A CURRICULUM VIEW OF ROBUST LOSS FUNCTIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Robust loss functions are designed to mitigate the adverse impacts of label noise, which enjoy theoretical guarantees that is agnostic to the training dynamics. However, these guarantees fail to characterize some empirical phenomenons. We unify a broad array of loss functions into a novel standard form with a shared implicit loss function and different implicit design choices along the sample-weighting function and the output regularizer. The resulting curriculum view connects the designs of loss functions and learning curricula, leading to a straightforward analysis of the training dynamics that helps demystify existing empirical observations. In particular, we show that robust sample-weighting function sift and neglect corrupted samples. We further analyze how regularizers affect the training dynamics with different loss functions. Finally, we dissect the cause of the notorious underfitting issue and support our explanation with our effective fixes.

1 INTRODUCTION

Label noise is non-negligible in automatic labeling (Liu et al., 2021), crowd-sourcing (Russakovsky et al., 2015) and expert annotation (Kato & Matsubara, 2010). Their adverse impacts can be mitigated with loss functions that are theoretically robust against label noise (Song et al., 2020), which enjoys a bounded discrepancy between the optima obtained with clean or noisy labels (Ghosh et al., 2017; Zhou et al., 2021). However, existing theoretical bounds do not account for the training dynamics towards these optima. This oversight has led to confounding empirical observations, e.g., robust loss functions can underfit, or perform poorly, on difficult tasks (Zhang & Sabuncu, 2018; Wang et al., 2019c). In addition, it is unclear why seemingly contradictory regularizers (Lukasik et al., 2020; Wei et al., 2021) can improve noise-robustness at the same time.

We unify loss functions with *distinct theoretical motivations* into a standard form, unveiling their *implicit* design choices along the sample-weighting functions and the output regularizers (§3). The *shared* implicit loss function in the standard form provides a common metric to track the learning progress of each sample. We can demystify existing empirical observations by analyzing how different design choices affect the evolution of implicit loss distributions. Notably, our derivation shows that each loss function implicitly defines a learning curriculum, which specifies a sequence of training criteria at different training steps (Wang et al., 2020). This curriculum view connects the designs of loss functions and curricula that are commonly viewed as distinct approaches (Song et al., 2020).

We provide a comprehensive analysis on how different design choices affect the training dynamics, leading to plenty empirically supported *understandings* for these loss functions. We first show that sample-weighting functions of robust loss functions act as sample sifts that neglect samples with erroneous labels (§4.1), which utilizes the memorization effect (Arpit et al., 2017) similar to curriculum based approaches (Li et al., 2020). We then analyze how regularizers affect the training dynamics with different loss functions (§4.2). In particular, we are the first to show the vital role of confidence reducing regularizers, e.g., weight decay, in successful training with robust loss function. Finally, we attribute the underfitting issue of robust loss functions to minimal sample weights at initialization and support it with our effective fixes (§4.3).

2 RELATED WORK

Most existing studies on robust loss functions (Ghosh et al., 2017; Zhang & Sabuncu, 2018; Wang et al., 2019c; Feng et al., 2020; Liu & Guo, 2020; Cheng et al., 2021; Zhou et al., 2021) focus on deriving bounds of the difference between optima obtained with noisy and clean labels, which are agnostic to the training dynamics. We unify them into a standard form and thoroughly analyze how implicit choices along sample-weighting functions and output regularizers affect the training dynamics. Although the underfitting issue has been heuristically mitigated (Zhang & Sabuncu, 2018; Wang et al., 2019a;c; Ma et al., 2020), we explicitly identify the cause and support it with our fixes. Both confidence reducing (Lukasik et al., 2020) and promoting (Wei et al., 2021; Cheng et al., 2021) regularizers are shown to improve the noise robustness of cross entropy. We analyze how different regularizers affect the training dynamics with different loss functions, especially with robust loss functions. We only consider loss functions with closed-form expressions and leave others (Amid et al., 2019; Wei & Liu, 2021) to future work.

Our curriculum view connects existing robust loss functions to the seemingly distinct (Song et al., 2020) curriculum learning. To achieve robustness against noisy labels, curriculum-based approaches use either sample selection (Chen et al., 2019; Huang et al., 2019) or sample weighting (Chang et al., 2017; Ren et al., 2018), typically based on the small loss trick (Song et al., 2020) utilizing the memorization effect (Arpit et al., 2017). Our work differs in three important perspectives. First, the sample weights analyzed in our work are *implicitly* defined by robust loss functions rather than *explicitly* designed (Chang et al., 2017; Wang et al., 2019a;b) or predicted by a model (Jiang et al., 2018; Ren et al., 2018). Second, we use the derived implicit loss function to track the learning of samples unlike common metrics based on loss values (Kumar et al., 2010; Loshchilov & Hutter, 2015) or gradient magnitudes (Gopal, 2016). Finally, instead of designing better sample-weighting curricula (Chang et al., 2017; Wang et al., 2019a;b), we aim to thoroughly understand how the identified design choices of *existing* loss functions affects the training dynamics. Our work is also related to the ongoing debate (Hacohen & Weinshall, 2019; Wang et al., 2020) on the strategies to select or weight samples in learning curricula: either easier first (Bengio et al., 2009; Kumar et al., 2010) or harder first (Loshchilov & Hutter, 2015; Zhang et al., 2018). The sample-weighting functions we identified in robust loss functions can be viewed as a combination of both strategies, which emphasizes samples with moderate difficulty, providing a novel weighting paradigm for curriculum learning.

Most related to our work, Wang et al. (2019a) view the L1 norm of logit gradients as sample weights and focus on an *intuitive* fix for the underfitting of MAE. We extend the analysis of logit gradients to *explicitly* identify the implicit loss function and the aforementioned design choices for *a wide range* of loss functions, leading to a unified standard form. In addition, we thoroughly analyze the training dynamics with different design choices, resulting in deeper understanding of noise robustness and vulnerability to underfitting that can facilitate better designs of loss functions and learning curricula.

3 Deriving the Standard Form

We formulate classification with label noise and noise robustness before presenting our results. The k-ary classification task with input $\boldsymbol{x} \in \mathbb{R}^d$ can be solved by classifier $\arg\max_i s_i$, where s_i is the score (logit) for the i-th class in function $\boldsymbol{s} : \mathbb{R}^d \to \mathbb{R}^k$ parameterized by $\boldsymbol{\theta}$. With a slight abuse of notation, we use $\boldsymbol{s}$ for both the scoring function $\boldsymbol{s}(\boldsymbol{x};\boldsymbol{\theta})$ and its output. Given label $y^* \in \{1,\dots,k\}$ for $\boldsymbol{x}$ in a clean data distribution $\mathcal{D}^*$ and a loss function $L(\boldsymbol{s}, y^*)$, we can estimate $\boldsymbol{\theta}$ with risk minimization $\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \mathbb{E}_{(\boldsymbol{x},y^*)\sim\mathcal{D}^*}[L(\boldsymbol{s}(\boldsymbol{x};\boldsymbol{\theta}), y^*)]$.

Labeling errors turn clean labels y^* into noisy ones, $y \sim P(y|\boldsymbol{x}, y^*)$, leading to a noisy data distribution $\mathcal{D}$ with noise rate $\eta = P(y \neq y^*)$. A sample is *corrupted* when $y \neq y^*$. Following Ghosh et al. (2017), label noise is *symmetric* (uniform) if $P(y|\boldsymbol{x}, y^*) = \eta/(k-1), \forall y \neq y^*$ and *asymmetric* (class-conditional) when $P(y|\boldsymbol{x}, y^*) = P(y|y^*)$. Loss function L is robust against label noise if parameters estimated with $\mathcal{D}$, $\boldsymbol{\theta'} = \arg\min_{\boldsymbol{\theta}} \mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}}[L(\boldsymbol{s}(\boldsymbol{x};\boldsymbol{\theta}), y)]$, leads to bounded extra risk

$$\mathbb{E}_{(\boldsymbol{x},y^*)\sim\mathcal{D}^*}[L(\boldsymbol{s}(\boldsymbol{x};\boldsymbol{\theta'}), y^*)] - \mathbb{E}_{(\boldsymbol{x},y^*)\sim\mathcal{D}^*}[L(\boldsymbol{s}(\boldsymbol{x};\boldsymbol{\theta}^*), y^*)] \leq \epsilon, \tag{1}$$

under clean data distribution $(\boldsymbol{x}, y^*) \sim \mathcal{D}^*$, where ϵ is a constant.

We examine a broad array of loss functions with closed-form expressions in this work, among which Cross Entropy (CE), Focal Loss (FL; Lin et al., 2017) and Symmetric Cross Entropy (SCE; Wang et al., 2019c) are not robust as they have unbounded extra risk. In contrast, Mean Absolute Error (MAE; Ghosh et al., 2017), Taylor Cross Entropy (TCE; Feng et al., 2020), Generalized Cross Entropy (GCE; Zhang & Sabuncu, 2018), Mean Square Error (MSE; Ghosh et al., 2017), Peer Loss (PL; Liu & Guo, 2020) and asymmetric losses (Zhou et al., 2021) including Asymmetric Generalized Cross Entropy (AGCE), Asymmetric Unhinged Loss (AUL) and Asymmetric Exponential Loss (AEL) are all robust. See Appendix A.2 for a more thorough review and the derivation of equivalence between the Reverse Cross Entropy (RCE; Wang et al., 2019c) and MAE.

Name	Formula with $\tilde{L}(p_y)$	$w(\boldsymbol{s}, y)$	Constraints
CE	$-\log p_y$	$1 - p_y$	
FL	$-(1-p_y)^q \log p_y$	$(1-p_y)^q(1-p_y-qp_y \log p_y)$	$q > 0$
SCE	$-(1-q)\log p_y + q(1-p_y)$	$(1-q+q\cdot p_y)(1-p_y)$	$0 < q < 1$
TCE	$\sum_{i=1}^q (1-p_y)^i/i$	$p_y \sum_{i=1}^q (1-p_y)^i$	$q \in \mathbb{N}^+$
GCE	$(1-p_y^q)/q$	$p_y^q(1-p_y)$	$0 < q \leq 1$
RCE / MAE	$1-p_y$	$p_y(1-p_y)$	
AEL	$e^{-p_y/q}$	$p_y(1-p_y)e^{-p_y/q}/q$	$q > 0$
AUL	$[(a-p_y)^q-(a-1)^q]/q$	$p_y(1-p_y)(a-p_y)^{q-1}$	$a > 1, q > 0$
AGCE	$[(a+1)^q-(a+p_y)^q]/q$	$p_y(1-p_y)(a+p_y)^{q-1}$	$a > 0, q > 0$

Table 1: Formulae, hyperparameter constraints and sample-weighting functions $w(\boldsymbol{s}, y)$ for loss functions conforming to Eq. (3). See Appendix A.2 for plots of $w(\boldsymbol{s}, y)$. Notably, $w(\boldsymbol{s}, y)$ of all robust loss functions emphasize samples with moderate $\Delta(\boldsymbol{s}, y)$.

We now derive the standard form of the aforementioned loss functions. As shown in Table 1, most loss functions are functions of the softmax probability, $L(\boldsymbol{s}, y) = \tilde{L}(p_y)$, where

$$p_y = \frac{e^{s_y}}{\sum_i e^{s_i}} = \frac{1}{e^{-(s_y - \log \sum_{i \neq y} e^{s_i})} + 1} = \frac{1}{e^{-\Delta(\boldsymbol{s},y)} + 1},$$

with

$$\Delta(\boldsymbol{s}, y) = s_y - \log \sum_{i \neq y} e^{s_i} \leq s_y - \max_{i \neq y} s_i \tag{2}$$

measuring the score margin between the labeled class y and any other classes. Since $\nabla_{\boldsymbol{s}} \Delta(\boldsymbol{s}, y)$ has a constant norm, $\|\nabla_{\boldsymbol{s}} \Delta(\boldsymbol{s}, y)\|_1 = 2$, extracting it from the sample gradient $\nabla_{\boldsymbol{s}} L(\boldsymbol{s}, y)$

$$\nabla_{\boldsymbol{s}} L(\boldsymbol{s}, y) = \frac{\mathrm{d}\tilde{L}(p_y)}{\mathrm{d}\Delta(\boldsymbol{s}, y)} \cdot \nabla_{\boldsymbol{s}} \Delta(\boldsymbol{s}, y) = -w(\boldsymbol{s}, y) \cdot \nabla_{\boldsymbol{s}} \Delta(\boldsymbol{s}, y)$$

factorizes the scale $w(\boldsymbol{s}, y)$ and direction $\nabla_{\boldsymbol{s}} \Delta(\boldsymbol{s}, y)$. After detaching $w(\boldsymbol{s}, y)$ from derivative and integral and integrating over $\boldsymbol{s}$, *equivalent under first-order optimizers*, $L(\boldsymbol{s}, y)$ can be rewritten into

$$L'(\boldsymbol{s}, y) = -w(\boldsymbol{s}, y) \cdot \Delta(\boldsymbol{s}, y) \tag{3}$$

where $w(\boldsymbol{s}, y) = -\kappa[\mathrm{d}\tilde{L}(p_y)/\mathrm{d}\Delta(\boldsymbol{s}, y)] > 0$ is a **sample-weighting function** with $\kappa(\cdot)$ the detach operator and $\Delta(\boldsymbol{s}, y)$ serves as a shared **implicit loss function**. We show $w(\boldsymbol{s}, y)$ of different loss functions in Table 1 and their plots in Fig. 5 of Appendix A. Intuitively, $\Delta(\boldsymbol{s}, y)$ determines the gradient directions to learn the labeled samples, while $w(\boldsymbol{s}, y)$ affects the priority of sample learning by emphasizing different samples during training. Loss functions conforming to Eq. (3) thus implicitly define different sample-weighting curricula. Unlike loss values (Loshchilov & Hutter, 2015) and gradient magnitudes (Gopal, 2016), $\Delta(\boldsymbol{s}, y)$ is *shared* by loss functions considered in this work and isolates the influence of $w(\boldsymbol{s}, y)$ and arbitrary loss scales. It further avoids the sigmoid transform in p_y rendering the change of large $|\Delta(\boldsymbol{s}, y)|$ less perceptible. $\Delta(\boldsymbol{s}, y)$ is thus an effective common metric for tracking the learning progress of samples with different loss functions.

Loss functions additionally depending on $\{p_i\}_{i \neq y}$ may fail to conform to Eq. (3), including Normalized Cross Entropy (NCE; Ma et al., 2020), Mean Square Error (MSE; Ghosh et al., 2017), Peer Loss (PL; Liu & Guo, 2020), Jensen-Shannon Divergence (JS; Englesson & Azizpour, 2021), and CE with Confidence Regularization (CR; Cheng et al., 2021), Negative Label Smoothing (NLS; Wei et al., 2021) or Label Smoothing (LS; Lukasik et al., 2020). However, we can extract an output regularizer $R(\boldsymbol{s})$ constraining the distribution of $\boldsymbol{s}$, leaving a **primary loss function** $L'(\boldsymbol{s}, y)$ conforming to Eq. (3). We show the derivation of NCE and leave other loss functions to Appendix A.5. Given

$$\nabla_{\boldsymbol{s}} L_{\mathrm{NCE}}(\boldsymbol{s}, y) = \nabla_{\boldsymbol{s}} \frac{L_{\mathrm{CE}}(\boldsymbol{s}, y)}{\sum_i L_{\mathrm{CE}}(\boldsymbol{s}, i)} = \frac{-1}{\sum_{i=1}^k \log p_i} \nabla_{\boldsymbol{s}} L_{\mathrm{CE}}(\boldsymbol{s}, y) + \frac{-k \log p_y}{(\sum_{i=1}^k \log p_i)^2} \nabla_{\boldsymbol{s}} \sum_{i=1}^k \frac{1}{k} \log p_i$$

Name	Formula with p_i	Primary	$R(\boldsymbol{s})$	$\Delta(\boldsymbol{s}^*, y)$
NCE	$\log p_y / \sum_{i=1}^{k} \log p_i$	$-\gamma \cdot \log p_i$	$\sum_{i=1}^{k} \frac{1}{k} \log p_i$	$\pm\infty$
MSE	$\sum_{i=1}^{k} [\mathbb{I}(i = y) - p_i]^2$	$1 - p_y$	$\sum_{i=1}^{k} p_i^2$	$-\log k$
PL (CE + CR)	$-\log p_y + \mathbb{E}_{\mathrm{y} \sim \mathcal{D}, \mathbf{x} \sim \mathcal{D}}[\log p_{\mathrm{y} \mid \mathbf{x}}]$	$-\log p_y$	$\sum_{i=1}^{k} P(y = i) \log p_i$	$\pm\infty$
JS	$a \sum_i p_i \log \frac{p_i}{m_i} + (1 - a) \sum_i e_i \log \frac{e_i}{m_i}$	$\tilde{L}'_{\mathrm{JS}}(p_y)$	$\sum_{i=1}^{k} p_i$	$-\log k$
CE + LS	$-\sum_{i=1}^{k} [\mathbb{I}(i = y)(1 - q) + \frac{q}{k}] \log p_i$	$-\log p_y$	$-\sum_{i=1}^{k} \frac{1}{k} \log p_i$	$-\log k$
CE + NLS	$-\sum_{i=1}^{k} [\mathbb{I}(i = y)(1 + r) - \frac{r}{k}] \log p_i$	$-\log p_y$	$\sum_{i=1}^{k} \frac{1}{k} \log p_i$	$\pm\infty$

Table 2: Formulae, primary loss functions, output regularizers and $\Delta(s^*, y)$ at the minimum s^* of the regularizers for loss functions conforming to Eq. (4), where $0 < a < 1$, $0 \le q < 1$ and $r > 0$ are all hyperparameters. We view PL in its expectation to derive its regularizer CR (Cheng et al., 2021). $p_{\mathrm{y}|\mathbf{x}}$ is the softmax probability of a random label y with a random input **x** sampled from the noisy data distribution. For JS, $e_i = \mathbb{I}(i = y)$ where $\mathbb{I}(\cdot)$ is the indicator function and $m_i = ap_i + (1-a)e_i$. Finally, $\tilde{L}'_{\mathrm{JS}}(p_y) = ap_y \log ap_y - (ap_y + 1 - a) \log(ap_y + 1 - a)$.

by detaching weighting functions from derivative and integral and integrating over $\boldsymbol{s}$, NCE becomes

$$L''_{\mathrm{NCE}}(\boldsymbol{s}, y) = \gamma(\boldsymbol{s}, y) \cdot w_{\mathrm{CE}}(\boldsymbol{s}, y) \cdot \Delta(\boldsymbol{s}, y) + \lambda_{\mathrm{NCE}}(\boldsymbol{s}, y) \cdot R_{\mathrm{NCE}}(\boldsymbol{s})$$

where $\gamma(\boldsymbol{s}, y) = \kappa(-1/\sum_{i=1}^{k} \log p_i)$ and $\lambda_{\mathrm{NCE}}(\boldsymbol{s}, y) = \kappa[-k \log p_y / (\sum_{i=1}^{k} \log p_i)^2]$ are weighting functions wrapped with the detach operator $\kappa(\cdot)$, and $R_{\mathrm{NCE}}(\boldsymbol{s}) = \sum_{i=1}^{k} \frac{1}{k} \log p_i$ is the output regularizer of NCE. Similar derivations thus lead to a **standard form** of loss functions,

$$L''(\boldsymbol{s}, y) = L'(\boldsymbol{s}, y) + \lambda(\boldsymbol{s}, y) \cdot R(\boldsymbol{s}) = -w(\boldsymbol{s}, y) \cdot \Delta(\boldsymbol{s}, y) + \lambda(\boldsymbol{s}, y) \cdot R(\boldsymbol{s}), \tag{4}$$

where $\lambda(\boldsymbol{s}, y)$ is a weighting function adjusting the per-sample strength of regularization, which is typically a constant when explicit regularization are applied, e.g., label smoothing. Intuitively, the output regularizer $R(\boldsymbol{s})$ constrains the distributions of $\Delta(\boldsymbol{s}, y)$ towards a predefined optimum $\Delta(\boldsymbol{s}^*, y)$, where $\boldsymbol{s}^* = \arg\min_{\boldsymbol{s}} R(\boldsymbol{s})$. We show loss functions conforming to Eq. (4), their primary loss functions and regularizers, and $\Delta(\boldsymbol{s}^*, y)$ of regularizers in Table 2.

Our derivation based on $\nabla_{\boldsymbol{s}} L(\boldsymbol{s}, y)$ is inspired by Wang et al. (2019a), who view $\|\nabla_{\boldsymbol{s}} L(\boldsymbol{s}, y)\|_1$ as sample weights for MAE and CE. We further use our derivative-detach-integral trick to *explicitly* extract the orthogonal components $w(\boldsymbol{s}, y)$, $\Delta(\boldsymbol{s}, y)$ and $R(\boldsymbol{s})$ from a wide range of loss functions, which unifies them into a standard form and identifies their implicit differences in $w(\boldsymbol{s}, y)$ and $R(\boldsymbol{s})$.

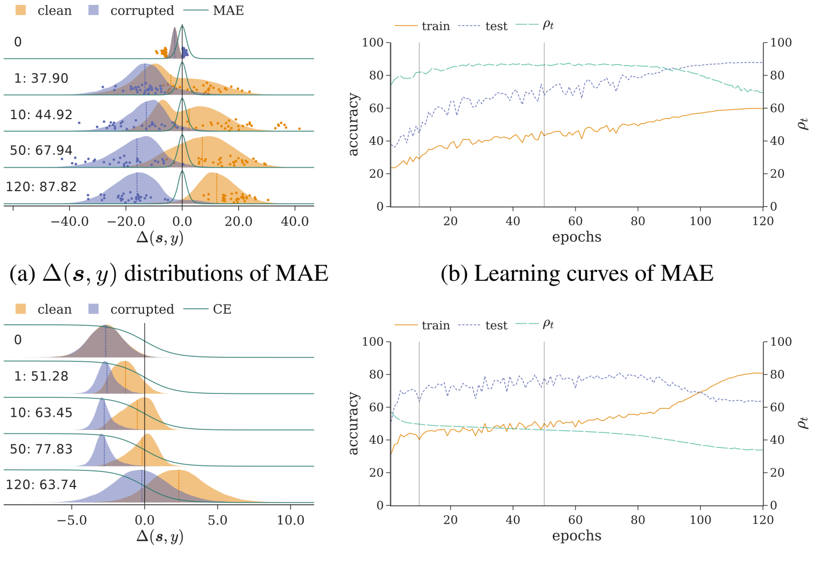
4 Training Dynamics of Robust Loss Functions

Based on the curriculum view with the standard form Eq. (4), we analyze the training dynamics of loss functions from random initializations towards the optima. Ideal loss functions should facilitate rapid learning while suppress the learning of corrupted samples. We analyze how implicit design choices along the sample-weighting function $w(\boldsymbol{s}, y)$ and the output regularizers $R(\boldsymbol{s})$ affect the evolution of $\Delta(\boldsymbol{s}, y)$ distributions, which provide a fine-grained view of the learning progress. We mainly use MAE and CE for illustration as they exhibit typical empirical observations. Analysis of other loss functions with similar results are left to Appendix B.

We conduct experiments on CIFAR10/100 (Krizhevsky, 2009) with synthetic symmetric and asymmetric label noise following Ma et al. (2020); Zhou et al. (2021), as well as human label noise (Wei et al., 2022). For symmetric label noise, the labels are randomly flipped to a different class. For asymmetric label noise on CIFAR10, we randomly flip TRUCK → AUTOMOBILE, BIRD → AIRPLANE, DEER → HORSE, CAR ↔ DOG. For CIFAR100, the 100 classes are grouped into 20 super-classes and labels are flipped within the same super-class into the next in a circular fashion. Human label noise is adopted from (Wei et al., 2022). To reflect more difficult settings than those in existing research, we include results on the large scale noisy dataset WebVision (Li et al., 2017) with *larger* subsets of classes. We use momentum 0.9, learning rate 0.1, batch size 128 and weight decay 5e-4 across all settings. We adopt an 8-layer CNN for CIFAR10, a ResNet-34 for CIFAR100 and a ResNet50 for WebVision, respectively, all with batch normalization. We train for 120, 200

	Clean	Asymmetric		Symmetric						Human	
		$\eta = 0.2$		$\eta = 0.2$		$\eta = 0.4$		$\eta = 0.8$		$\eta = 0.4$	
Loss	Acc	Δ_{acc}	ρ	Δ_{acc}	ρ	Δ_{acc}	ρ	Δ_{acc}	ρ	Δ_{acc}	ρ
CE	93.28	-6.79	74.24	-14.67	61.30	-29.54	47.38	-65.21	19.75	-27.73	44.70
FL	92.88	-3.31	77.33	-12.55	57.29	-27.20	43.42	-65.50	19.52	-25.64	41.08
SCE	93.09	-5.84	79.62	-6.72	74.02	-16.03	60.69	-55.27	23.39	-21.26	53.58
AEL	92.59	-2.03	90.00	-2.00	93.89	-4.69	84.88	-39.04	34.72	-11.15	66.41
AUL	92.52	-2.41	89.97	-1.83	93.66	-4.59	84.41	-35.25	36.30	-12.49	65.95
GCE	92.51	-1.99	90.92	-2.12	93.91	-4.98	84.92	-31.80	33.45	-11.21	67.99
TCE	92.67	-2.51	89.84	-2.12	93.65	-4.62	84.51	-35.08	36.79	-12.13	66.32
MAE	92.25	-10.60	87.64	-1.70	93.95	-4.43	84.41	-82.25	21.60	-22.18	65.89
AGCE	92.23	-20.27	85.95	-2.36	93.35	-3.99	83.20	-82.23	20.66	-29.75	64.37

Table 3: Robust loss functions assign larger weights to clean samples. We report drop in accuracy Δ_{acc} and proportion of cumulative weights for clean samples ρ on CIFAR10 averaged with 3 different runs. Hyperparameters tuned with symmetric noise $\eta = 0.4$ are listed in Table 7 of Appendix B.



(a) $\Delta(\boldsymbol{s}, y)$ distributions of MAE

(b) Learning curves of MAE

(c) $\Delta(\boldsymbol{s}, y)$ distributions of CE

(d) Learning curves of CE

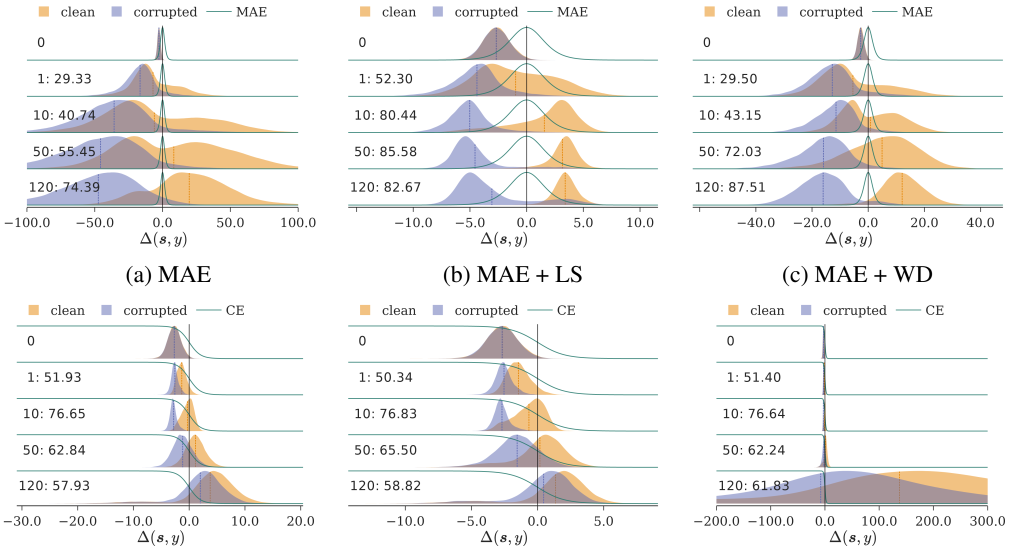
Figure 1: Training dynamics of MAE and CE on CIFAR10 with symmetric noise $\eta = 0.4$. Points of the top 1% and bottom 1% $\Delta(\boldsymbol{s}, y)$ of corrupted and clean samples, respectively, are included to demonstrate the effect of generalization in (a). We mark the median line of each distribution and include test accuracies in (a, c). Epochs sampled in (a, c) are marked in (b, d).

and 250 epochs on CIFAR10, CIFAR100 and WebVision, respectively. Unlike standard settings, we scale all $w(\boldsymbol{s}, y)$ to *unit maximum* to avoid complications, since hyperparameters of loss functions can change the scale of $w(\boldsymbol{s}, y)$, essentially adjusting the learning rate of SGD.

4.1 Effects of Sample-Weighting Functions

We show that $w(\boldsymbol{s}, y)$ of robust loss functions implicitly implement a sample sift that neglects corrupted samples by utilizing the memorization effect (Arpit et al., 2017), i.e., models tend to learn simple and general patterns before overfitting to noisy patterns. In addition, we show that the shared monotonically decreasing right tail of $w(\boldsymbol{s}, y)$ for bounded parameter update is responsible to the eventual shift into a noise overfitting phase observed in the memorization effect.

We first show that robust loss functions assign larger weights to clean samples, which leads to better noise robustness. The emphasis on clean samples can be estimated with the ratio of their cumulative weights, $\rho_t = \sum_i \mathbb{I}(y_i = y_i^*) w_{i,t} / \sum_i w_{i,t}$, where $w_{i,t}$ is the weight of the i-th sample at epoch t and $\mathbb{I}(\cdot)$ is the indicator function. The weighted-averaged ρ_t with learning rate α_t, $\rho = \sum_t \alpha_t \rho_t / \sum_t \alpha_t$,



(a) MAE (b) MAE + LS (c) MAE + WD

(d) CE (e) CE + LS (f) CE + NLS

Figure 2: Training dynamics with various loss functions and regularizers on CIFAR10 with symmetric noise $\eta = 0.4$. We mark the median (dashed lines) of each distribution, and include the test accuracy of each sampled epoch.

reflects the overall emphasis of clean samples during training. In Table 3, loss functions with robustness guarantees have higher ρ in general. In addition, higher ρ generally corresponds to less performance drop under label noise, i.e., being more robust empirically.

A monotonically increasing left tail of $w(\boldsymbol{s}, y)$ sifts samples via the memorization effect. Learning is determined by gradients estimated using samples with nontrivial weights. In Fig. 1a, $\Delta(\boldsymbol{s}, y)$ of samples are randomly initialized around the left tail of $w_{\mathrm{MAE}}(\boldsymbol{s}, y)$. The similar or random gradient directions among clean or corrupted samples enhance or cancel each other, respectively Chatterjee & Zielinski (2022), leading to the assimilation of generalizable patterns during the early learning stage, when clean samples quickly improves and receive more weights, which facilitate even faster learning. Conversely, corrupted samples are unlearned due to generalization from clean samples and improve slowly with lower sample weights. As $\lim_{\Delta(\boldsymbol{s},y)\to-\infty} w_{\mathrm{MAE}}(\boldsymbol{s}, y) = 0$, unlearned samples can be ignored with minimal weights. Unlike $w_{\mathrm{MAE}}(\boldsymbol{s}, y)$, in Fig. 1c, $w_{\mathrm{CE}}(\boldsymbol{s}, y)$ emphasizes samples that are not well-learned, typically the corrupted ones, thus risk overfitting as shown in Fig. 1d. See Fig. 6 and 7 in Appendix B for similar results with different loss functions and noise settings.

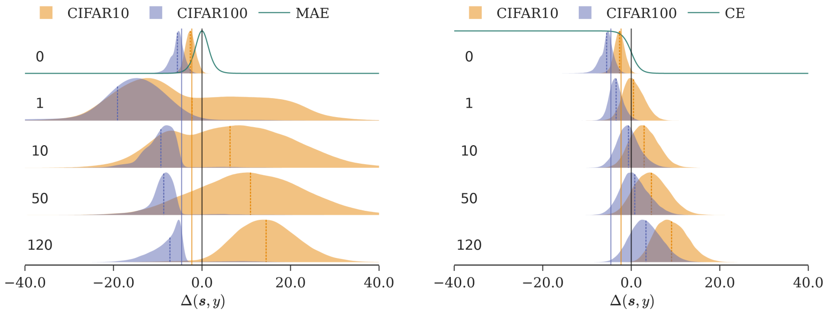
Sample-weighting functions of the loss functions considered share a monotonically decreasing right tail with $\lim_{\Delta(\boldsymbol{s},y)\to\infty} w(\boldsymbol{s}, y) = 0$. Although it prevents unbounded updates of well-learned samples and ensures the convergence of gradient descent training, it is also responsible for the shift into the noise overfitting stage. Despite initial dominance of the expected gradient, the faster learning clean samples will eventually receive minimal weights. Corrupted samples then come into dominance due to their relatively larger weights, when further training leads to overfitting, as shown by the synchronous drop of ρ_t and test accuracies in Fig. 1d. Such explanation complements the theoretical analysis for binary linear classification in Liu et al. (2020), which reach similar conclusion that corrupted samples dominate the expected gradients in the late training stage. Regularizers preventing the overfitting phase, such as early stopping (Song et al., 2019), label smoothing (Lukasik et al., 2020) or temporal ensembling (Liu et al., 2020), can thus improve noise robustness.

4.2 Effects of Regularization

We examine how the extracted regularizers in Table 2 help noise-robust learning with different loss functions. They either promote (with $\Delta(\boldsymbol{s}^*, y) = \pm\infty$) or reduce (with $\Delta(\boldsymbol{s}^*, y) = 0$) the prediction confidence p_y. We show benefits of proper regularization when using robust loss functions, and further resolve the apparent contradiction that they both improve noise robustness of CE (Lukasik et al.,

	CIFAR10						CIFAR100					
	Clean $\eta = 0$			Symmetric $\eta = 0.4$			Clean $\eta = 0$			Symmetric $\eta = 0.4$		
Loss	Train	Test	α^*	Train	Test	α^*	Train	Test	α^*	Train	Test	α^*
CE	99.96	93.24	2.27	80.05	65.66	5.48	99.96	77.36	5.37	99.89	48.45	7.19
SCE	99.96	93.74	2.35	80.29	64.01	5.61	99.96	77.21	5.53	99.88	48.50	7.24
FL	99.89	93.08	1.94	80.14	66.14	5.38	99.93	77.70	5.08	99.84	52.87	6.71
AEL	99.88	92.97	2.01	77.05	67.31	3.75	99.51	76.75	2.05	67.92	62.07	1.63
AGCE	99.70	93.33	1.88	66.21	83.80	2.44	94.71	72.37	1.76	61.88	62.43	1.40
AUL	99.90	93.49	1.99	75.51	69.39	3.39	99.11	75.67	1.74	65.30	61.18	1.41
GCE	99.33	92.82	1.43	63.16	86.62	1.60	97.39	74.51	1.55	61.36	64.07	1.20
TCE	99.61	92.82	1.63	64.76	84.80	1.94	71.42	57.95	0.87	39.50	43.09	0.74
MAE	98.00	92.00	0.68	59.81	87.61	0.52	8.56	8.25	0.06	3.47	4.70	0.08

Table 4: Robust loss functions can underfit CIFAR100 but CIFAR10. We report training and test accuracies and $\alpha^* = \sum_t \alpha_t^*$ (scaled by 1000) at the final step under different noise settings averaged with 3 different runs. Hyperparameters tuned on CIFAR100 with symmetric noise $\eta = 0.4$ are listed in Table 7 in Appendix B.



(a) MAE

(b) CE

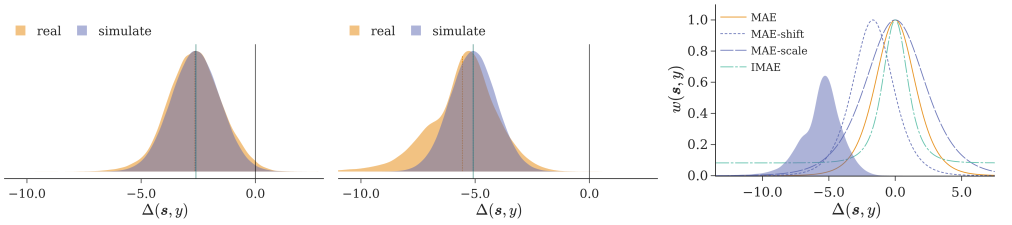
Figure 3: Training dynamics of clean CIFAR100 and CIFAR10 with MAE and CE. We mark the median (dashed line) of $\Delta(\boldsymbol{s}, y)$ distributions and $\Delta(\boldsymbol{s}, y) = -\log k$ (solid line).

2020; Wei et al., 2021; Cheng et al., 2021). As weight decay constrains $\Delta(\boldsymbol{s}, y)$ towards $-\log k$, it is disabled in this section except explicitly noted. We present results with typical regularizers on CIFAR10 and leave additional results to Fig. 8 in Appendix B.

Confidence reducing regularizers can improve convergence rate and training stability[1] of robust loss functions. In Fig. 2a, a random subset of samples are assign with large weights at initialization with $w_{\text{MAE}}(\boldsymbol{s}, y)$. With a high noise rate, the false generalization from these faster learning samples keep moving other samples into the low weight region, resulting in much spread and mixed $\Delta(\boldsymbol{s}, y)$ distributions of clean and corrupted samples. In contrast, with confidence reducing regularizers, samples get dragged towards $\Delta(\boldsymbol{s}, y) = -\log 10$ that corresponds to moderate sample weights, helping the accidentally unlearned samples move to high weight region and preventing overfitting faster learning samples, leading to compact and well separated $\Delta(\boldsymbol{s}, y)$ distributions in Fig. 2b and 2c.

Both types of regularizers abate the emphasis of $w_{\text{CE}}(\boldsymbol{s}, y)$ on slow-learning samples, typically the corrupted ones with peculiar patterns. In Fig. 2e, confidence-reducing regularizers shrink $\Delta(\boldsymbol{s}, y)$ towards $-\log k$, hampering the learning of corrupted samples. Conversely, in Fig. 2f, confidence-promoting regularizers preserve the generalization from the faster learning clean samples, preventing the learning of corrupted samples from distorting the correct generalization. Such explanation complements the intuition that increasing prediction confidence improves noise robustness (Wei et al., 2021; Cheng et al., 2021). However, as confidence promoting regularizers lead to unbounded parameter updates, they should be combined with early stopping to avoid numerical overflow.

[1] We observe that MAE without regularization can underfit on CIFAR10 with $\eta = 0.8$ under some random initializations. In contrast, all experiments with proper weight decay stably achieve similars performance.



(a) CIFAR10

(b) CIFAR100

(c) Shifted/Scaled MAE and IMAE

Figure 4: (a-b): comparisons between the simulated $\Delta(\boldsymbol{s}, y)$ with $s_i \sim \mathcal{N}(0, 1)$ and real $\Delta(\boldsymbol{s}, y)$ distributions at random initialization. The estimated $\mathbb{E}_k[\Delta(\boldsymbol{s}, y)]$ is marked with green vertical lines. (c): w_{IMAE} and the shifted and scaled w_{MAE} of MAE with hyperparamters $T = 10$ and $\tau = 3.4$. The $\Delta(\boldsymbol{s}, y)$ distribution of CIFAR100 at random initialization is included for reference.

4.3 Dissecting the Underfitting Issue

Robust loss functions are notorious for underfitting difficult tasks (Song et al., 2020). By comparing MAE with CE, Zhang & Sabuncu (2018) attribute underfitting of MAE to the lack of the $1/p_y$ term in sample gradients, which "treats every sample equally" and thus hampers learning. In contrast, we show that MAE emphasizes samples with moderate $\Delta(\boldsymbol{s}, y)$ which is also observed by Wang et al. (2019a). Ma et al. (2020) attribute underfitting to failure in balancing the active ($\forall i \neq y,\ l(\boldsymbol{s}, i) = 0$) and passive ($\exists i \neq y,\ l(\boldsymbol{s}, i) \neq 0$) parts of loss functions $L(\boldsymbol{s}, y) = \sum_{i=1}^{k} l(\boldsymbol{s}, i)$. However, ambiguities arise when specifying $l(\boldsymbol{s}, i)$: given $L_{\text{MAE}}(\boldsymbol{s}, y) \propto \sum_i |\mathbb{I}(i = y) - p_i| \propto \sum_i \mathbb{I}(i = y)(1 - p_i)$ where $\mathbb{I}(\cdot)$ is the indicator function, MAE can be active with $l(\boldsymbol{s}, i) = \mathbb{I}(i = y)(1 - p_i)$ but passive with $l(\boldsymbol{s}, i) = |\mathbb{I}(i = y) - p_i|$. Wang et al. (2019a) view $\|\nabla_{\boldsymbol{s}} L(\boldsymbol{s}, y)\|_1$ as weights for sample gradients and argue that their low variance makes informative and uninformative samples less distinguishable. However, it is unclear how low variance of gradient magnitudes leads to underfitting.

Alternatively, for loss functions conforming to Eq. (3), we show that underfitting results from minimal sample weights at initialization. The scale of parameter updates

$$\|\boldsymbol{\theta}_t - \boldsymbol{\theta}_{t-1}\|_p \leq \|\boldsymbol{\theta}_t - \boldsymbol{\theta}_{t-1}\|_1 \leq \alpha_t \cdot \|\nabla_{\boldsymbol{s}} L(\boldsymbol{s}, y)\|_1 \cdot \|\nabla_{\boldsymbol{\theta}} \boldsymbol{s}(\boldsymbol{x}; \boldsymbol{\theta})\|_1 = 2\alpha_t w(\boldsymbol{s}, y) \cdot \|\nabla_{\boldsymbol{\theta}} \boldsymbol{s}(\boldsymbol{x}; \boldsymbol{\theta})\|_1\,,$$

is affected by the effective learning rate $\alpha_t^* = 2\alpha_t w(\boldsymbol{s}, y)$ with various loss functions. Loss functions that underfit correspond to negligible cumulative effective learning rate $\alpha^* = \sum_t \alpha_t^*$ in Table 4. As shown in Fig. 3, since CIFAR100 has much smaller $\Delta(\boldsymbol{s}, y)$ at initialization, with robust loss functions like MAE, only a limited random proportion of samples receive noneligible weights, leading to a less representative estimation of the expected gradient. The majority of samples thus get stuck in the region of low sample weights. Weight decay helps little as $\Delta(\boldsymbol{s}^*, y) = -\log 100$ corresponds to negligible sample weights. These effects add up to the underfitting of MAE on CIFAR100.

The underfitting issue on CIFAR100 instead of CIFAR10 has been vaguely attributed to the increased difficulty of CIFAR100. We further dissect what makes a task difficult. In particular, the increased number of classes leads to smaller expected $\Delta(\boldsymbol{s}, y)$ at initialization. Assume that class scores s_i are i.i.d. Gaussian variables $s_i \sim \mathcal{N}(\mu, \sigma)$ at initialization. Typically, we have $\mu = 0$ and $\sigma = 1$ with standard settings (Glorot & Bengio, 2010; He et al., 2015; Ioffe & Szegedy, 2015). The expected $\Delta(\boldsymbol{s}, y)$ with number of classes k can be approximated with

$$\mathbb{E}_k[\Delta(\boldsymbol{s}, y)] \approx -\log(k-1) - \sigma^2/2 + \frac{e^{\sigma^2} - 1}{2(k-1)} \tag{5}$$

where $\mathbb{E}_k[\Delta(\boldsymbol{s}, y)] < 0$ with $\sigma \approx 1$ and a large k, e.g., $k = 100$. See Fig. 4a and 4b for a comparison between the simulated and real $\Delta(\boldsymbol{s}, y)$ distributions at random initialization. We leave derivations and more comparisons between our assumptions and real settings to Appendix B. Combined with the decreased $\Delta(\boldsymbol{s}^*, y) = -\log k$ of the confidence reducing regularizers, initial learning with $\lim_{\Delta(\boldsymbol{s},y) \to -\infty} w(\boldsymbol{s}, y) = 0$ can get stuck with minimal sample weights, thus leading to underfitting.

Our analysis suggests that the fixed sample-weighting function $w(\boldsymbol{s}, y)$ is to blame for underfitting. We can thus morph the weighting functions for nontrivial initial sample weights, which can be simply achieved by scaling

$$w^*(\boldsymbol{s}, y) = w^*(\Delta(\boldsymbol{s}, y)) = w(\Delta(\boldsymbol{s}, y)/|\mathbb{E}_k[\Delta(\boldsymbol{s}, y)]| \cdot \tau)$$

Loss	Clean	Symmetric		Asymmetric	Human
	$\eta = 0$	$\eta = 0.4$	$\eta = 0.8$	$\eta = 0.4$	$\eta = 0.4$
CE‡	71.33 ± 0.43	39.92 ± 0.10	7.59 ± 0.20	40.17 ± 1.31	/
GCE‡	63.09 ± 1.39	56.11 ± 1.35	17.42 ± 0.06	40.91 ± 0.57	/
NCE+AGCE‡	69.03 ± 0.37	59.47 ± 0.36	24.72 ± 0.60	43.76 ± 0.70	/
CE	$\mathbf{77.44 \pm 0.13}$	49.96 ± 0.02	11.23 ± 0.45	45.73 ± 0.49	54.40 ± 0.35
GCE	73.88 ± 0.25	64.67 ± 0.49	23.90 ± 2.69	45.14 ± 0.13	56.95 ± 0.63
NCE+AGCE	76.37 ± 0.25	64.55 ± 0.46	$\mathbf{26.19 \pm 1.14}$	40.93 ± 1.22	53.67 ± 0.18
TCE	58.04 ± 1.15	45.91 ± 1.25	20.47 ± 1.45	28.35 ± 0.74	32.22 ± 1.22
TCE shift, $\tau = 4.2$	77.14 ± 0.11	60.17 ± 0.47	18.16 ± 0.37	44.56 ± 0.71	55.07 ± 0.19
TCE scale, $\tau = 4.2$	75.79 ± 0.17	62.88 ± 0.59	20.78 ± 1.53	43.57 ± 1.19	56.13 ± 0.22
MAE	7.46 ± 1.92	4.65 ± 1.55	3.21 ± 0.57	1.61 ± 0.53	1.54 ± 0.47
IMAE, $T = 10$	5.26 ± 3.19	45.01 ± 1.10	4.51 ± 0.69	48.09 ± 0.68	2.85 ± 1.05
MAE shift, $\tau = 3.4$	76.65 ± 0.30	61.29 ± 0.49	19.30 ± 1.00	44.06 ± 1.23	54.83 ± 0.49
MAE scale, $\tau = 3.4$	73.54 ± 0.32	$\mathbf{64.92 \pm 0.20}$	23.00 ± 2.44	$\mathbf{48.88 \pm 0.79}$	$\mathbf{57.56 \pm 0.41}$

Table 5: Shifting or scaling $w(\boldsymbol{s}, y)$ mitigates underfitting on CIFAR100 under different label noise. We report test accuracies with 3 different runs. Both τ of the shifted/scaled MAE/TCE and T of IMAE are tuned with symmetric noise $\eta = 0.4$. Previous best results from Zhou et al. (2021) are included as context (denoted with ‡). See Appendix B for results with more noise rates.

Settings	CE	MAE	IMAE	MAE shift	MAE scale	TCE	TCE shift	TCE scale
$k = 50$	66.40	3.68		60.76	66.72			
$k = 200$	70.26	0.50		59.31	71.92			
$k = 400$	70.16	0.25		47.32	71.87			

Table 6: Shifting or scaling $w(\boldsymbol{s}, y)$ mitigates underfitting on WebVision subsampled with different numbers of classes. $k = 50$ is the standard "mini" setting in previous work (Ma et al., 2020; Zhou et al., 2021). Hyperparameters roughly tuned for each setting are left to Table 10 of Appendix B. We report test accuracy with a single run due to a limited computation budget.

or shifting

$$w^{+}(\boldsymbol{s}, y) = w^{+}(\Delta(\boldsymbol{s}, y)) = w(\Delta(\boldsymbol{s}, y) + |\mathbb{E}_k[\Delta(\boldsymbol{s}, y)]| - \tau)$$

the sample-weighting functions, where τ is a hyperparameter. Intuitively, these approaches cancel the effect of large k on the weight of $\mathbb{E}_k[\Delta(\boldsymbol{s}, y)]$ at initialization. Small τ thus lead to high initial sample weights regardless of k. Alternatively, Wang et al. (2019a) propose IMAE as *a fix to MAE*, which essentially applies a scaled exponential transform to $w_{\text{MAE}}(\boldsymbol{s}, y)$

$$w_{\text{IMAE}}(\boldsymbol{s}, y) = e^{T p_y (1 - p_y)}$$

where $T \geq 0$ is a hyperparameter. We visualize scaled and shifted sample-weighting functions of MAE and w_{IMAE}[2] in Fig. 4c. Notably, $\lim_{\Delta(\boldsymbol{s}, y) \to \infty} w_{\text{IMAE}}(\boldsymbol{s}, y) > 0$, which can lead to excessive learning of well-learned samples and thus risk overfitting. We report results on CIFAR100 with different label noise in Table 5, and results on the noisy large-scale dataset WebVision in Table 6. In summary, both shifting and scaling significantly improve performance of MAE, thus alleviating underfitting, making MAE comparable to the previous state-of-the-art.

5 Conclusion

We unify a broad array of loss functions into the same standard form, explicitly connecting the design of loss functions to the design of sample-weighting curricula. Based on the curriculum view, we gain more insights into how different design choices affect the training dynamics, especially the noise robustness and vulnerability to underfitting. Our theoretical and empirical findings can help design better loss functions and learning curricula in future work.

[2] w_{IMAE} is also normalized to unit maximum.

REFERENCES

Ehsan Amid, Manfred K. Warmuth, Rohan Anil, and Tomer Koren. Robust bi-tempered logistic loss based on bregman divergences, 2019.

Devansh Arpit, Stanislaw Jastrzebski, Nicolas Ballas, David Krueger, Emmanuel Bengio, Maxinder S. Kanwal, Tegan Maharaj, Asja Fischer, Aaron C. Courville, Yoshua Bengio, and Simon Lacoste-Julien. A closer look at memorization in deep networks. In Doina Precup and Yee Whye Teh (eds.), *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pp. 233–242. PMLR, 2017. URL `http://proceedings.mlr.press/v70/arpit17a.html`.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In Andrea Pohoreckyj Danyluk, Léon Bottou, and Michael L. Littman (eds.), *Proceedings of the 26th Annual International Conference on Machine Learning, ICML 2009, Montreal, Quebec, Canada, June 14-18, 2009*, volume 382 of *ACM International Conference Proceeding Series*, pp. 41–48. ACM, 2009. doi: 10.1145/1553374.1553380. URL `https://doi.org/10.1145/1553374.1553380`.

Haw-Shiuan Chang, Erik G. Learned-Miller, and Andrew McCallum. Active bias: Training more accurate neural networks by emphasizing high variance samples. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pp. 1002–1012, 2017. URL `https://proceedings.neurips.cc/paper/2017/hash/2f37d10131f2a483a8dd005b3d14b0d9-Abstract.html`.

Satrajit Chatterjee and Piotr Zielinski. On the generalization mystery in deep learning. *ArXiv preprint*, abs/2203.10036, 2022. URL `https://arxiv.org/abs/2203.10036`.

Pengfei Chen, Benben Liao, Guangyong Chen, and Shengyu Zhang. Understanding and utilizing deep neural networks trained with noisy labels. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pp. 1062–1070. PMLR, 2019. URL `http://proceedings.mlr.press/v97/chen19g.html`.

Hao Cheng, Zhaowei Zhu, Xingyu Li, Yifei Gong, Xing Sun, and Yang Liu. Learning with instance-dependent label noise: A sample sieve approach. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. URL `https://openreview.net/forum?id=2VXyy9mIyU3`.

Barry Cobb, Rafael Rumí, and Antonio Salmerón. Approximating the distribution of a sum of log-normal random variables. 2012.

Erik Englesson and Hossein Azizpour. Generalized jensen-shannon divergence loss for learning with noisy labels. (arXiv:2105.04522), October 2021. doi: 10.48550/arXiv.2105.04522. URL `http://arxiv.org/abs/2105.04522`. arXiv:2105.04522 [cs, stat].

Lei Feng, Senlin Shu, Zhuoyi Lin, Fengmao Lv, Li Li, and Bo An. Can cross entropy loss be robust to label noise? In Christian Bessiere (ed.), *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020*, pp. 2206–2212. ijcai.org, 2020. doi: 10.24963/ijcai.2020/305. URL `https://doi.org/10.24963/ijcai.2020/305`.

Aritra Ghosh, Himanshu Kumar, and P. S. Sastry. Robust loss functions under label noise for deep neural networks. In Satinder P. Singh and Shaul Markovitch (eds.), *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA*, pp. 1919–1925. AAAI Press, 2017. URL `http://aaai.org/ocs/index.php/AAAI/AAA,I17/paper/view/14759`.

Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pp. 249–256. JMLR Workshop and Conference Proceedings, 2010. URL https://proceedings.mlr.press/v9/glorot10a.html.

Siddharth Gopal. Adaptive sampling for SGD by exploiting side information. In Maria-Florina Balcan and Kilian Q. Weinberger (eds.), *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, volume 48 of *JMLR Workshop and Conference Proceedings*, pp. 364–372. JMLR.org, 2016. URL http://proceedings.mlr.press/v48/gopal16.html.

Guy Hacohen and Daphna Weinshall. On the power of curriculum learning in training deep networks. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2535–2544. PMLR, 2019. URL http://proceedings.mlr.press/v97/hacohen19a.html.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, pp. 1026–1034. IEEE Computer Society, 2015. doi: 10.1109/ICCV.2015.123. URL https://doi.org/10.1109/ICCV.2015.123.

Jinchi Huang, Lie Qu, Rongfei Jia, and Binqiang Zhao. O2u-net: A simple noisy label detection approach for deep neural networks. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pp. 3325–3333. IEEE, 2019. doi: 10.1109/ICCV.2019.00342. URL https://doi.org/10.1109/ICCV.2019.00342.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Francis R. Bach and David M. Blei (eds.), *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, pp. 448–456. JMLR.org, 2015. URL http://proceedings.mlr.press/v37/ioffe15.html.

Lu Jiang, Zhengyuan Zhou, Thomas Leung, Li-Jia Li, and Li Fei-Fei. Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels. In Jennifer G. Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pp. 2309–2318. PMLR, 2018. URL http://proceedings.mlr.press/v80/jiang18c.html.

Yoshihide Kato and Shigeki Matsubara. Correcting errors in a treebank based on synchronous tree substitution grammar. In *Proceedings of the ACL 2010 Conference Short Papers*, pp. 74–79, Uppsala, Sweden, 2010. Association for Computational Linguistics. URL https://aclanthology.org/P10-2014.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, 2009.

M. Pawan Kumar, Benjamin Packer, and Daphne Koller. Self-paced learning for latent variable models. In John D. Lafferty, Christopher K. I. Williams, John Shawe-Taylor, Richard S. Zemel, and Aron Culotta (eds.), *Advances in Neural Information Processing Systems 23: 24th Annual Conference on Neural Information Processing Systems 2010. Proceedings of a meeting held 6-9 December 2010, Vancouver, British Columbia, Canada*, pp. 1189–1197. Curran Associates, Inc., 2010. URL https://proceedings.neurips.cc/paper/2010/hash/e57c6b956a6521b28495f2886ca0977a-Abstract.html.

Junnan Li, Richard Socher, and Steven C. H. Hoi. Dividemix: Learning with noisy labels as semi-supervised learning, 2020.

Wen Li, Limin Wang, Wei Li, Eirikur Agustsson, and Luc Van Gool. Webvision database: Visual learning and understanding from web data, 2017. URL https://arxiv.org/abs/1708.02862.

Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pp. 2999–3007. IEEE Computer Society, 2017. doi: 10.1109/ICCV.2017.324. URL https://doi.org/10.1109/ICCV.2017.324.

Kun Liu, Yao Fu, Chuanqi Tan, Mosha Chen, Ningyu Zhang, Songfang Huang, and Sheng Gao. Noisy-labeled NER with confidence estimation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 3437–3445, Online, 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.269. URL https://aclanthology.org/2021.naacl-main.269.

Sheng Liu, Jonathan Niles-Weed, Narges Razavian, and Carlos Fernandez-Granda. Early-learning regularization prevents memorization of noisy labels, 2020.

Yang Liu and Hongyi Guo. Peer loss functions: Learning from noisy labels without knowing noise rates. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pp. 6226–6236. PMLR, 2020. URL http://proceedings.mlr.press/v119/liu20e.html.

Ilya Loshchilov and Frank Hutter. Online batch selection for faster training of neural networks. *ArXiv preprint*, abs/1511.06343, 2015. URL https://arxiv.org/abs/1511.06343.

Michal Lukasik, Srinadh Bhojanapalli, Aditya Krishna Menon, and Sanjiv Kumar. Does label smoothing mitigate label noise? In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pp. 6448–6458. PMLR, 2020. URL http://proceedings.mlr.press/v119/lukasik20a.html.

Xingjun Ma, Hanxun Huang, Yisen Wang, Simone Romano, Sarah M. Erfani, and James Bailey. Normalized loss functions for deep learning with noisy labels. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pp. 6543–6553. PMLR, 2020. URL http://proceedings.mlr.press/v119/ma20c.html.

Mengye Ren, Wenyuan Zeng, Bin Yang, and Raquel Urtasun. Learning to reweight examples for robust deep learning. In Jennifer G. Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pp. 4331–4340. PMLR, 2018. URL http://proceedings.mlr.press/v80/ren18a.html.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *arXiv:1409.0575 [cs]*, 2015. URL http://arxiv.org/abs/1409.0575. arXiv: 1409.0575.

Hwanjun Song, Minseok Kim, Dongmin Park, and Jae-Gil Lee. How does early stopping help generalization against label noise?, 2019. URL https://arxiv.org/abs/1911.08059.

Hwanjun Song, Minseok Kim, Dongmin Park, Yooju Shin, and Jae-Gil Lee. Learning from noisy labels with deep neural networks: A survey. *ArXiv preprint*, abs/2007.08199, 2020. URL https://arxiv.org/abs/2007.08199.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pp. 2818–2826. IEEE Computer Society, 2016. doi: 10.1109/CVPR.2016.308. URL https://doi.org/10.1109/CVPR.2016.308.

Yee Whye Teh, David Newman, and Max Welling. A collapsed variational bayesian inference algorithm for latent dirichlet allocation. In Bernhard Schölkopf, John C. Platt, and Thomas Hofmann (eds.), *Advances in Neural Information Processing Systems*

19, Proceedings of the Twentieth Annual Conference on Neural Information Processing Systems, Vancouver, British Columbia, Canada, December 4-7, 2006, pp. 1353–1360. MIT Press, 2006. URL https://proceedings.neurips.cc/paper/2006/hash/532b7cbe070a3579f424988a040752f2-Abstract.html.

Xin Wang, Yudong Chen, and Wenwu Zhu. A survey on curriculum learning. *ArXiv preprint*, abs/2010.13166, 2020. URL https://arxiv.org/abs/2010.13166.

Xinshao Wang, Yang Hua, Elyor Kodirov, and Neil M. Robertson. Imae for noise-robust learning: Mean absolute error does not treat examples equally and gradient magnitude's variance matters. *ArXiv preprint*, abs/1903.12141, 2019a. URL https://arxiv.org/abs/1903.12141.

Xinshao Wang, Elyor Kodirov, Yang Hua, and Neil M. Robertson. Derivative manipulation for general example weighting. *ArXiv preprint*, abs/1905.11233, 2019b. URL https://arxiv.org/abs/1905.11233.

Yisen Wang, Xingjun Ma, Zaiyi Chen, Yuan Luo, Jinfeng Yi, and James Bailey. Symmetric cross entropy for robust learning with noisy labels. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pp. 322–330. IEEE, 2019c. doi: 10.1109/ICCV.2019.00041. URL https://doi.org/10.1109/ICCV.2019.00041.

Jiaheng Wei and Yang Liu. When optimizing f-divergence is robust with label noise, 2021.

Jiaheng Wei, Hangyu Liu, Tongliang Liu, Gang Niu, Masashi Sugiyama, and Yang Liu. To smooth or not? when label smoothing meets noisy labels. *ArXiv preprint*, abs/2106.04149, 2021. URL https://arxiv.org/abs/2106.04149.

Jiaheng Wei, Zhaowei Zhu, Hao Cheng, Tongliang Liu, Gang Niu, and Yang Liu. Learning with noisy labels revisited: A study using real-world human annotations. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=TBWA6PLJZQm.

Xuan Zhang, Gaurav Kumar, Huda Khayrallah, Kenton Murray, Jeremy Gwinnup, Marianna J. Martindale, Paul McNamee, Kevin Duh, and Marine Carpuat. An empirical exploration of curriculum learning for neural machine translation. *ArXiv preprint*, abs/1811.00739, 2018. URL https://arxiv.org/abs/1811.00739.

Zhilu Zhang and Mert R. Sabuncu. Generalized cross entropy loss for training deep neural networks with noisy labels. In Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pp. 8792–8802, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/f2925f97bc13ad2852a7a551802feea0-Abstract.html.

Xiong Zhou, Xianming Liu, Junjun Jiang, Xin Gao, and Xiangyang Ji. Asymmetric loss functions for learning with noisy labels. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pp. 12846–12856. PMLR, 2021. URL http://proceedings.mlr.press/v139/zhou21f.html.

A Review and Derivations of Loss Functions

We provide a comprehensive review of loss functions in Tables 1 and 2. We ignore the differences in constant scaling factors and additive bias as they can be absorb into other hyperparameters. We examine how hyperparameters affect different sample-weighting functions in Fig. 5.

A.1 Loss Functions without Robustness Guarantees

We first review loss functions not satisfying Eq. (1).

Cross Entropy (CE)

$$L_{\mathrm{CE}}(\boldsymbol{s}, y) = -\log p_y$$

is the standard loss function for classification.

Focal Loss (FL; Lin et al. 2017)

$$L_{\mathrm{FL}}(\boldsymbol{s}, y) = -(1-p_y)^q \log p_y$$

aims to address label imbalance when training object detection models.

Symmetric Cross Entropy (SCE; Wang et al. 2019c)

$$\begin{aligned} L_{\mathrm{SCE}}(\boldsymbol{s}, y) &= a \cdot L_{\mathrm{CE}}(\boldsymbol{s}, y) + b \cdot L_{\mathrm{RCE}}(\boldsymbol{s}, y) \\ &\propto (1-q) \cdot (-\log p_y) + q \cdot (1-p_y) \end{aligned}$$

is a weighted average of CE and RCE (MAE), where $a > 0$, $b > 0$, and $0 < q < 1$. It exhibit better noise-robustness due to the combination with RCE (MAE).

A.2 Symmetric Loss Functions

A loss function L is *symmetric* (Ghosh et al., 2017) if

$$\sum_{i=1}^{k} L(\boldsymbol{s}, i) = C, \ \forall \boldsymbol{s} \in \mathbb{R}^k,$$

with a constant C. It is proved to be robust against *symmetric* label noise when $\eta < (k-1)/k$.

Mean Absolute Error (MAE; Ghosh et al. 2017)

$$L_{\mathrm{MAE}}(\boldsymbol{s}, y) = \sum_{i=1}^{k} |\mathbb{I}(i = y) - p_i| = 2 - 2p_y \propto 1 - p_y$$

is a classic symmetric loss function, where $\mathbb{I}(i = y)$ is the indicator function.

Reverse Cross Entropy (RCE; Wang et al. 2019c)

$$L_{\mathrm{RCE}}(\boldsymbol{s}, y) = -\sum_{i=1}^{k} p_i \log \mathbb{I}(i = y) = -\sum_{i \neq y} p_i \log 0 = -(1-p_y)A \propto 1 - p_y = L_{\mathrm{MAE}}(\boldsymbol{s}, y)$$

is equivalent to MAE in implementation, where $\mathbb{I}(\cdot)$ is the indicator function and $\log 0$ is truncated to a negative constant A to avoid numerical overflow.

Ma et al. (2020) argued that any generic loss functions with $L(\boldsymbol{s}, i) > 0, \forall i \in \{1, ..., k\}$ can become symmetric by simply normalizing them. As an example,

Normalized Cross Entropy (NCE; Ma et al. 2020)

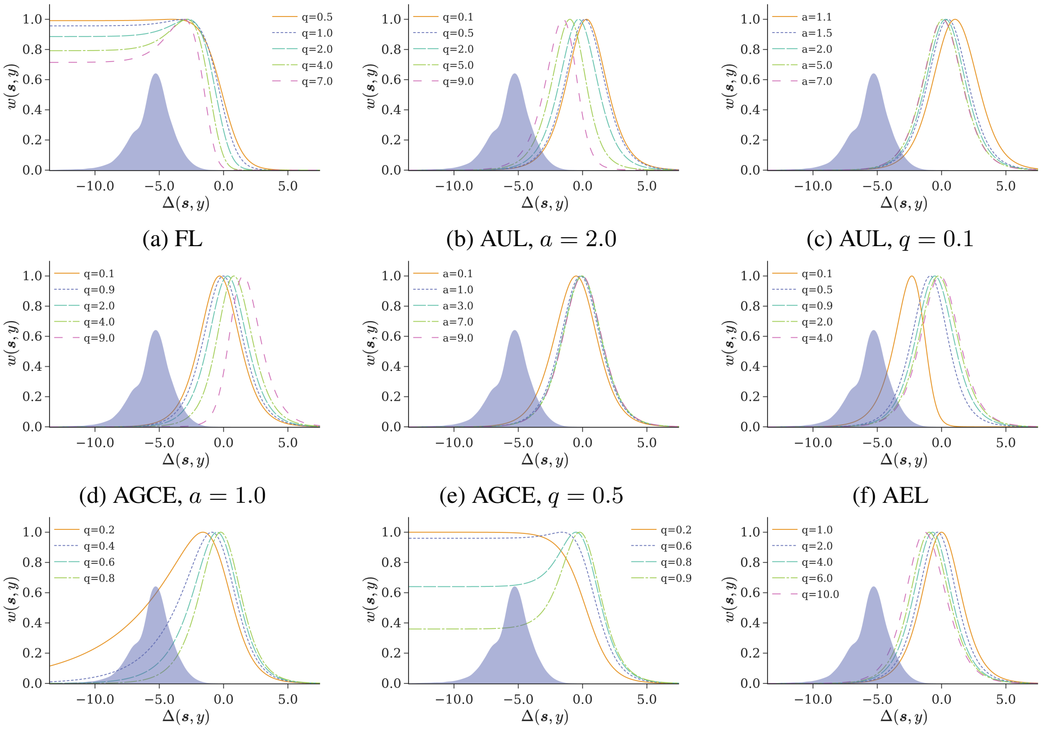
$$L_{\mathrm{NCE}}(\boldsymbol{s}, y) = \frac{L_{\mathrm{CE}}(\boldsymbol{s}, y)}{\sum_{i=1}^{k} L_{\mathrm{CE}}(\boldsymbol{s}, i)} = \frac{-\log p_y}{\sum_{i=1}^{k} -\log p_i}$$

is a symmetric loss function.

A.3 Asymmetric Loss Functions

L as a function of the softmax probability p_i, $L(\boldsymbol{s}, i) = l(p_i)$, is *asymmetric* (Zhou et al., 2021) if

$$\tilde{r} = \max_{i \neq y} \frac{P(\tilde{y} = i|\boldsymbol{x}, y)}{P(\tilde{y} = y|\boldsymbol{x}, y)} \leq \inf_{\substack{0 \leq p_i, p_j \leq 1 \\ p_i + p_j \leq 1}} \frac{l(p_i) - l(p_i + p_j)}{l(0) - l(p_j)},$$



(a) FL (b) AUL, $a = 2.0$ (c) AUL, $q = 0.1$

(d) AGCE, $a = 1.0$ (e) AGCE, $q = 0.5$ (f) AEL

(g) GCE (h) SCE (i) TCE

Figure 5: How hyperparameters affect the sample-weighting functions in Table 1. The initial $\Delta(\boldsymbol{s}, y)$ distributions of CIFAR100 extracted with a randomly initialized model are included as reference.

where p_j is a valid increment of p_i. An asymmetric loss function is robust against *generic* label noise when $\tilde{r} < 1$, i.e., there are more clean samples than corrupted samples. Zhou et al. (2021) propose the following asymmetric loss functions with conditions to satisfy the asymmetric criterion:

Asymmetric Generalized Cross Entropy (AGCE)

$$L_{\text{AGCE}}(\boldsymbol{s}, y) = \frac{(a+1) - (a+p_y)^q}{q}$$

where $a > 0$ and $q > 0$. It is asymmetric when $\mathbb{I}(q \leq 1)(\frac{a+1}{a})^{1-q} + \mathbb{I}(q > 1) \leq 1/\tilde{r}$.

Asymmetric Unhinged Loss (AUL)

$$L_{\text{AUL}}(\boldsymbol{s}, y) = \frac{(a - p_y)^q - (a-1)^q}{q}$$

where $a > 1$ and $q > 0$. It is asymmetric when $\mathbb{I}(q \leq 1)(\frac{a}{a-1})^{q-1} + \mathbb{I}(q \leq 1) \leq 1/\tilde{r}$.

Asymmetric Exponential Loss (AEL)

$$L_{\text{AEL}}(\boldsymbol{s}, y) = e^{-p_y/q}$$

where $q > 0$. It is asymmetric when $e^{1/q} \leq 1/\tilde{r}$.

A.4 Other Robust Loss Functions

Generalized Cross Entropy (GCE; Zhang & Sabuncu 2018)

$$L_{\text{GCE}}(\boldsymbol{s}, y) = \frac{1 - p_y^q}{q}$$

can be viewed as a smooth interpolation between CE and MAE, where $0 < q \leq 1$. CE or MAE can be recovered by setting $q \to 0$ or $q = 1$.

Taylor Cross Entropy (TCE; Feng et al. 2020)

$$L_{\mathrm{TCE}}(\boldsymbol{s}, y) = \sum_{i=1}^{q} \frac{(1-p_y)^i}{i}$$

is derived from Taylor series of the $\log$ function in CE. It reduces to MAE when $q = 1$. Interestingly, the summand of TCE $(1 - p_y)^i/i$ with $i > 2$ is proportional to AUL with $a = 1$ and $q = i$. Thus TCE can be viewed as a combination of symmetric and asymmetric loss functions.

Active-Passive Loss (APL; Ma et al. 2020) Ma et al. (2020) propose weighted combinations of active and passive loss functions. Active loss follows $\forall i \neq y,\ l(\boldsymbol{s}, i) = 0$, which emphasize the learning of labeled class. Passive loss complies with $\exists i \neq y,\ l(\boldsymbol{s}, i) \neq 0$, which focus on unlearning other classes. In their derivation, NCE is active while MAE is passive. However, ambiguities arise when determine whether $l(\boldsymbol{s}, i)$ is active or pasive: given

$$L_{\mathrm{MAE}}(\boldsymbol{s}, y) \propto \sum\nolimits_i |\mathbb{I}(i = y) - p_i| \propto \sum\nolimits_i \mathbb{I}(i = y)(1 - p_i)$$

with $\mathbb{I}(\cdot)$ the indicator function, MAE can be active with $l(\boldsymbol{s}, i) = \mathbb{I}(i = y)(1 - p_i)$ but passive with $l(\boldsymbol{s}, i) = |\mathbb{I}(i = y) - p_i|$. We include NCE+MAE as an example:

$$\begin{aligned} L_{\mathrm{NCE+MAE}}(\boldsymbol{s}, y) &= a \cdot L_{\mathrm{NCE}}(\boldsymbol{s}, y) + b \cdot L_{\mathrm{MAE}}(\boldsymbol{s}, y) \\ &\propto (1-q) \cdot \frac{-\log p_y}{\sum_{i=1}^{k} -\log p_i} + q \cdot (1 - p_y) \end{aligned}$$

where $a > 0$, $b > 0$, and $0 < q < 1$.

A.5 Loss Functions with Output Regularizers

We extract the output regularizers from existing loss functions.

Mean Square Error (MSE; Ghosh et al. 2017)

$$\begin{aligned} L_{\mathrm{MSE}}(\boldsymbol{s}, y) &= \sum_{i=1}^{k} (\mathbb{I}(i = y) - p_i)^2 = 1 - 2p_y + \sum_{i=1}^{k} p_i^2 \\ &\propto 1 - p_y + \frac{1}{2} \cdot \sum_{i=1}^{k} p_i^2 = L_{\mathrm{MAE}}(\boldsymbol{s}, y) + \alpha \cdot R_{\mathrm{MSE}}(\boldsymbol{s}) \end{aligned}$$

is more robust than CE (Ghosh et al., 2017), where $\alpha = 0.5$ and the regularizer

$$R_{\mathrm{MSE}}(\boldsymbol{s}) = \sum_{i=1}^{k} p_i^2 \tag{6}$$

increases the entropy of the softmax output. We can generalize α to a hyperparamter, making MSE a combination of MAE and an entropy regularizer R_{MSE}.

Peer Loss (PL; Liu & Guo 2020)

$$L_{\mathrm{PL}}(\boldsymbol{s}, y) = L(\boldsymbol{s}, y) - L(\mathbf{s}, \mathrm{y})$$

makes a generic loss function $L(\boldsymbol{s}, y)$ robust against label noise, where **s** denotes the score of an input **x** and y the label, both randomly sampled from the noisy data distribution $\mathcal{D}$. Its noise robustness is theoretically established for binary classification and extended to multi-class setting (Liu & Guo, 2020).

Confidence Regularizer (CR; Cheng et al. 2021)

$$R_{\mathrm{CR}}(\boldsymbol{s}) = -\mathbb{E}_{\mathrm{y} \sim \mathcal{D}, \mathbf{x} \sim \mathcal{D}}[\log p_{\mathrm{y}|\mathbf{x}}]$$

is shown (Cheng et al., 2021) to be the regularizer induced by PL in expectation. Minimizing $R_{\mathrm{CR}}(\boldsymbol{s})$ makes the softmax output distribution $\boldsymbol{p}$ deviate from the prior label distribution of the noisy dataset $P(\tilde{y} = i)$, reducing the entropy of the softmax output.

	CIFAR10		CIFAR100	
Loss	a	q	a	q
FL	/	3.0	/	3.0
AGCE	5.4	1.5	0.1	0.1
AUL	6.1	4.8	2.0	8.7
AEL	/	5.0	/	0.2
GCE	/	0.9	/	0.7
TCE	/	2	/	6
SCE	/	0.9	/	0.15

Table 7: Hyperparameters of loss functions for results in Table 3 and Table 4, tuned with symmetric label noise $\eta = 0.4$ on CIFAR10 and CIFAR100, respectively.

Generalized Label Smoothing (GLS; Wei et al. 2021)

Lukasik et al. (2020) show that label smoothing (LS; Szegedy et al. 2016) can mitigate overfitting with label noise, which is later extended to GLS. Cross entropy with GLS is

$$\begin{aligned} L_{\mathrm{CE+GLS}}(\boldsymbol{s}, y) &= \sum_{i=1}^{k} -[\mathbb{I}(i=y)(1-\alpha) + \frac{\alpha}{k}] \log p_i \\ &= -(1-\alpha)\log p_y - \alpha \cdot \frac{1}{k}\sum_{i=1}^{k} \log p_i \\ &\propto -\log p_y - \frac{\alpha}{1-\alpha} \cdot \frac{1}{k}\sum_{i=1}^{k} \log p_i = L_{\mathrm{CE}}(\boldsymbol{s}, y) + \alpha' \cdot R_{\mathrm{GLS}}(\boldsymbol{s}) \end{aligned}$$

where $\alpha' = \alpha/(1-\alpha)$, has regularizer R_{GLS}

$$R_{\mathrm{GLS}}(\boldsymbol{s}) = -\sum_{i=1}^{k} \frac{1}{k} \log p_i \tag{7}$$

With $\alpha' > 0$, R_{GLS} corresponds to the original LS, which increases the entropy of softmax outputs. In contrast, $\alpha' < 0$ corresponding to Negative Label Smoothing (NLS; Wei et al. 2021), which decreases the output entropy similar to R_{CR}.

Normalized Cross Entropy (NCE; Ma et al. 2020)

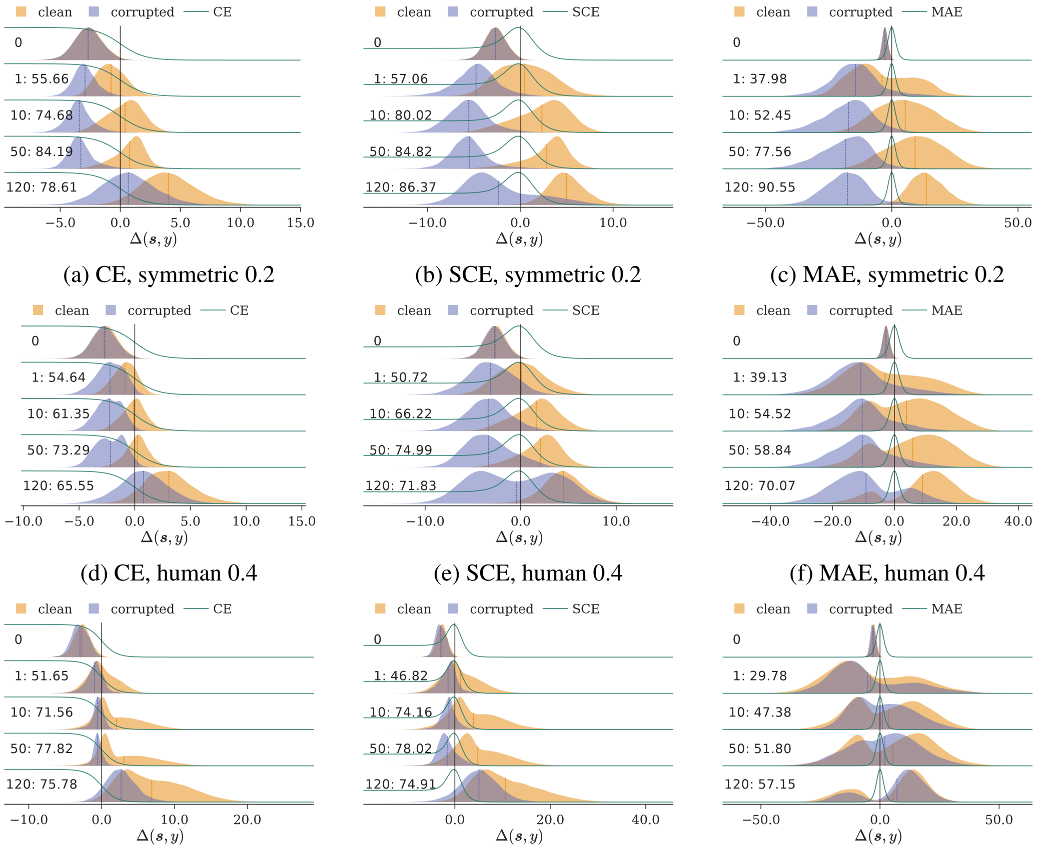
$$L_{\mathrm{NCE}}(\boldsymbol{s}, y) = \frac{L_{\mathrm{CE}}(\boldsymbol{s}, y)}{\sum_{i=1}^{k} L_{\mathrm{CE}}(\boldsymbol{s}, i)} = \frac{-\log p_y}{\sum_{i=1}^{k} -\log p_i}$$

has been reviewed in Appendix A.2. However, our derivations in §3 shows that it encompass an output regularizer

$$R_{\mathrm{NCE}}(\boldsymbol{s}) = \sum_{i=1}^{k} \frac{1}{k} \log p_i$$

which is equivalent to NLS. **Jensen-Shannon Divergence** (JS; Englesson & Azizpour, 2021)

$$\begin{aligned} L_{\mathrm{JS}}(\boldsymbol{s}, y) &= a\sum_i p_i \log \frac{p_i}{ap_i + (1-a)e_i} + (1-a)\sum_i e_i \log \frac{e_i}{ap_i + (1-a)e_i} \\ &= a\sum_i p_i \log \frac{p_i}{ap_i + (1-a)e_i} + (1-a)\log \frac{1}{ap_y + (1-a)} \\ &= a\sum_{i \neq y} p_i \log \frac{p_i}{ap_i} + ap_y \log \frac{p_y}{ap_y + 1 - a} + (1-a)\log \frac{1}{ap_y + (1-a)} \\ &= a\sum_i p_i \log \frac{p_i}{ap_i} - ap_y \log \frac{p_y}{ap_y} + ap_y \log \frac{p_y}{ap_y + 1 - a} + (1-a)\log \frac{1}{ap_y + (1-a)} \\ &= ap_y \log ap_y - (ap_y + 1 - a)\log(ap_y + 1 - a) - a\log a \sum_i p_i \\ &= L'_{\mathrm{JS}}(\boldsymbol{s}, y) + a' R_{\mathrm{JS}}(\boldsymbol{s}) \end{aligned}$$



(a) CE, symmetric 0.2

(b) SCE, symmetric 0.2

(c) MAE, symmetric 0.2

(d) CE, human 0.4

(e) SCE, human 0.4

(f) MAE, human 0.4

(g) CE, asymmetric 0.4

(h) SCE, asymmetric 0.4

(i) MAE, asymmetric 0.4

Figure 6: Additional results to Fig. 1 with different label noise and loss functions: (a-c) symmetric label noise with $\eta = 0.2$; (d-f) human label noise with $\eta = 0.4$; (g-i) asymmetric label noise with $\eta = 0.4$. We include the median line of each distribution, test accuracies and the sampled epochs for reference.

is proposed as another combination of CE and MAE, where $0 < a < 1$ is the hyperparameter with $\lim_{a \to 0} L_{\text{JS}}(\boldsymbol{s}, y) = L_{\text{CE}}(\boldsymbol{s}, y)$ and $\lim_{a \to 1} L_{\text{JS}}(\boldsymbol{s}, y) = L_{\text{MAE}}(\boldsymbol{s}, y)$. $a' = -a \log a$ is the weight for regularizer

$$R_{\text{JS}}(\boldsymbol{s}) = \sum_i p_i$$

which can be regarded as a hyperparameter.

$$L'_{\text{JS}}(\boldsymbol{s}, y) = ap_y \log ap_y - (ap_y + 1 - a) \log(ap_y + 1 - a)$$

is the primary loss function of JS conforming to Eq. (3) with

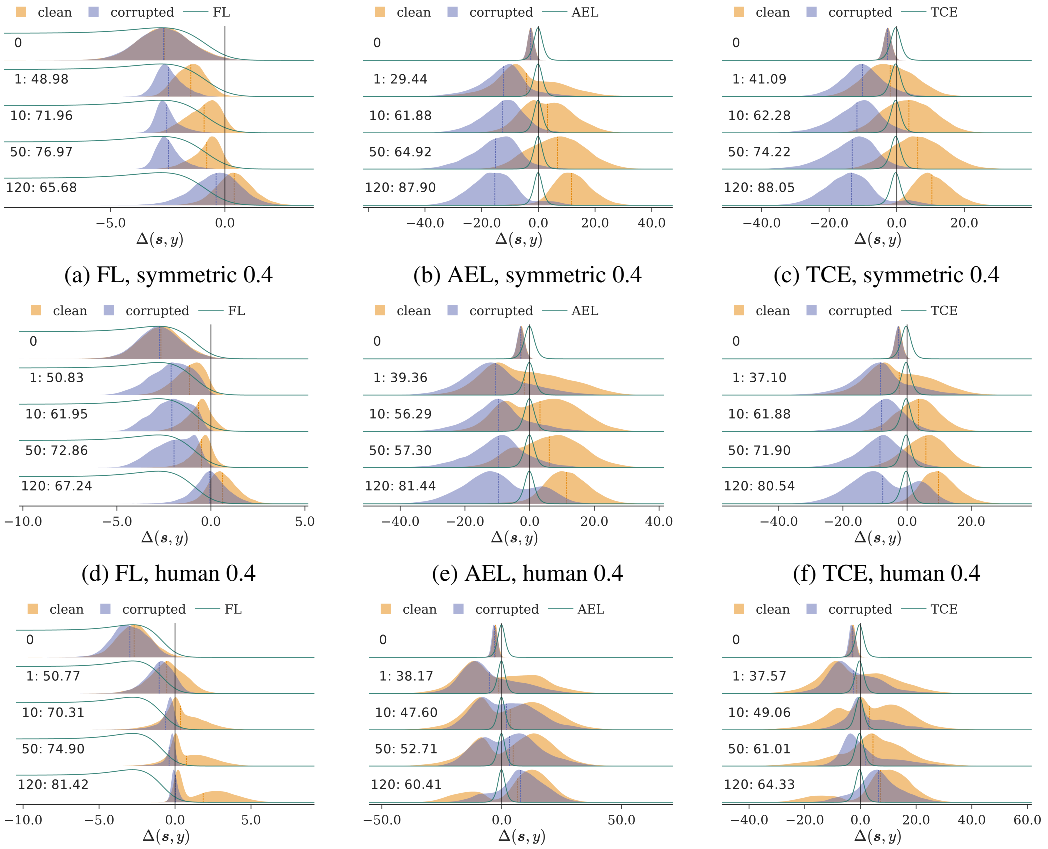
$$w_{\text{JS}}(\boldsymbol{s}, y) = ap_y(p_y - 1)(\log ap_y - \log(ap_y + 1 - q))$$

B Additional Results of the Training Dynamics

We complement §4 in the main text with detailed derivations, experiment settings and additional results. Hyperparameters of robust loss functions are summarized in Table 7.

Additional results of Fig. 1. In Fig. 6 and 7 we show extended results on the training dynamics with different label noise and loss functions. They follow the same trend as the results in Fig. 1 in the main text.

Additional results of Fig. 2. In Fig. 8 we show extended results on how different regularizers affect noise robust training. They follow the same trend as the results in Fig. 2 in the main text.



(a) FL, symmetric 0.4

(b) AEL, symmetric 0.4

(c) TCE, symmetric 0.4

(d) FL, human 0.4

(e) AEL, human 0.4

(f) TCE, human 0.4

(g) FL, asymmetric 0.4

(h) AEL, asymmetric 0.4

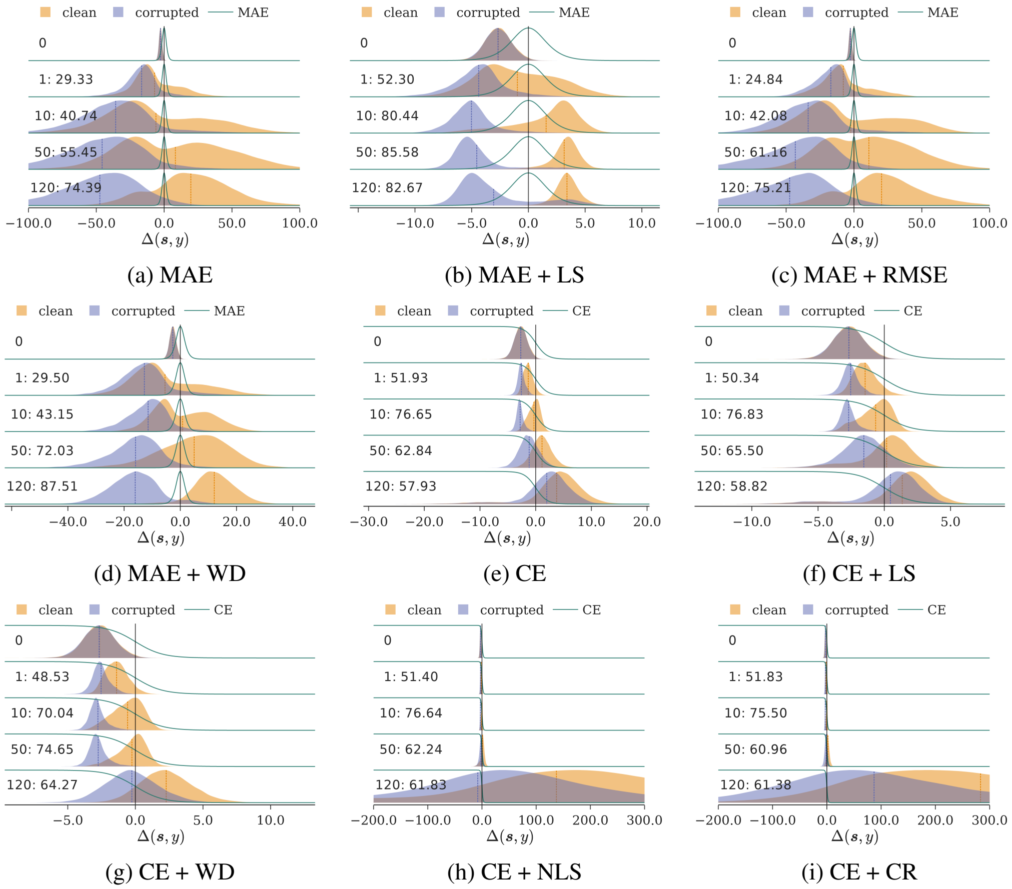
(i) TCE, asymmetric 0.4

Figure 7: Additional results to Fig. 1 with different label noise and loss functions: (a-c) symmetric label noise with $\eta = 0.2$; (d-f) human label noise with $\eta = 0.4$; (g-i) asymmetric label noise with $\eta = 0.4$. We include the median line of each distribution, test accuracies and the sampled epochs for reference.

Derivation of $\mathbb{E}_k[\Delta(\boldsymbol{s}, y)]$ in Eq. (5) Assume that class scores at *initialization* are i.i.d. normal variables $s_i \sim \mathcal{N}(\mu, \sigma)$,

$$\begin{aligned}
\mathbb{E}_k[\Delta(\boldsymbol{s}, y)] &= \mathbb{E}[s_y - \log \sum_{i \neq y} e^{s_i}] = \mu - \mathbb{E}[\log \sum_{i \neq y} e^{s_i}] \\
&\approx_1 \mu - \log \mathbb{E}[\sum_{i \neq y} e^{s_i}] + \frac{\mathbb{V}[\sum_{i \neq y} e^{s_i}]}{2\mathbb{E}[\sum_{i \neq y} e^{s_i}]^2} \\
&=_2 \mu - \log\{(k-1)\mathbb{E}[e^{s_y}]\} + \frac{(k-1)\mathbb{V}[e^{s_y}]}{2\{(k-1)\mathbb{E}[e^{s_y}]\}^2} \\
&=_3 \mu - \log[(k-1)e^{\mu + \sigma^2/2}] + \frac{(k-1)(e^{\sigma^2} - 1)e^{2\mu + \sigma^2}}{2[(k-1)e^{\mu + \sigma^2/2}]^2} \\
&= -\log(k-1) - \sigma^2/2 + \frac{e^{\sigma^2} - 1}{2(k-1)}
\end{aligned}$$

where $\approx_1$ follows the approximation with Taylor expansion $\mathbb{E}[\log X] \approx \log \mathbb{E}[X] - \mathbb{V}[X]/(2\mathbb{E}[X]^2)$ (Teh et al., 2006), $=_2$ utilizes properties of sum of log-normal variables (Cobb et al., 2012), and $=_3$ substitutes $\mathbb{E}[e^{s_y}]$ and $\mathbb{V}[e^{s_y}]$ with expressions for log-normal distributions.



(a) MAE

(b) MAE + LS

(c) MAE + RMSE

(d) MAE + WD

(e) CE

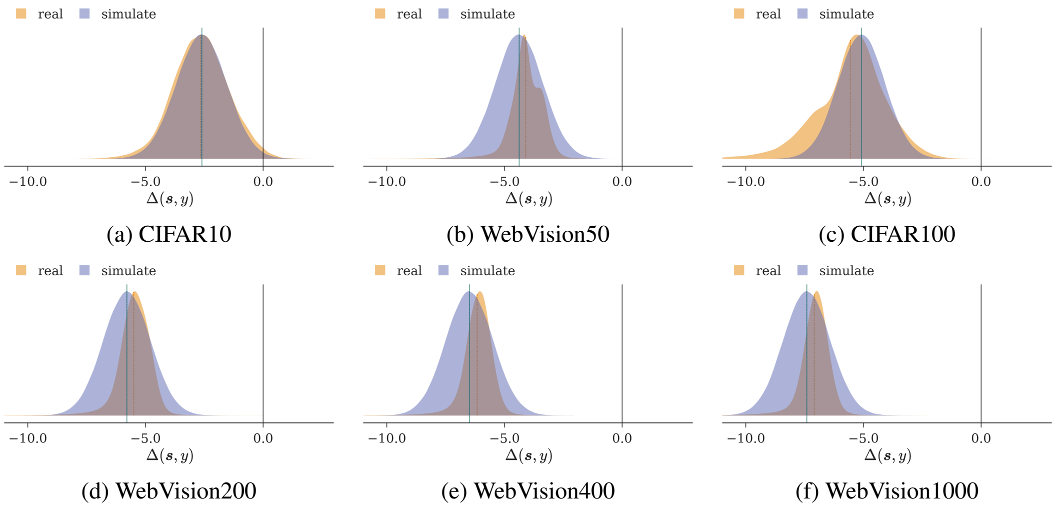
(f) CE + LS

(g) CE + WD

(h) CE + NLS

(i) CE + CR

Figure 8: Training dynamics with various loss functions and regularizers on CIFAR10 with symmetric noise $\eta = 0.4$. We include the median (dashed) of each distribution and $-\log k$ (solid) lines and the test accuracy of each epoch.



(a) CIFAR10

(b) WebVision50

(c) CIFAR100

(d) WebVision200

(e) WebVision400

(f) WebVision1000

Figure 9: Comparisons between shapes of the simulated and real $\Delta(\boldsymbol{s}, y)$ distributions at initialization. Simulations are based on the assumption that class scores follow normal distribution $s_i \sim \mathcal{N}(0, 1)$ at initialization. Real distributions are extracted with randomly initialized models. We also mark the estimated $\mathbb{E}_k[\Delta(\boldsymbol{s}, y)]$ with vertical green lines.

	Clean	Symmetric			
Loss	$\eta = 0$	$\eta = 0.2$	$\eta = 0.4$	$\eta = 0.6$	$\eta = 0.8$
CE‡	71.33 ± 0.43	56.51 ± 0.39	39.92 ± 0.10	21.39 ± 1.17	7.59 ± 0.20
GCE‡	63.09 ± 1.39	61.57 ± 1.06	56.11 ± 1.35	45.28 ± 0.61	17.42 ± 0.06
NCE+TCE‡	69.03 ± 0.37	65.66 ± 0.46	59.47 ± 0.36	48.02 ± 0.58	24.72 ± 0.60
CE	**77.44 ± 0.13**	64.84 ± 0.19	49.96 ± 0.02	30.41 ± 0.60	11.23 ± 0.45
GCE	73.88 ± 0.25	70.55 ± 0.73	64.67 ± 0.49	52.57 ± 0.47	23.90 ± 2.69
NCE+AGCE	76.37 ± 0.25	**72.19 ± 0.26**	64.55 ± 0.46	52.79 ± 0.22	**26.19 ± 1.14**
TCE	58.04 ± 1.15	52.13 ± 1.92	45.91 ± 1.25	33.64 ± 0.22	20.47 ± 1.45
TCE shift, $\tau = 4.2$	77.14 ± 0.11	71.28 ± 0.44	60.17 ± 0.47	40.91 ± 0.33	18.16 ± 0.37
TCE scale, $\tau = 4.2$	75.79 ± 0.17	71.11 ± 0.39	62.88 ± 0.59	46.52 ± 0.39	20.78 ± 1.53
MAE	7.46 ± 1.92	4.65 ± 1.55	3.21 ± 0.57	1.61 ± 0.53	1.54 ± 0.47
IMAE, $T = 10$	5.26 ± 3.19	11.81 ± 3.10	45.01 ± 1.10	28.94 ± 1.92	4.51 ± 0.69
MAE shift	76.65 ± 0.30	71.36 ± 0.70	61.29 ± 0.49	42.68 ± 1.14	19.30 ± 1.00
MAE scale	73.54 ± 0.32	69.96 ± 0.55	**64.92 ± 0.20**	**54.98 ± 0.17**	23.00 ± 2.44

Table 8: Extended result of Table 5: shifting or scaling $w(\boldsymbol{s}, y)$ mitigates underfitting on CIFAR100 with symmetric label noise. We report test accuracies with 3 different runs. Both τ of the shifted/scaled MAE/TCE and T of IMAE are tuned with symmetric noise $\eta = 0.4$. Previous best results from Zhou et al. (2021) are included as context (denoted with ‡). Previous best results from Zhou et al. (2021) are included as context (denoted with ‡).

Simulated $\Delta(\boldsymbol{s}, y)$ approximate real settings. We compare the simulated $\Delta(\boldsymbol{s}, y)$ distributions based on $s_i \sim \mathcal{N}(0, 1)$ to distributions of real datasets at initialization in Fig. 9. The expectations of simulated $\Delta(\boldsymbol{s}, y)$ are similar to real settings, which supports the analysis in §4.3. The estimated $\mathbb{E}_k[\Delta(\boldsymbol{s}, y)]$ fits the median of simulation well. Finally, increasing the number of classes k consistently decreases $\Delta(\boldsymbol{s}, y)$ at initialization.

Additional results on the effectiveness of shifting and scaling $w(\boldsymbol{s}, y)$. We show additional results on how shifting and scaling $w(\boldsymbol{s}, y)$ affect noise robust training in Tables 8 and 9. Hyperparameters for results of the WebVision dataset are shown in

Loss	Clean	Asymmetric			
	$\eta = 0$	$\eta = 0.1$	$\eta = 0.2$	$\eta = 0.3$	$\eta = 0.4$
CE[‡]	71.33 ± 0.43	64.85 ± 0.37	58.11 ± 0.32	50.68 ± 0.55	40.17 ± 1.31
GCE[‡]	63.09 ± 1.39	63.01 ± 1.01	59.35 ± 1.10	53.83 ± 0.64	40.91 ± 0.57
NCE+AGCE[‡]	69.03 ± 0.37	67.22 ± 0.12	63.69 ± 0.19	55.93 ± 0.38	43.76 ± 0.70
CE	**77.44 ± 0.13**	72.08 ± 0.19	64.75 ± 0.49	55.62 ± 0.12	45.73 ± 0.49
GCE	73.88 ± 0.25	72.29 ± 0.16	67.96 ± 0.24	57.83 ± 1.02	45.14 ± 0.13
NCE+AGCE	76.37 ± 0.25	**73.73 ± 0.21**	64.58 ± 0.37	52.23 ± 1.18	40.93 ± 1.22
TCE	58.04 ± 1.15	53.51 ± 1.18	45.14 ± 1.25	34.59 ± 2.47	28.35 ± 0.74
TCE shift, $\tau = 4.2$	77.14 ± 0.11	71.82 ± 0.60	63.41 ± 0.06	54.91 ± 0.47	44.56 ± 0.71
TCE scale, $\tau = 4.2$	75.79 ± 0.17	72.64 ± 0.48	63.58 ± 0.32	53.73 ± 0.58	43.57 ± 1.19
MAE	7.46 ± 1.92	4.65 ± 1.55	3.21 ± 0.57	1.61 ± 0.53	1.54 ± 0.47
IMAE, $T = 10$	5.26 ± 3.19	10.05 ± 2.16	14.90 ± 1.41	32.61 ± 0.24	48.09 ± 0.68
MAE shift, $\tau = 3.4$	76.65 ± 0.30	71.82 ± 0.44	63.37 ± 0.28	53.81 ± 0.57	44.06 ± 1.23
MAE scale, $\tau = 3.4$	73.54 ± 0.32	71.85 ± 0.31	**69.43 ± 0.39**	**63.14 ± 0.37**	**48.88 ± 0.79**

Table 9: Extended result of Table 5: shifting or scaling $w(\boldsymbol{s}, y)$ mitigates underfitting on CIFAR100 with asymmetric label noise. We report test accuracies with 3 different runs. Both τ of the shifted/scaled MAE/TCE and T of IMAE are tuned with symmetric noise $\eta = 0.4$. Previous best results from Zhou et al. (2021) are included as context (denoted with ‡). Previous best results from Zhou et al. (2021) are included as context (denoted with ‡).

Settings	CE	MAE	IMAE	MAE shift	MAE scale	TCE	TCE shift	TCE scale
$k = 50$	/	/	$T = ?$	$\tau = 2.0$	$\tau = 2.0$	$q = ?$	$q = ?, \tau = ?$	$q = ?, \tau = ?$
$k = 200$	/	/	$T = ?$	$\tau = 1.8$	$\tau = 1.8$	$q = ?$	$q = ?, \tau = ?$	$q = ?, \tau = ?$
$k = 400$	/	/	$T = ?$	$\tau = 1.6$	$\tau = 1.6$	$q = ?$	$q = ?, \tau = ?$	$q = ?, \tau = ?$

Table 10: Hyperparameters roughly tuned with each settings for results of WebVision in Table 6. For TCE shift/scale, we only tune τ and leave q intact to demonstrate the effect of the shift/scale fixes.