et al. 2020). In the presence of label noise, Lukasik et al. (2020) empirically demonstrate that LS improves the model performance. Liu (2021) provides theoretical analysis for the memorization behavior of LS. Wei et al. (2022) show the effectiveness of negative label smoothing. Many state-of-the-art noise-robust methods incorporate LS, BL or ERL in their frameworks (Ma et al. 2018; Arazo et al. 2019; Li, Socher, and Hoi 2020; Zhou, Wang, and Bilmes 2020).

Conclusion and Future Work

In this paper, we generalize four label refurbishment methods and investigate their effects on loss function and model confidence. We conduct extensive experiments to show that label refurbishment methods can effectively improve classification performance in the presence of label noise. In theory, we explain two important phenomena in classification with noisy labels: (1) The regularization effect caused by extreme label smoothing ensures that the model has generalization guarantees on clean data. (2) The models with high-confidence predictions are robust to label noise. Overall, our findings shed light on the potential benefits of label refurbishment methods, and provide formal exploration of their denoising effects.

Given most real-world datasets contain noisy labels, we believe our findings can trigger interest in designing new forms of techniques that improves model robustness to label noise in practical applications. More broadly, this work represents a step towards studying the effects of common tricks in training deep neural networks. Explaining more tricks theoretically is left for future work.

References

Arazo, E.; Ortego, D.; Albert, P.; O'Connor, N.; and McGuinness, K. 2019. Unsupervised label noise modeling and loss correction. In *International conference on machine learning*, 312–321. PMLR.

Arora, S.; Du, S. S.; Hu, W.; Li, Z.; Salakhutdinov, R. R.; and Wang, R. 2019. On exact computation with an infinitely wide neural net. *Advances in Neural Information Processing Systems*, 32.

Chen, B.; Ziyin, L.; Wang, Z.; and Liang, P. P. 2020. An investigation of how label smoothing affects generalization. *arXiv preprint arXiv:2010.12648*.

Chorowski, J.; and Jaitly, N. 2017. Towards Better Decoding and Language Model Integration in Sequence to Sequence Models. *Proc. Interspeech 2017*, 523–527.

Geng, X. 2016. Label distribution learning. *IEEE Transactions on Knowledge and Data Engineering*, 28(7): 1734–1748.

Ghosh, A.; Kumar, H.; and Sastry, P. 2017. Robust loss functions under label noise for deep neural networks. In *Thirty-First AAAI Conference on Artificial Intelligence*.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Hu, W.; Li, Z.; and Yu, D. 2019. Simple and Effective Regularization Methods for Training on Noisily Labeled Data with Generalization Guarantee. In *International Conference on Learning Representations*.

Jacot, A.; Gabriel, F.; and Hongler, C. 2018. Neural tangent kernel: convergence and generalization in neural networks. In *Advances in neural information processing systems*, 8580–8589.

Krizhevsky, A.; et al. 2009. Learning multiple layers of features from tiny images. *Technical report, CIFAR*.

LeCun, Y.; Bottou, L.; Bengio, Y.; and Haffner, P. 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11): 2278–2324.

Li, J.; Socher, R.; and Hoi, S. C. 2020. DivideMix: Learning with Noisy Labels as Semi-supervised Learning. In *International Conference on Learning Representations*.

Li, W.; Dasarathy, G.; and Berisha, V. 2020. Regularization via structural label smoothing. In *International Conference on Artificial Intelligence and Statistics*, 1453–1463. PMLR.

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

Lu, Y.; and He, W. 2022. SELC: Self-Ensemble Label Correction Improves Learning with Noisy Labels. *International Joint Conference on Artificial Intelligence*.

Lukasik, M.; Bhojanapalli, S.; Menon, A.; and Kumar, S. 2020. Does label smoothing mitigate label noise? In *International Conference on Machine Learning*, 6448–6458. PMLR.

Ma, X.; Huang, H.; Wang, Y.; Romano, S.; Erfani, S.; and Bailey, J. 2020. Normalized loss functions for deep learning with noisy labels. In *International Conference on Machine Learning*, 6543–6553. PMLR.

Ma, X.; Wang, Y.; Houle, M. E.; Zhou, S.; Erfani, S.; Xia, S.; Wijewickrema, S.; and Bailey, J. 2018. Dimensionality-driven learning with noisy labels. In *International Conference on Machine Learning*, 3355–3364. PMLR.

Müller, R.; Kornblith, S.; and Hinton, G. E. 2019. When does label smoothing help? *Advances in Neural Information Processing Systems*, 32: 4694–4703.

Natarajan, N.; Dhillon, I. S.; Ravikumar, P. K.; and Tewari, A. 2013. Learning with noisy labels. *Advances in neural information processing systems*, 26: 1196–1204.

Patrini, G.; Rozza, A.; Krishna Menon, A.; Nock, R.; and Qu, L. 2017. Making deep neural networks robust to label noise: A loss correction approach. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1944–1952.

Pereyra, G.; Tucker, G.; Chorowski, J.; Kaiser, Ł.; and Hinton, G. 2017. Regularizing neural networks by penalizing confident output distributions. *International Conference on Learning Representations (Workshop)*.

Reed, S. E.; Lee, H.; Anguelov, D.; Szegedy, C.; Erhan, D.; and Rabinovich, A. 2015. Training Deep Neural Networks on Noisy Labels with Bootstrapping. In *ICLR (Workshop)*.

Shen, Z.; Liu, Z.; Xu, D.; Chen, Z.; Cheng, K.-T.; and Savvides, M. 2020. Is Label Smoothing Truly Incompatible with Knowledge Distillation: An Empirical Study. In *International Conference on Learning Representations*.

Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; and Wojna, Z. 2016. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2818–2826.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Wang, Y.; Ma, X.; Chen, Z.; Luo, Y.; Yi, J.; and Bailey, J. 2019. Symmetric Cross Entropy for Robust Learning with Noisy Labels. *Proceedings of the IEEE International Conference on Computer Vision*.

Wei, J.; Liu, H.; Liu, T.; Niu, G.; and Liu, Y. 2022. Understanding Generalized Label Smoothing when Learning with Noisy Labels. *International Conference on Machine Learning*.

Yuan, L.; Tay, F. E.; Li, G.; Wang, T.; and Feng, J. 2020. Revisiting knowledge distillation via label smoothing regularization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3903–3911.

Zhang, C.; Bengio, S.; Hardt, M.; Recht, B.; and Vinyals, O. 2018. Understanding deep learning requires rethinking generalization. In *International Conference on Learning Representations*.

Zhang, C.-B.; Jiang, P.-T.; Hou, Q.; Wei, Y.; Han, Q.; Li, Z.; and Cheng, M.-M. 2021. Delving deep into label smoothing. *IEEE Transactions on Image Processing*, 30: 5984–5996.

Zhou, T.; Wang, S.; and Bilmes, J. 2020. Robust curriculum learning: from clean label detection to noisy label self-correction. In *International Conference on Learning Representations*.