To Smooth or Not? When Label Smoothing Meets Noisy Labels

Jiaheng Wei [1] **Hangyu Liu** [2] **Tongliang Liu** [3] **Gang Niu** [4] **Masashi Sugiyama** [4 5] **Yang Liu** [1]

Abstract

Label smoothing (LS) is an arising learning paradigm that uses the positively weighted average of both the hard training labels and uniformly distributed soft labels. It was shown that LS serves as a regularizer for training data with hard labels and therefore improves the generalization of the model. Later it was reported LS even helps with improving robustness when learning with noisy labels. However, we observed that the advantage of LS vanishes when we operate in a high label noise regime. Intuitively speaking, this is due to the increased entropy of $\mathbb{P}(\text{noisy label}|X)$ when the noise rate is high, in which case, further applying LS tends to "over-smooth" the estimated posterior. We proceeded to discover that several learning-with-noisy-labels solutions in the literature instead relate more closely to *negative/not label smoothing* (NLS), which acts counter to LS and defines as using a negative weight to combine the hard and soft labels! We provide understandings for the properties of LS and NLS when learning with noisy labels. Among other established properties, we theoretically show NLS is considered more beneficial when the label noise rates are high. We provide extensive experimental results on multiple benchmarks to support our findings too. Code is publicly available at https://github.com/UCSC-REAL/negative-label-smoothing.

1. Introduction

Label smoothing (LS) (Szegedy et al., 2016) is an arising learning paradigm that uses positively weighted average of both the hard training labels and the uniformly distributed soft label:

$$\mathbf{y}^{\text{LS},r} = (1-r)\cdot\mathbf{y} + \frac{r}{K}\cdot\mathbf{1}, \tag{1}$$

where we denote the one-hot vector form of hard label and an all one vector as $\mathbf{y}$, $\mathbf{1}$ respectively. K is the number of label classes, and r is the smooth rate in the range of $[0, 1]$. It was shown that LS serves as a regularizer for the hard training data and therefore improves generalization of the model. The regularizer role of LS prevents the model from fitting overly on the target class. Empirical studies have demonstrated the effectiveness of LS in improving the model performance across various benchmarks (Pereyra et al., 2017) (such as image classification (Szegedy et al., 2016), machine translation (Vaswani et al., 2017), language modelling (Chorowski & Jaitly, 2017)), and model calibration (Müller et al., 2019). Later it was reported LS even helps with improving robustness when learning with noisy labels (Lukasik et al., 2020). However, we observed that the advantage of LS vanishes when we operate in a high label noise regime: in Figure 1, we present a set of experiments on some UCI datasets (Dua & Graff, 2017) with synthetic noisy labels. We highlight the best two smooth rates when the classifier is trained under each label noise rate. Since UCI datasets are of small scales, it is possible to have tied smooth rates when evaluating the classifier on the separate clean test data. Indeed, non-negative smooth rates (circles colored in red) outperform negative ones when the label noise rates are low. Nonetheless, with the increasing of noise rates, negative smooth rates $r < 0$ (Eqn. (1), diamonds colored in green) appear to be more competitive when learning with noisy labels. Intuitively speaking, this is due to the increased entropy of $\mathbb{P}(\text{noisy label}|X)$ when the noise rate is high, in which case, further applying LS tends to "oversmooth" the estimated posterior. Motivated by this observation, we aim to provide a more thorough understanding of whether should we adopt label smoothing or not when learning with noisy labels, specifically, how to make a choice between LS and negative/not label smoothing (NLS)?

With the presence of label noise, we theoretically demonstrate that there exists a phase transition when finding the optimal label smoothing rate for $r \in (-\infty, 1]$. Particularly, when the label noise rate is low, LS is able to uncover the optimal model while NLS is considered more beneficial in

[1]University of California, Santa Cruz [2]Brown University [3]TML Lab, Sydney AI Centre, The University of Sydney [4]RIKEN AIP [5]University of Tokyo. Correspondence to: Yang Liu <yangliu@ucsc.edu>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022.

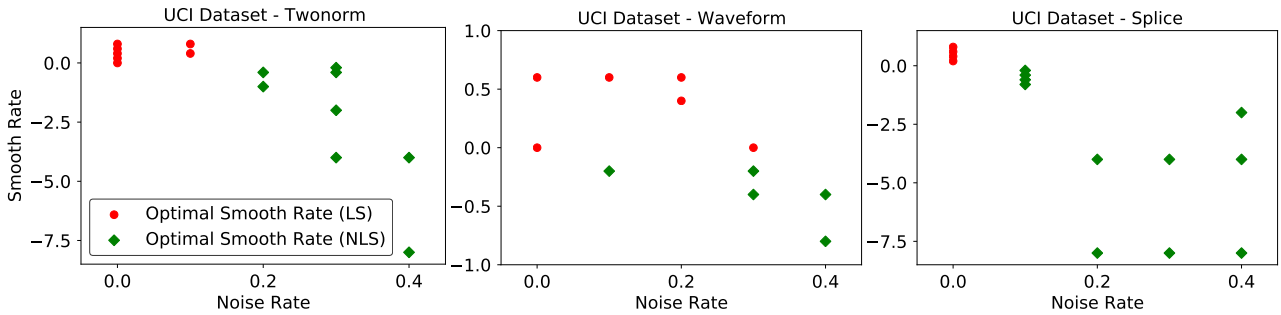

Figure 1. Optimal smooth rates on UCI datasets with different label noise rates (possible to have tied smooth rates).

a high label noise regime. Discovering that NLS differs substantially from LS in their achieved model confidence, we then proceed to explain such a transition. We also bridge the gap between NLS and several learning-with-noisy-labels solutions in the literature, including Loss Correction (Patrini et al., 2017), NLNL (Kim et al., 2019) and Peer Loss (Liu & Guo, 2020), to further validate our results.

We provide extensive experimental evidences to support our findings. For instance, on multiple benchmark datasets, we present the clear transition of the optimal smoothing rate going from positive to negative when we keep increasing noise rates. In particular, we show a negative smoothing rate elicits higher model confidence on correct predictions and lower confidence on wrong predictions compared with the behavior of a positive one on CIFAR-10 test data.

Our contributions summarize as follows:

- We provide understandings for the decision between LS and NLS, when learning with noisy labels.
- We demonstrate learning with a negative smooth rate can be more robust to label noise compared with a positive rate when label noise rates are high. And this is best explained by the fact that NLS improves the confidence of model prediction. (Section 3 and 4)
- We show that several robust loss functions in the label-noise literature correspond to learning with NLS, under certain noise rate models. (Section 5)
- Extensive empirical results validate our main theoretical conclusions. In Appendix, we discuss practical considerations to mitigate the impact of label noise, and empirically show how LS and NLS result in trade-offs in model confidence, bias and variance of the generalization error.

We defer all proofs to Appendix F. Our work primarily contributes to the literature of learning with noisy labels (Scott et al., 2013; Natarajan et al., 2013; Liu & Tao, 2015; Patrini et al., 2017; Liu & Guo, 2020). Our core results are contingent on recent works of understanding the effect of label smoothing when training deep neural network models, i.e., label smoothing improves model calibration (Müller et al., 2019), more complicated forms of label smoothing (Li et al., 2020; Yuan et al., 2020), and in particular when label noise presents (Lukasik et al., 2020; Liu, 2021). Due to the space limit, we defer a more detailed discussion of related works to Appendix A.

2. Preliminaries

2.1. Learning with noisy labels

For a K-class classification task, we denote by $X \in \mathcal{X}$ a high-dimensional feature and $Y \in \mathcal{Y} := \{1, 2, ..., K\}$ the corresponding label. Suppose $(X, Y) \in \mathcal{X} \times \mathcal{Y}$ are drawn from a joint distribution $\mathcal{D}$. The noisy label literature (Natarajan et al., 2013; Liu & Tao, 2015; Patrini et al., 2017) considers the setting where we only have access to samples with noisy labels from $(X, \widetilde{Y})$. Suppose random variables $(X, \widetilde{Y}) \in \mathcal{X} \times \widetilde{\mathcal{Y}}$ are drawn from a noisy joint distribution $\widetilde{\mathcal{D}}$. Statistically, the random variable of noisy labels $\widetilde{Y}$ can be characterized by a noise transition matrix T, where each element $T_{i,j}$ represents the probability of flipping the clean label $Y = i$ to the noisy label $\widetilde{Y} = j$, i.e., $T_{ij} = \mathbb{P}(\widetilde{Y} = j | Y = i)$. In this paper, we concentrate on the widely adopted class-dependent label noise (Natarajan et al., 2013; Liu & Tao, 2015; Patrini et al., 2017), which assumes that the label noise is conditionally independent of features X, i.e.,

$$\mathbb{P}(\widetilde{Y} = j | Y = i) = \mathbb{P}(\widetilde{Y} = j | X, Y = i), \forall i, j \in [K].$$

For the binary classification setting, define $e_0 := \mathbb{P}(\widetilde{Y} = 1 | Y = 0)$, $e_1 := \mathbb{P}(\widetilde{Y} = 0 | Y = 1)$. Without loss of generality, we assume $e_1 - e_0 = e_\Delta \geq 0$. The binary noise transition matrix in the noisy label setting then becomes:

$$T = \begin{pmatrix} 1 - e_0 & e_0 \\ e_1 & 1 - e_1 \end{pmatrix}.$$

2.2. Learning with smoothed labels

Let $\mathbf{y}_i$ be the one-hot encoded vector form of y_i which generates according to Y. The random variable of smoothed

label $Y^{\text{LS},r}$ with smooth rate $r \in [0,1]$ generates $\mathbf{y}_i^{\text{LS},r}$ as (Szegedy et al., 2016):

$$\mathbf{y}_i^{\text{LS},r} = (1-r)\cdot \mathbf{y}_i + \frac{r}{K}\cdot \mathbf{1}.$$

For example, when $r = 0.3$, the smoothed label of $\mathbf{y}_i = [1,0,0]^\top$ becomes $\mathbf{y}_i^{\text{LS},r=0.3} = [0.8, 0.1, 0.1]^\top$.

To enable ease of presentations (instead of highlighting a crucial concept), we unify LS (Szegedy et al., 2016; Lukasik et al., 2020) and NLS into the generalized label smoothing (GLS), i.e., $r \in (-\infty, 1]$:

$$\mathbf{y}_i^{\text{GLS},r} := (1-r)\cdot \mathbf{y}_i + \frac{r}{K}\cdot \mathbf{1}, \tag{2}$$

where $\mathbf{y}_i^{\text{GLS},r}$ is given by the random variable of generalized smooth label $Y^{\text{GLS},r}$. We name the scenario $r < 0$ as negative/not label smoothing (NLS). A negative r indicates that the smoothed label might be negatively related to the corresponding feature and should not be (positively) smoothed. For example, when $r = -0.3$, the smoothed label of $\mathbf{y}_i = [1,0,0]^\top$ becomes $\mathbf{y}_i^{\text{GLS},r=-0.3} = [1.2, -0.1, -0.1]^\top$. We observe that the entries in $\mathbf{y}_i^{\text{GLS},r}$ still add up to 1: $1 - r + \frac{r}{K}\cdot K = 1$. Nonetheless we want to point out $\mathbf{y}_i^{\text{GLS},r}$ is no longer a valid probability measure since for entries $y \neq y_i$, the corresponding weight will be negative ($\frac{r}{K}$) when $r < 0$. This points us to the definition of an extended label distribution:

Definition 2.1 (Extended label distribution)**.** We call $\mathbf{y}$ an extended label distribution if $\mathbf{1}^\top \mathbf{y} = 1$, but $\mathbf{y}$ is not necessarily entry-wise non-negative.

What negative labels really mean Negative label smoothing is indeed one of such extended label distribution. We proceed the illustration using the previous three-class classification example: a one-hot label $[1,0,0]^\top$ (three elements stand for class dog (0), cat (1), deer (2), respectively) means this sample x is categorized as a dog and is irrelevant to class cat and deer. LS $[0.8, 0.1, 0.1]^\top$ indicates that the representation x might encode uncertainty and is slightly related to cat/deer (positive correlation between cat/deer and dog given x). NLS $[1.2, -0.1, -0.1]^\top$ not only implies high confidence in label dog, but it is even more so that predicting cat (1) or deer (2) should be penalized by 0.1, i.e., given any loss ℓ that is linear in y (e.g., CE loss), this x receives the loss $1.2\cdot\ell(\mathbf{f}(x),0) - 0.1\cdot\ell(\mathbf{f}(x),1) - 0.1\cdot\ell(\mathbf{f}(x),2)$. Such a penalization mechanism is not uncommon and it appeared in the design of backward loss correction (Natarajan et al., 2013) $\sum_{i\in\{0,1,2\}} T_{0,i}^{-1}\cdot\ell(\mathbf{f}(x),i)$ with $T_{0,1}^{-1}, T_{0,2}^{-1} \leq 0$ (T^{-1} is the inverse matrix of T), peer loss (Liu & Guo, 2020) $\ell(\mathbf{f}(x),0) - \ell(\mathbf{f}(x), y_{\text{rand}})$ where $y_{\text{rand}} = i$ with probability $\mathbb{P}(\widetilde{Y} = i)$ for $i \in \{0,1,2\}$, and complementary loss (more details in Section 5).

We will present the surprising power of negative labels in handling label noise, though a bit counter-intuitive at first sight. To clarify, although we may adopt the negative label for calculating the loss, the model prediction is processed by the soft-max function, so the prediction still lies on the K-simplex. Besides, we do not assume a strict lower bound for r. If $r \to -\infty$, normalizing $\mathbf{y}_i^{\text{GLS},r}$ by $1-r$ returns $\mathbf{y}_i^{\text{GLS},r} = \mathbf{y}_i - \frac{\mathbf{1}}{K}$. We will show when imposing a negative smoothing parameter will be considered beneficial as compared to a positive one. In the main paper, we mainly focus on the binary classification task where $y_i \in \{0,1\}$ and $K = 2$, although we do include the discussion of multi-class extensions in Section 3.4.

2.3. Model confidence

Denote a deep neural network as f, $\mathbf{f}(x_i)$ is the model prediction of $x_i \in X$ with element $\mathbf{f}(x_i)_{y_i} := \mathbb{P}(Y = y_i | X = x_i, f)$, the binary cross-entropy loss is then defined as $\ell_{\text{CE}}(\mathbf{f}(x_i), y_i) := -\log(\mathbf{f}(x_i)_{y_i})$. Throughout this paper, we shorthand ℓ_{CE} as ℓ for a clean presentation. We define a key quantity, model confidence, that plays an important role in later sections.

Definition 2.2 (Confidence of model f for sample (x,y))**.** Given a model f, a sample x with its target label $y \in \{0,1\}$, the model confidence of f w.r.t. sample x is defined as $\text{MC}(f;x,y) = \mathbf{f}(x)_y - \mathbf{f}(x)_{1-y}$.

$\text{MC}(f;x,y)$ in Definition 2.2 characterizes the difference of the predicted probability between the target class and the other class. $\text{MC}(f;x,y) = 0$ simply means f has no confidence on its predictions since the model can not identify the target class of x. $\text{MC}(f;x,y)$ is negative when f gives a wrong prediction and is not confident to predict the label of x as the target label y. To dig into how GLS influences the model confidence on correct and wrong predictions in following sections, we separate the distribution $\mathcal{D}$ into:

$$\mathcal{D}_f^+ := \{(X,Y) \sim \mathcal{D} : \text{MC}(f;X,Y) > 0\},$$
$$\mathcal{D}_f^- := \{(X,Y) \sim \mathcal{D} : \text{MC}(f;X,Y) \leq 0\}.$$

Similarly, we introduce the confidence of model prediction under the metric of ℓ-loss as:

Definition 2.3 (ℓ-based confidence of model f for sample (x,y))**.** Given a model f, a sample x with its target label $y \in \{0,1\}$, the ℓ-based model confidence of f w.r.t. sample x is defined as $\text{MC}_\ell(f;x,y) := -(\ell(\mathbf{f}(x),y) - \ell(\mathbf{f}(x),1-y))$.

3. To Smooth or Not? In the View of Risk Minimization

In this section, we aim to characterize the optimal candidates of r in the unified setting to distinguish the preferences for LS and NLS, when the label noise presents.

Let $\tilde{\mathbf{y}}$ be the vector form of noisy label $\tilde{y}$ obtained from $\widetilde{Y}$. For $r \leq 1$, we define the r-smoothed label of $\tilde{y}$ as $\tilde{\mathbf{y}}^{\text{GLS},r}$, where $\tilde{\mathbf{y}}^{\text{GLS},r} := (1-r)\cdot\tilde{\mathbf{y}} + \frac{r}{K}\cdot\mathbf{1}$ and is generated by the random variable $\widetilde{Y}^{\text{GLS},r}$. Risk minimization w.r.t. smoothed noisy label distribution $\widetilde{Y}^{\text{GLS},r}$ is then defined as:

$$\min_{f\in\mathcal{F}} \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\ell(\mathbf{f}(X),\widetilde{Y}^{\text{GLS},r})\Big], \tag{3}$$

where in above $\mathcal{F}$ is the hypothesis space we consider. In Figure 1, we have shown that given the unseen test data, learning with non-negative smooth rates may not always return the best outcome. Based on this observation, we delve into details to show when NLS is more favorable than LS and Vanilla Loss (VL, $r=0$). We start with stating Assumption 3.1:

Assumption 3.1. We assume learning with clean data distribution $\mathcal{D}$ with smooth rate $r^* \leq 1$ in GLS makes the corresponding classifier $f^*_{\mathcal{D}}$ return the best performance on the unseen clean test data distribution $\mathcal{D}_{test}$, where $f^*_{\mathcal{D}}$ is given by: $f^*_{\mathcal{D}} \leftarrow \arg\min_{f\in\mathcal{F}} \mathbb{E}_{(X,Y)\sim\mathcal{D}}[\ell(\mathbf{f}(X), Y^{\text{GLS},r^*})]$.

Assumption 3.1 simply offers us a view to initiate our analysis for the noisy label setting. To clarify, the expected risk of random variables could be approximated/replaced by the empirical one over a finite number of samples: i.e., when $\mathcal{D} = \{x_i, y_i\}_{i=1}^N, \widetilde{\mathcal{D}} = \{x_i, \tilde{y}_i\}_{i=1}^N$, Eqn. (3) becomes: $\min_{f\in\mathcal{F}} \frac{1}{N}\sum_{i\in[N]} \ell(\mathbf{f}(x_i), \tilde{y}_i^{\text{GLS},r})$. In this case, $\mathcal{D}$ represents the empirical distribution for the finite dataset, and $\mathcal{D}_{\text{test}}$ can be thought of as the expected risk with infinite samples. With this being said, our analysis does require taking the expectation over the noisy labels (over $\widetilde{Y}|X,Y$). Besides, we don't rule out the possibility that other methods outperform LS, VL or NLS with optimal smooth rate r^*. At the end of this section and Appendix D, we will empirically test what r^* usually is on various benchmarks. We denote the r^* smoothed label distribution as Y^*: $Y^* = Y^{\text{GLS},r^*}$. With the introduction of r^* and $f^*_{\mathcal{D}}$, our goal is then to recover the classifier $f^*_{\mathcal{D}}$ using the noisy training labels. We define λ_1, λ_2 and offer Theorem 3.2.

$$\lambda_1 := \Big[(e_0 - \frac{r^*}{2}) + (1-2e_0)\cdot\frac{r}{2}\Big], \quad \lambda_2 := e_\Delta\cdot(1-r).$$

Theorem 3.2. *The risk minimization w.r.t. $\widetilde{Y}^{GLS,r}$ in the noisy setting (Eqn. (3)) is equivalent to the risk w.r.t Y^* defined on the clean data, with two additional bias terms:*

$$\min_{f\in\mathcal{F}} \underbrace{\mathbb{E}_{(X,Y)\sim\mathcal{D}}\Big[\ell(\mathbf{f}(X),Y^*)\Big]}_{\textit{True Risk}} + \underbrace{\lambda_1\cdot\mathbb{E}_{(X,Y)\sim\mathcal{D}}\Big[\ell(\mathbf{f}(X),1-Y)-\ell(\mathbf{f}(X),Y)\Big]}_{\textit{M-Inc1}} + \underbrace{\lambda_2\cdot\mathbb{E}_{X,Y=1}\Big[\ell(\mathbf{f}(X),0)-\ell(\mathbf{f}(X),1)\Big]}_{\textit{M-Inc2}}. \tag{4}$$

Remember that $\ell = \ell_{\text{ce}}$, we have: $\text{MC}_\ell(f;X,Y) = \log\left(\mathbf{f}(X)_Y/\left(1-\mathbf{f}(X)_Y\right)\right)$, $\text{MC}(f;X,Y) = 2\mathbf{f}(X)_Y - 1$. Both $\log(\frac{x}{1-x})$ and $2x-1$ are monotonically increasing for $x\in(0,1)$, model f with a high $\text{MC}_\ell(f;X,Y)$ has high $\text{MC}(f;X,Y)$. The two extra bias terms explicitly affect the model confidence. Now we proceed to answer "what r is preferred in the noisy setting".

3.1. Symmetric noise rates with $e_\Delta = 0$

Symmetric noise rates $e := e_0 = e_1$ indicates the probability of flipping to the other class is equal for both classes. In this case, $\lambda_2 = 0$, Term M-Inc2 is cancelled and Eqn. (4) reduces to

$$\min_{f\in\mathcal{F}} \underbrace{\mathbb{E}_{(X,Y)\sim\mathcal{D}}\Big[\ell(\mathbf{f}(X),Y^*)\Big]}_{\text{True Risk}} + \underbrace{\lambda_1\cdot\mathbb{E}_{(X,Y)\sim\mathcal{D}}\Big[\ell(\mathbf{f}(X),1-Y)-\ell(\mathbf{f}(X),Y)\Big]}_{\text{M-Inc1}}. \tag{5}$$

Noisy labels impair model confidence on Vanilla Loss In the unified framework, define the optimal r that will cancel the impact of Term M-Inc1 as:

$$\text{when } r_{\text{opt}} := \frac{r^*-2e}{1-2e}, \quad \text{M-Inc1} = 0. \tag{6}$$

The threshold r_{opt} in Eqn. 6 implies:

Theorem 3.3. *With Assumption 3.1, learning with smooth rate $r = r_{opt}$ under $(X,\widetilde{Y}) \sim \widetilde{\mathcal{D}}$ yields $f^*_{\mathcal{D}}$:*

- *When noise rate $e < r^*/2$, $r = r_{opt} > 0$ (LS);*
- *When noise rate $e = r^*/2$, $r = 0$ (VL);*
- *When noise rate $e > r^*/2$, $r = r_{opt} < 0$ (NLS).*

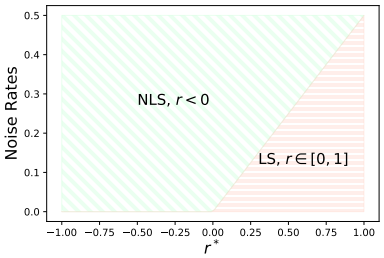

Figure 2. Decision between NLS, LS given e, r^*.

In Theorem 3.3, adopting NLS when noise rate $e < \frac{r^*}{2}$ induces $\lambda_1 < 0$, Term M-Inc1 makes f overly-confident on its predictions compared with Y^*. In Figure 2, with the decreasing of r^*, LS is less tolerant of labels with high noise. Similarly, if $e \geq \frac{r^*}{2}$, with the decreasing of r^*, NLS is more robust in the high noise regime while LS makes the model f become less-confident on its predictions. Clearly, NLS outperforms LS especially when noise rates are large and r^* is small.

3.2. Asymmetric noise rates with $e_\Delta \neq 0$

In this case, adopting $r = \frac{r^*-2e_0}{1-2e_0}$ removes the Term M-Inc1. However, when $r < 1$, Term M-Inc2 is not negligible due to assymetric noise transition matrix. As a result, Term M-Inc2 becomes:

$$e_\Delta \cdot \frac{1-r^*}{1-2e_0} \cdot \mathbb{E}_{X,Y=1}\Big[\ell(\mathbf{f}(X),0) - \ell(\mathbf{f}(X),1)\Big],$$

with $e_\Delta \cdot \frac{1-r^*}{1-2e_0} \geq 0$. Term M-Inc2 in the minimization increases the model confidence on $(X, Y = 0) \sim \mathcal{D}_f^+$. The model will then become overly-confident with the class that has a low noise rate e_0. Meanwhile, Term M-Inc2 decreases the model confidence on $(X, Y = 1) \sim \mathcal{D}_f^+$ (less-confident to the class with a high noise rate e_1).

3.3. Analysis of empirical risks

The popularity of LS is largely due to its effectiveness in practice, i.e., through the optimization of the smoothed empirical risk. Given the smooth rate r, potentially we could adopt the Rademacher bound on the maximal deviation between the expected risk $R^r_{\text{exp}}(f)$ (objective in Eqn. (3)) and the empirical risk $R^r_{\text{emp}}(f) := \frac{1}{N}\sum_{i\in[N]} \ell(\mathbf{f}(x_i), \tilde{y}_i^{\text{GLS},r})$ when learning with noisy labels, formally, we have:

Theorem 3.4. *With probability at least $1-\delta$, we have:*

$$\max_{f\in\mathcal{F}} |R^r_{emp}(f) - R^r_{exp}(f)|$$

$$\leq (2+|r|-r)\cdot L \cdot \mathfrak{R}(\mathcal{F}) + (1-r)\cdot(\overline{\ell}-\underline{\ell})\cdot\sqrt{\frac{\log(1/\delta)}{2N}},$$

where $\overline{\ell}, \underline{\ell}$ denote the upper/lower bound of ℓ, $\mathfrak{R}$ is the Rademacher complexity.

Theorem 3.4 bridges the gap between the expected risk $R^r_{\text{exp}}(f)$ and the empirical risk $R^r_{\text{emp}}(f)$ by offering an upper bound. Intuitively, with a large sample size N and a low Rademacher complexity of the hypothesis space $\mathfrak{R}(\mathcal{F})$, $R^r_{\text{emp}}(f)$ is supposed to well-approximate $R^r_{\text{exp}}(f)$. When learning with finite samples, LS is popular by referring to its impacts on reducing the model confidence (or avoids over-fitting). NLS indeed may force model become confident on the prediction, including wrong ones. What we observe in practice is that neural nets firstly memorize on easy/clean samples (Liu et al., 2020), warm-up with CE and then switch to NLS significantly improves the model performance. Since in the latter stage, the model is encouraged to be more confident on learned patterns (clean samples) and less likely to over-confident on samples with wrong labels (large-loss samples). When noise rate is high, the noisy training data is already over-smoothing the training process. Think of the noisy label flipping corresponding to a certain smooth rate, and a case where a certain representation x, with its similar patterns, are sampled multiple times - then their associated noisy labels formed a smoothed distribution. In this case, applying the NLS corrects the over-smoothness.

3.4. Multi-class extension

As an extension to the binary classification task, we next show how Theorem 3.3 could be generalized to the multi-class setting under two broad families of noise transition model. We assume Assumption 3.1 holds in the multi-class setting. And for $Y, \widetilde{Y} \in [K]$, we extend the definition of model confidence to multi-class classification tasks as:

Definition 3.5 (Model confidence of sample (x, y) (K-class classification))**.** Given a model f, a sample x with its target label $y \in [K]$, the model confidence score of f w.r.t. sample x is defined as $\text{MC}(f; x, y) = \mathbf{f}(x)_y - \frac{1}{K-1}\sum_{i\neq y}\mathbf{f}(x)_i$.

Sparse noise transition matrix Sparse noise model (Wei & Liu, 2021) assumes K is an even number. For $c \in [\frac{K}{2}]$, $i_c < j_c$, sparse noise model specifies $\frac{K}{2}$ disjoint pairs of classes (i_c, j_c) to simulate the scenario where particular pairs of classes are ambiguity and misleading for human annotators. The off-diagonal element of T reads $T_{i_c,j_c} = e_0$, $T_{j_c,i_c} = e_1$. Suppose $e_0 + e_1 < 1$, the diagonal entries become $T_{i_c,i_c} = 1 - e_0$, $T_{j_c,j_c} = 1 - e_1$. Clearly, our conclusions in Theorem 3.3 extends directly to the sparse noise transition matrix by simply splitting the K-class classification task into $\frac{K}{2}$ disjoint binary ones.

Symmetric noise transition matrix Symmetric noise model (Kim et al., 2019) is a widely accepted synthetic noise model in the literature of learning with noisy labels. The symmetric noise model generates the noisy labels by randomly flipping the clean label to the other possible classes with probability ϵ. $\forall i \neq j$, $T_{i,j} = \frac{\epsilon}{K-1}$, and the diagonal entry is $T_{i,i} = 1 - \epsilon$. Define the optimal r under the unified setting in the multi-class setting as $r_{\text{opt}} := \frac{(K-1)\cdot r^* - K\cdot\epsilon}{(K-1)-K\cdot\epsilon}$, Theorem 3.3 can be extended to the multi-class setting as:

Theorem 3.6. *Under Assumption 3.1, suppose the symmetric noise rate is not too large, i.e, $\epsilon < \frac{K-1}{K}$, learning with smooth rate $r = r_{opt}$ under $(X, \widetilde{Y}) \sim \widetilde{\mathcal{D}}$ yields $f^*_{\mathcal{D}}$:*

- *When noise rate $\epsilon < \frac{(K-1)\cdot r^*}{K}$, $r = r_{opt} > 0$ (LS);*
- *When noise rate $\epsilon = \frac{(K-1)\cdot r^*}{K}$, $r = 0$ (VL);*
- *When noise rate $\epsilon > \frac{(K-1)\cdot r^*}{K}$, $r = r_{opt} < 0$ (NLS).*

3.5. Clean empirical risk v.s. noisy empirical risk

Now we empirically verify Theorem 3.2 under symmetric noise setting, which relates the risk in the noisy setting to the clean ones. Assume the the noise label is generated through the symmetric noise transition matrix. We name the noisy risk as $\mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\ell(\mathbf{f}(X), \widetilde{Y}^{\text{GLS},r})\Big]$, which is the objective in Eqn. (4).

We use a UCI dataset (Waveform, binary classification) for illustration where the value of r^* is approximately 0. When the noise rates are $0.1, 0.2, 0.3, 0.4$, the optimal smooth rate

Table 1. Test accuracies of LS, VL, NLS on clean and noisy UCI Heart, Splice datasets, with best two smooth rates highlighted (green: NLS; red: VL or LS). We adopt the two independent sample T-test (5 non-negative smooth rates V.S. the last 5 rows of reported negative smooth rates) to verify the overall performance comparisons between VL/LS and NLS. p-value is highlighted in green if NLS generally returns a higher accuracy (i.e., t-value< 0) than VL/LS, otherwise, in red. Results on more benchmark datasets are given in Appendix D.

Smooth Rate	*UCI-Heart*					*UCI-Splice*				
	$e_i = 0$	$e_i = 0.1$	$e_i = 0.2$	$e_i = 0.3$	$e_i = 0.4$	$e_i = 0$	$e_i = 0.1$	$e_i = 0.2$	$e_i = 0.3$	$e_i = 0.4$
$r = 0.8$	0.885	0.853	0.836	0.820	0.738	0.980	0.946	0.919	0.856	0.760
$r = 0.6$	0.902	0.836	0.820	0.836	0.738	0.978	0.939	0.913	0.869	0.778
$r = 0.4$	0.885	0.853	0.836	0.820	0.771	0.978	0.948	0.922	0.885	0.797
$r = 0.2$	0.902	0.853	0.820	0.803	0.754	0.978	0.948	0.919	0.878	0.800
$r = 0.0$	0.902	0.853	0.820	0.820	0.771	0.976	0.948	0.926	0.876	0.806
$r = -0.4$	0.869	0.836	0.803	0.853	0.754	0.961	0.956	0.928	0.880	0.817
$r = -0.6$	0.869	0.836	0.820	0.853	0.721	0.961	0.956	0.926	0.880	0.819
$r = -1.0$	0.885	0.869	0.803	0.853	0.754	0.956	0.954	0.932	0.889	0.819
$r = -2.0$	0.885	0.869	0.820	0.853	0.787	0.952	0.946	0.935	0.898	0.830
$r = -4.0$	0.885	0.869	0.853	0.885	0.820	0.946	0.943	0.939	0.911	0.830
$r = -8.0$	0.869	0.869	0.885	0.853	0.853	0.943	0.946	0.939	0.915	0.845
$r_{\text{opt}} =$	[0.0, 0.6]	[-8.0, -1.0]	-8.0	-4.0	-8.0	0.8	[-0.6, -0.4]	[-8.0, -4.0]	-8.0	-8.0
p-value $=$	0.020	0.136	0.549	0.002	0.243	0.001	0.332	0.002	0.015	0.005

should be $-0.25, -0.67, -1.5, -4$ according to Eqn. (6). The estimated noisy risk of LS/VL/NLS on these noise settings can be summarized in Table 1. Clearly, when $e = 0.1$, $r = -0.25$ is closest to the estimated (clean) true risk (also returns the best test accuracy among these smooth rates). Similar observations hold for all other e. Learning with r_{opt} on the noisy data yields the closest risk to the corresponding clean risk with r^*!

Table 2. The difference between the empirical true risk of Y^* on the clean data and empirical risk of LS/VL/NLS on noisy labels (UCI-Waveform data): r^*, empirical true risk, and empirical noisy risks of r_{opt} under various noise levels are highlighted in purple.

Smooth rate	Risk (clean)	Risk ($e_i = 0.1$)	Risk ($e_i = 0.2$)	Risk ($e_i = 0.3$)
$r = 0.8$	0.6773	0.6831	0.6873	0.6899
$r = 0.6$	0.6295	0.6521	0.6689	0.6833
$r = 0.4$	0.5437	0.5994	0.6408	0.6718
$r = 0.2$	0.4134	0.5212	0.5956	0.6550
$\mathbf{r^* = 0.0}$	**0.1798**	0.4057	0.5399	0.6314
$r = -0.25$	-36.8095	**0.1983**	0.4381	0.5957
$r = -0.67$	-333.1283	-28.3508	**0.2167**	0.5132
$r = -1.5$	-97.4378	-61892.8047	-94.9509	**0.1911**

3.6. What is the practical distribution of r^* and r_{opt}?

r^* and r_{opt} on UCI datasets (Dua & Graff, 2017) As for UCI datasets, we pick Twonorm and Splice for illustration. The noisy labels are generated by a symmetric noise transition matrix with noise rate $e_i = [0.1, 0.2, 0.3, 0.4]$. As highlighted in Table 1 (top of this page), r_{opt} appears with positive values when the data is clean (same as r^*) or of a low noise rate. With the increasing of noise rates, the performance of LS results in a much larger degradation compared with NLS. We color-code different noise regimes where either VL/LS (red-ish) or NLS (green-ish) outperforms the other. Clearly there is a separation of the favored smoothing rate for different noise scenarios (upper left & low noise for VL/LS, bottom right & high noise for NLS).

r^* and r_{opt} on CIFAR datasets (Krizhevsky et al., 2009) When learning with a larger scale and more complex dataset, like CIFAR-10 and CIFAR-100, models are prone to converge on a local optimal solution rather than the global optimum. This phenomenon occurs frequently in NLS which ends up with performance degradation. Thus, in Table 3 and 4, when learning with noisy labels, we report the better performance of LS and NLS between direct training and loading the same warm-up model. We observe that the performance of NLS is more competitive than LS when learning with clean data. Clearly, NLS outperforms LS in CIFAR-10 and CIFAR-100 under various synthetic noise settings. The gap is larger when the noise rates are high. The results of two independent sample T-test [1] further verify this conclusion.

Table 3. Test accuracy (mean±std) comparisons on symmetric noisy CIFAR-10 datasets. Best two smooth rates for each synthetic noise setting are highlighted for each ϵ (green: NLS; red: VL/LS).

Smooth Rate	*CIFAR-10 Symmetric*			
	$\varepsilon = 0.0$	$\varepsilon = 0.2$	$\varepsilon = 0.4$	$\varepsilon = 0.6$
$r = 0.8$	92.91±0.06	88.88±1.61	81.48±2.91	73.16±0.16
$r = 0.6$	92.33±0.09	87.50±1.31	82.11±0.86	73.59±0.15
$r = 0.4$	93.05±0.04	87.13±0.07	81.50±1.42	74.21±0.19
$r = 0.0$	91.44±0.16	85.08±0.86	80.42±2.29	75.34±0.13
$r = -0.4$	93.55±0.06	87.55±0.08	81.58±0.19	75.95±0.13
$r = -0.8$	92.74±0.05	88.46±0.11	81.56±0.15	76.15±0.14
$r = -1.0$	92.58±0.08	88.58±0.08	81.95±0.10	76.20±0.10
$r = -2.0$	93.30±0.03	88.78±0.09	83.64±0.15	76.11±0.07
$r = -4.0$	93.13±0.04	88.90±0.07	84.34±0.13	77.22±0.09
$r = -6.0$	93.14±0.08	88.94±0.11	84.52±0.13	77.42±0.16
p-value $=$	0.0004	0.008	0.011	$< 1e-14$

[1] 4 non-negative smooth rates V.S. the smallest 4 negative smooth rates, p-value is highlighted in green if NLS generally returns a higher accuracy (i.e., t-value< 0) than VL/LS, otherwise, in red.

Table 4. Test accuracy (mean±std) comparisons on asymmetric noisy CIFAR-10, symmetric CIFAR-100 datasets. Best two smooth rates for each synthetic noise setting are highlighted for each ϵ (green: NLS; red: VL/LS).

Smooth Rate	*CIFAR-10 Asymmetric*		*CIFAR-100 Symmetric*	
	$\varepsilon = 0.2$	$\varepsilon = 0.3$	$\varepsilon = 0.4$	$\varepsilon = 0.6$
$r = 0.8$	90.45±0.06	87.83±0.13	54.04±0.93	39.50±0.18
$r = 0.6$	90.41±0.09	87.83±0.13	52.72±0.15	40.49±0.07
$r = 0.4$	90.49±0.10	87.90±0.13	54.26±0.07	41.57±0.05
$r = 0.0$	88.32±0.24	86.27±0.32	48.03±0.29	38.11±0.14
$r = -0.4$	87.27±1.83	88.33±0.06	56.87±0.08	43.70±0.16
$r = -0.8$	86.40±1.32	87.96±0.43	57.35±0.08	44.10±0.06
$r = -1.0$	88.47±0.15	87.50±0.73	57.44±0.09	43.85±0.19
$r = -2.0$	88.66±0.17	87.27±0.70	58.10±0.08	44.88±0.11
$r = -4.0$	89.56±0.17	87.29±0.59	58.35±0.09	46.38±0.05
$r = -6.0$	89.70±0.24	87.57±0.42	57.73±0.10	46.46±0.09
p-value =	$< 1e-7$	0.106	$< 1e-14$	$< 1e-15$

4. The Impacts on the Model Confidence

Continuing the discussion of differed model confidence in the previous section, we now empirically explore how such differences distinguish LS and NLS.

Remember that when the label is clean ($e_0 = e_1 = 0$), Eqn. (3) reduces to:

$$\min_{f\in\mathcal{F}} \mathbb{E}_{(X,Y)\sim\mathcal{D}}\Big[\ell(\mathbf{f}(X), Y)\Big] + \frac{r}{2}\cdot \mathbb{E}_{(X,Y)\sim\mathcal{D}} \underbrace{\Big[\ell(\mathbf{f}(X), 1-Y) - \ell(\mathbf{f}(X), Y)\Big]}_{\text{Term MC}_\ell(f;X,Y)}. \quad (7)$$

The difference between LS and NLS lie in the weight of Term $\text{MC}_\ell(f; X, Y)$ when learning with clean labels: NLS encourages high $\text{MC}_\ell(f; X, Y)$ and $\text{MC}(f; X, Y)$ while LS has an opposite effect.

4.1. Side-effects of over-confident

We adopt the generation of 2D (binary) synthetic dataset from (Amid et al., 2019) by randomly sampling two circularly distributed classes. The inner annulus indicates one class (blue), while the outer annulus denotes the other class (red). We hold 20% data samples for performance comparison.

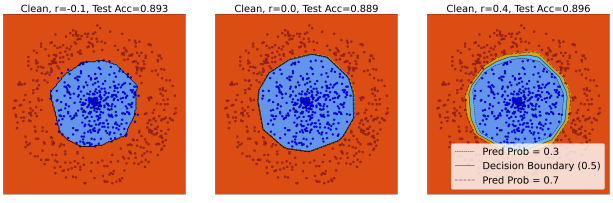

Figure 3. Model confidence visualization of NLS, VL, and LS on synthetic data (Type 1) with the clean data. The optimal smooth rate falls in $[0, 0.4]$. (left: NLS; middle: Vanilla Loss; right: LS). The test accuracy is annotated above each plot.

In Figure 3, the colored bands depict the different levels of prediction probabilities: light blue + orange bands indicate samples that satisfy MC < 0.4 (low model confidence). When learning with the clean data, a non-positive smooth rate may yield over-confidence on the model prediction and a relatively low test accuracy.

4.2. Label noise reduces model confidence

Recent works (Liu, 2021; Cheng et al., 2021a) have demonstrated that with the presence of label noise, learning with noisy labels directly will eventually result in unconfident model predictions. Continuing the synthetic 2D dataset, we flip the clean labels according to a symmetric noise transition matrix with noise rate e_i for both classes. With the presence of label noise in Figure 4, the trained models generally become less confident on its predictions. Besides, when the smooth rate increases from negative to positive, more samples are of uncertain predictions. Thus, a smaller/negative smooth rate is beneficial when the noise rate increases by encouraging more confident predictions.

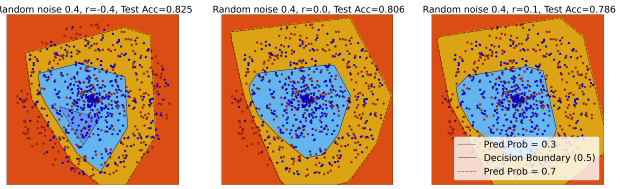

Figure 4. Model confidence visualization of NLS, VL, and LS on synthetic data (Type 1) with noise rate $e_i = 0.4$. The optimal smooth rate is -0.4. (left: NLS; middle: Vanilla Loss; right: LS). The test accuracy is annotated above each plot.

4.3. Model confidence on CIFAR-10 test dataset

When trained on symmetric 0.2 noisy CIFAR-10 training dataset (see Figure 5), with the decreasing of smooth rates (from right to left), the model confidence on correct predictions gradually approach to its maximum, while for wrong predictions, the model confidence converges to its minimum value. We observe that NLS makes the model prediction become over-confident on correct predictions and in-confident on wrong predictions.

5. Connection to Other Robust Methods

In this section, we aim to theoretically explore the connection between NLS and popular robust methods such as backward/forward loss correction (Natarajan et al., 2013; Patrini et al., 2017), NLNL (Kim et al., 2019) and peer loss (Liu & Guo, 2020), under the unified setting. We defer the corresponding empirical verification to Appendix B.

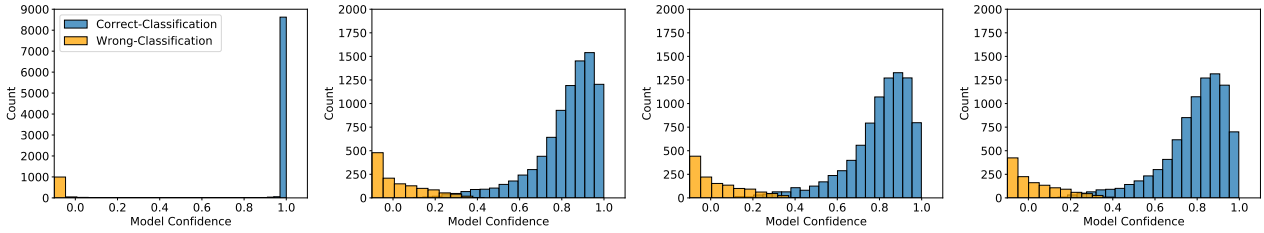

Figure 5. Model confidence distribution of correct and wrong predictions on CIFAR-10 test data samples. (From left to right: NLS ($r = -0.8, -0.4$), Vanilla Loss, LS ($r = 0.4$), trained on symmetric 0.2 noisy CIFAR-10 dataset).

5.1. Loss correction

Loss correction (Patrini et al., 2017) studies two robust loss designs which are based on the knowledge of non-singular noise transition matrix T. The backward correction $\ell^{\leftarrow}(\mathbf{f}(X), \widetilde{Y})$ re-weights the loss $\ell(\mathbf{f}(X), \widetilde{Y})$ by $T^{-1}_{\hat{Y},\widetilde{Y}}$ with $\hat{Y}$ being the model predicted label, while the proposed forward correction $\ell^{\rightarrow}(\mathbf{f}(X), \widetilde{Y})$ multiplies the model predictions by T.

Proposition 5.1. *For $r_{LC} := \frac{2e_0}{2e_0-1} < 0$, $\lambda_{LC} := e_\Delta \cdot \frac{1}{1-2e_0}$, risk minimization of both backward and forward correction (with the knowledge of noise rates) are equivalent to the combination of NLS and an extra bias term Bias-LC*

$$\begin{aligned}
&\min_{f\in\mathcal{F}} \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\left[\ell^{\leftarrow}(\mathbf{f}(X), \widetilde{Y})\right] \\
=&\min_{f\in\mathcal{F}} \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\left[\ell^{\rightarrow}(\mathbf{f}(X), \widetilde{Y})\right] \\
=&\min_{f\in\mathcal{F}} \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\left[\ell(\mathbf{f}(X), \widetilde{Y}^{GLS,r_{LC}})\right] \\
&+ \lambda_{LC} \cdot \underbrace{\mathbb{E}_{X,Y=1}\left[\ell(\mathbf{f}(X), 1) - \ell(\mathbf{f}(X), 0)\right]}_{\textit{Bias-LC}}.
\end{aligned}$$

The incurred Bias-LC controls the model confidence on $(X, Y = 1) \sim \mathcal{D}_f$. Note that when the noise rate is not substantially high, i.e., $e_0 \in [0, \frac{1}{2})$, $\lambda_{\text{LC}} > 0$. Then, compared with loss correction, NLS with smooth rate r_{LC} makes the model f to be less confident on $(X, Y = 1) \sim \mathcal{D}_f^+$ and more confident on $(X, Y = 1) \sim \mathcal{D}_f^-$ (wrong predictions). However, the impact of term Bias-LC is diminishing when either $e_\Delta \to 0$ (symmetric noise rates) or $e_0 \to 0$ (low noise rates) as specified in Theorem 5.2.

Theorem 5.2. *Assume the noise transition matrix is symmetric, i.e., $e_\Delta = 0$, backward and forward loss correction are a special form of NLS with smooth rate r_{LC}.*

5.2. Learning from complementary labels

Complementary labels (Ishida et al., 2017) were firstly introduced to mitigate the cost of collecting data. Rather than encouraging the model to fit directly on the target, learning from complementary labels trains the model to not fit on the complementary label which differs from the target. Later, an indirect training method "Negative Learning" (NL) (Kim et al., 2019) was proposed to reduce the risk of providing incorrect information with the presence of noisy labels and is robust to label noise in multi-class classification tasks. A more generic unbiased risk estimator of learning with complementary labels was proposed (Ishida et al., 2019), a popular case is: $\ell_{\text{CL}}(\mathbf{f}(X), \widetilde{Y}) := \ell(\mathbf{f}(X), \widetilde{Y}) - \ell(\mathbf{f}(X), 1 - \widetilde{Y})$.

Theorem 5.3. *Learning from complementary labels with ℓ_{CL} is equivalent to NLS with smooth rate $r_{CL} \to -\infty$:*

$$\begin{aligned}
&\min_{f\in\mathcal{F}} \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\left[\ell_{CL}(\mathbf{f}(X), \widetilde{Y})\right] \\
=&\min_{f\in\mathcal{F}} \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}[\ell(\mathbf{f}(X), \widetilde{Y}^{GLS,r_{CL}\to-\infty})].
\end{aligned}$$

5.3. Peer loss functions

Peer loss functions (Liu & Guo, 2020) proposed a family of robust loss measures which do not require the knowledge of noise rates. The mathematical representation of peer loss functions is $\ell_{\text{PL}}(\mathbf{f}(X), \widetilde{Y}) := \ell(\mathbf{f}(X), \widetilde{Y}) - \ell(\mathbf{f}(X_1), \widetilde{Y}_2)$, where $(X_i, \widetilde{Y}_i) \sim \widetilde{\mathcal{D}}$. The second term of the peer loss evaluates on randomly paired data samples and labels (X_1 and $\widetilde{Y}_2$ for two randomly selected samples) to punish f from overly fitting on noisy labels.

Proposition 5.4. *For $r_{PL} := 2 \cdot \mathbb{P}(\widetilde{Y} = 1)$, $\lambda_{PL} := 1 - r_{PL}$, risk minimization of peer loss is equivalent to negative label smoothing regularization with an extra term Bias-PL, i.e.,*

$$\begin{aligned}
&\min_{f\in\mathcal{F}} \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\left[\ell_{PL}(\mathbf{f}(X), \widetilde{Y})\right] \\
=&\min_{f\in\mathcal{F}} \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\left[\ell(\mathbf{f}(X), \widetilde{Y}) - \ell(\mathbf{f}(X), \widetilde{Y}^{GLS,r_{PL}})\right] \\
&+ \lambda_{PL} \cdot \underbrace{\mathbb{E}_{X,\widetilde{Y}=1}\left[\ell(\mathbf{f}(X), 1) - \ell(\mathbf{f}(X), 0)\right]}_{\textit{Bias-PL}}.
\end{aligned}$$

The incurred term Bias-PL controls the model confidence on $(X, \widetilde{Y} = 1) \sim \widetilde{\mathcal{D}}$ and has a diminishing effect as $\mathbb{P}(\widetilde{Y} = 1) \to 1/2$. Generally, the peer loss relates to the unified setting (GLS) as the negatively weighted GLS term appears to be a regularizer. Note that we have access to the

Table 5. Performance comparisons on synthetic noisy CIFAR datasets: we adopt the same model architecture for all methods (ResNet 34 (He et al., 2016)), best achieved test accuracy is reported.

Method	*CIFAR-10, Symmetric*			*CIFAR-10, Asymmetric*		*CIFAR-100, Symmetric*	
	$\varepsilon = 0.2$	$\varepsilon = 0.4$	$\varepsilon = 0.6$	$\varepsilon = 0.2$	$\varepsilon = 0.3$	$\varepsilon = 0.4$	$\varepsilon = 0.6$
Cross Entropy	86.45	82.72	74.04	88.59	86.14	48.20	38.27
Bootstrap (Reed et al., 2014)	86.06	81.65	75.26	87.69	85.51	47.28	35.81
FLC (Patrini et al., 2017)	84.85	84.98	73.97	89.42	88.25	53.04	41.59
SCE (Wang et al., 2019)	89.39	80.31	75.28	88.07	85.93	49.34	38.87
APL (Ma et al., 2020)	88.42	81.27	76.62	88.75	87.41	51.63	42.31
Peer Loss (Liu & Guo, 2020)	90.21	86.40	79.64	91.38	89.65	62.16	53.72
ELR (Liu et al., 2020)	92.57	91.32	88.86	93.48	92.21	68.03	60.49
AUM (Pleiss et al., 2020)	91.52	87.85	81.71	92.17	90.63	59.29	44.05
Label Smoothing (LS) (Lukasik et al., 2020)	90.24	83.78	75.01	90.61	88.04	55.17	41.63
Negative/Not Label Smoothing (NLS)	89.05	84.85	77.82	90.02	88.42	58.47	46.58

$\mathbb{P}(\widetilde{Y} = 1)$, we can bridge the gap by adding an estimable term Bias-PL. With some derivations, we further show in Theorem 5.5, when noisy priors are equal, the peer loss has an exact NLS form.

Theorem 5.5. *When the noisy labels have equal prior, i.e.,* $\mathbb{P}(\widetilde{Y} = 0) = \mathbb{P}(\widetilde{Y} = 1)$*, the peer loss is a special form of NLS regularization with the smooth rate* r_{PL}*. Besides,*

$$\min_{f \in \mathcal{F}} \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}} \left[\ell_{PL}(\mathbf{f}(X), \widetilde{Y}) \right] = \min_{f \in \mathcal{F}} \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}} \left[\ell(\mathbf{f}(X), \widetilde{Y}^{GLS, r \to -\infty}) \right].$$

5.4. Practical significance

In Table 5, we compare VL(CE), LS and NLS with several robust methods in synthetic noisy CIFAR datasets. Clearly, LS and NLS can be viewed as competitive and efficient robust loss functions which outperform Cross Entropy, Bootstrap (Reed et al., 2014), SCE (Wang et al., 2019), APL (Ma et al., 2020) and Forward loss correction (FLC) (Patrini et al., 2017) in most settings.

We also provide experimental results of LS and NLS on real-world human noise benchmarks: CIFAR-N (Wei et al., 2022b) and Clothing 1M (Xiao et al., 2015), along with several baseline methods for comparisons, i.e., backward loss correction (BLC) (Natarajan et al., 2013; Patrini et al., 2017), forward loss correction (FLC) (Patrini et al., 2017), Peer Loss (PL) (Liu & Guo, 2020), and F-div (Wei & Liu, 2021). Table 6 demonstrates the effectiveness of NLS. Besides, we observe that NLS ranks 4-th among 21 existing robust methods on Clothing 1M (no extra train data, evaluated on the clean test data)[2]. This simple trick clearly reveals the importance and great potential of NLS. Nonetheless we would like to clarify that our main purposes are (instead of chasing SOTA): (1) Provide new understandings of whether we should smooth the label or not when learning with noisy labels. (2) Reveal the importance and effectiveness of NLS at different scenarios. The popularity of label smoothing is largely due to its simplicity and being complementary, so we expect our observations for NLS can be combined with other SOTA methods to further improve model performance in the high-noise regime.

[2]Public leaderboard of CIFAR-N, Clothing 1M: http://noisylabels.com/, https://paperswithcode.com/sota/image-classification-on-clothing1m

Table 6. Performance comparisons on Clothing 1M and CIFAR-N: results of baselines are obtained through the public leader-board.

Method	**Clothing 1M**	**CIFAR-10N Aggre**	**CIFAR-10N Rand1**	**CIFAR-10N Worse**	**CIFAR-100N Fine**
CE	68.94	87.77	85.02	77.69	55.50
BLC	69.13	88.13	87.14	77.61	57.14
FLC	69.84	88.24	86.88	79.79	57.01
PL	72.60	90.75	89.06	82.53	57.59
F-div	73.09	91.64	89.70	82.53	57.10
LS (best)	73.44	91.57	89.80	82.76	55.84
NLS (best)	**74.24**	**91.97**	**90.29**	**82.99**	**58.59**

6. Conclusion

In this paper, we provide understandings for whether should we adopt label smoothing or not when learning with noisy labels. We show that learning with negatively smoothed labels explicitly improves the confidence of model prediction. This key property acts as a significant role when the confidence of model prediction drops. In contrast to existing works that promote the use of positive label smoothing, we show both theoretically and empirically the advantage of a negative smooth rate when the label noise rate increases. We also bridge the gap between negative label smoothing and existing learning with noisy label solutions, which further demonstrates the importance of negative/not label smoothing. In a nutshell, our observations provide new understanding for the effects of label smoothing, especially when the training labels are imperfect. Future works include exploring the benefits of negative labels in other tasks.

Acknowledgement YL and JHW are partially supported by the grants IIS-2007951 and IIS-2143895. TLL is partially supported by Australian Research Council Projects DE-190101473, IC-190100031, and DP-220102121. MS and GN are supported by JST CREST Grant Number JPMJCR18A2. The authors thank anonymous ICML reviewers for their comments that improved the presentation.

References

Amid, E., Warmuth, M. K., Anil, R., and Koren, T. Robust bi-tempered logistic loss based on bregman divergences. In *Advances in Neural Information Processing Systems*, pp. 14987–14996, 2019.

Bai, Y., Yang, E., Han, B., Yang, Y., Li, J., Mao, Y., Niu, G., and Liu, T. Understanding and improving early stopping for learning with noisy labels. *Advances in Neural Information Processing Systems*, 34, 2021.

Berthon, A., Han, B., Niu, G., Liu, T., and Sugiyama, M. Confidence scores make instance-dependent label-noise learning possible. In *International Conference on Machine Learning*, pp. 825–836. PMLR, 2021.

Cheng, D., Liu, T., Ning, Y., Wang, N., Han, B., Niu, G., Gao, X., and Sugiyama, M. Instance-dependent label-noise learning with manifold-regularized transition matrix estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16630–16639, 2022.

Cheng, H., Zhu, Z., Li, X., Gong, Y., Sun, X., and Liu, Y. Learning with instance-dependent label noise: A sample sieve approach. In *International Conference on Learning Representations*, 2021a. URL https://openreview.net/forum?id=2VXyy9mIyU3.

Cheng, H., Zhu, Z., Sun, X., and Liu, Y. Demystifying how self-supervised features improve training from noisy labels. *arXiv preprint arXiv:2110.09022*, 2021b.

Chorowski, J. and Jaitly, N. Towards better decoding and language model integration in sequence to sequence models. *Proc. Interspeech 2017*, pp. 523–527, 2017.

Dawson, G. and Polikar, R. Rethinking noisy label models: Labeler-dependent noise with adversarial awareness. *arXiv preprint arXiv:2105.14083*, 2021.

Dua, D. and Graff, C. UCI machine learning repository, 2017. URL http://archive.ics.uci.edu/ml.

Englesson, E. and Azizpour, H. Consistency regularization can improve robustness to label noise. *arXiv preprint arXiv:2110.01242*, 2021a.

Englesson, E. and Azizpour, H. Generalized jensen-shannon divergence loss for learning with noisy labels. *Advances in Neural Information Processing Systems*, 34, 2021b.

Feng, L., Lv, J., Han, B., Xu, M., Niu, G., Geng, X., An, B., and Sugiyama, M. Provably consistent partial-label learning. *Advances in Neural Information Processing Systems*, 33:10948–10960, 2020.

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

Han, B., Yao, Q., Yu, X., Niu, G., Xu, M., Hu, W., Tsang, I., and Sugiyama, M. Co-teaching: Robust training of deep neural networks with extremely noisy labels. In *Advances in neural information processing systems*, pp. 8527–8537, 2018.

Harish, R., Scott, C., and Tewari, A. Mixture proportion estimation via kernel embeddings of distributions. *In International conference on machine learning*, pp. 2052–2060, 2016.

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

Hinton, G., Vinyals, O., and Dean, J. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Ishida, T., Niu, G., Hu, W., and Sugiyama, M. Learning from complementary labels. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pp. 5644–5654, 2017.

Ishida, T., Niu, G., Menon, A., and Sugiyama, M. Complementary-label learning for arbitrary losses and models. In *International Conference on Machine Learning*, pp. 2971–2980. PMLR, 2019.

Jiang, L., Zhou, Z., Leung, T., Li, L.-J., and Fei-Fei, L. Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels. In *International Conference on Machine Learning*, pp. 2304–2313. PMLR, 2018.

Jiang, Z., Zhou, K., Liu, Z., Li, L., Chen, R., Choi, S.-H., and Hu, X. An information fusion approach to learning with instance-dependent label noise. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=ecH2FKaARUp.

Kim, Y., Yim, J., Yun, J., and Kim, J. Nlnl: Negative learning for noisy labels. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 101–110, 2019.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Krizhevsky, A., Hinton, G., et al. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009.

Kumar, A. and Amid, E. Constrained instance and class reweighting for robust learning under label noise. *arXiv preprint arXiv:2111.05428*, 2021.

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

Liu, S., Niles-Weed, J., Razavian, N., and Fernandez-Granda, C. Early-learning regularization prevents memorization of noisy labels. *Advances in neural information processing systems*, 33:20331–20342, 2020.

Liu, S., Li, X., Zhai, Y., You, C., Zhu, Z., Fernandez-Granda, C., and Qu, Q. Convolutional normalization: Improving deep convolutional network robustness and training. *Advances in Neural Information Processing Systems*, 34, 2021.

Liu, S., Zhu, Z., Qu, Q., and You, C. Robust training under label noise by over-parameterization. *arXiv preprint arXiv:2202.14026*, 2022.

Liu, T. and Tao, D. Classification with noisy labels by importance reweighting. *IEEE Transactions on pattern analysis and machine intelligence*, 38(3):447–461, 2015.

Liu, T. and Tao, D. Classification with noisy labels by importance reweighting. *IEEE Transactions on pattern analysis and machine intelligence*, 38(3):447–461, 2016.

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

Liu, Y. and Guo, H. Peer loss functions: Learning from noisy labels without knowing noise rates. In *International Conference on Machine Learning*, pp. 6226–6236. PMLR, 2020.

Liu, Y. and Wang, J. Can less be more? when increasing-to-balancing label noise rates considered beneficial. *Advances in Neural Information Processing Systems*, 34, 2021.

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

Lv, J., Feng, L., Xu, M., An, B., Niu, G., Geng, X., and Sugiyama, M. On the robustness of average losses for partial-label learning. *arXiv preprint arXiv:2106.06152*, 2021.

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

Majidi, N., Amid, E., Talebi, H., and Warmuth, M. K. Exponentiated gradient reweighting for robust training under label noise and beyond. *arXiv preprint arXiv:2104.01493*, 2021.

Menon, A., Van Rooyen, B., Ong, C. S., and Williamson, B. Learning from corrupted binary labels via class-probability estimation. In *International Conference on Machine Learning*, pp. 125–134, 2015.

Müller, R., Kornblith, S., and Hinton, G. When does label smoothing help? In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, pp. 4696–4705, 2019.

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

Northcutt, C., Jiang, L., and Chuang, I. Confident learning: Estimating uncertainty in dataset labels. *Journal of Artificial Intelligence Research*, 70:1373–1411, 2021.

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

Pereyra, G., Tucker, G., Chorowski, J., Kaiser, Ł., and Hinton, G. Regularizing neural networks by penalizing confident output distributions. *arXiv preprint arXiv:1701.06548*, 2017.

Pleiss, G., Zhang, T., Elenberg, E. R., and Weinberger, K. Q. Identifying mislabeled data using the area under the margin ranking. *arXiv preprint arXiv:2001.10528*, 2020.

Reed, S., Lee, H., Anguelov, D., Szegedy, C., Erhan, D., and Rabinovich, A. Training deep neural networks on noisy labels with bootstrapping. *arXiv preprint arXiv:1412.6596*, 2014.

Robbins, H. and Monro, S. A stochastic approximation method. *The annals of mathematical statistics*, pp. 400–407, 1951.

Scott, C., Blanchard, G., Handy, G., Pozzi, S., and Flaska, M. Classification with asymmetric label noise: Consistency and maximal denoising. In *COLT*, pp. 489–511, 2013.

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

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pp. 6000–6010, 2017.

Wang, H., Xiao, R., Li, Y., Feng, L., Niu, G., Chen, G., and Zhao, J. PiCO: Contrastive label disambiguation for partial label learning. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=EhYjZy6e1gJ.

Wang, J., Liu, Y., and Levy, C. Fair classification with group-dependent label noise. In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, pp. 526–536, 2021.

Wang, Y., Ma, X., Chen, Z., Luo, Y., Yi, J., and Bailey, J. Symmetric cross entropy for robust learning with noisy labels. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 322–330, 2019.

Wei, H., Feng, L., Chen, X., and An, B. Combating noisy labels by agreement: A joint training method with co-regularization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13726–13735, 2020.

Wei, H., Tao, L., Xie, R., and An, B. Open-set label noise can improve robustness against inherent label noise. *Advances in Neural Information Processing Systems*, 34, 2021.

Wei, H., Xie, R., Feng, L., Han, B., and An, B. Deep learning from multiple noisy annotators as a union. *IEEE Transactions on Neural Networks and Learning Systems*, 2022a.

Wei, J. and Liu, Y. When optimizing f-divergence is robust with label noise. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=WesiCoRVQ15.

Wei, J., Zhu, Z., Cheng, H., Liu, T., Niu, G., and Liu, Y. Learning with noisy labels revisited: A study using real-world human annotations. In *International Conference on Learning Representations*, 2022b. URL https://openreview.net/forum?id=TBWA6PLJZQm.

Wei, J., Zhu, Z., Luo, T., Amid, E., Kumar, A., and Liu, Y. To aggregate or not? learning with separate noisy labels, 2022c. URL https://arxiv.org/abs/2206.07181.

Xia, X., Liu, T., Han, B., Gong, C., Wang, N., Ge, Z., and Chang, Y. Robust early-learning: Hindering the memorization of noisy labels. In *International conference on learning representations*, 2020a.

Xia, X., Liu, T., Han, B., Wang, N., Deng, J., Li, J., and Mao, Y. Extended T: Learning with mixed closed-set and open-set noisy labels. *arXiv preprint arXiv:2012.00932*, 2020b.

Xia, X., Liu, T., Han, B., Gong, M., Yu, J., Niu, G., and Sugiyama, M. Instance correction for learning with open-set noisy labels. *arXiv preprint arXiv:2106.00455*, 2021a.

Xia, X., Liu, T., Han, B., Gong, M., Yu, J., Niu, G., and Sugiyama, M. Sample selection with uncertainty of losses for learning with noisy labels. *arXiv preprint arXiv:2106.00445*, 2021b.

Xiao, T., Xia, T., Yang, Y., Huang, C., and Wang, X. Learning from massive noisy labeled data for image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2691–2699, 2015.

Xu, Y., Xu, Y., Qian, Q., Li, H., and Jin, R. Towards understanding label smoothing. *arXiv preprint arXiv:2006.11653*, 2020.

Yang, S., Yang, E., Han, B., Liu, Y., Xu, M., Niu, G., and Liu, T. Estimating instance-dependent label-noise transition matrix using dnns. *arXiv preprint arXiv:2105.13001*, 2021.

Yang, Z., Yu, Y., You, C., Steinhardt, J., and Ma, Y. Rethinking bias-variance trade-off for generalization of neural networks. In *International Conference on Machine Learning*, pp. 10767–10777. PMLR, 2020.

Yao, Q., Yang, H., Han, B., Niu, G., and Kwok, J. T. Searching to exploit memorization effect in learning with noisy labels. In *Proceedings of the 37th International Conference on Machine Learning*, ICML '20, 2020a.

Yao, Y., Liu, T., Han, B., Gong, M., Deng, J., Niu, G., and Sugiyama, M. Dual t: Reducing estimation error for transition matrix in label-noise learning. In *Advances in Neural Information Processing Systems*, volume 33, pp. 7260–7271, 2020b.

Yi, L., Liu, S., She, Q., McLeod, A. I., and Wang, B. On learning contrastive representations for learning with noisy labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16682–16691, 2022.

Yu, X., Han, B., Yao, J., Niu, G., Tsang, I., and Sugiyama, M. How does disagreement help generalization against label corruption? In *Proceedings of the 36th International Conference on Machine Learning*, volume 97, pp. 7164–7173. PMLR, 09–15 Jun 2019.

Yuan, L., Tay, F. E., Li, G., Wang, T., and Feng, J. Revisiting knowledge distillation via label smoothing regularization.

In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3903–3911, 2020.

Zhang, X., Zhao, J., and LeCun, Y. Character-level convolutional networks for text classification. *Advances in neural information processing systems*, 28, 2015.

Zhou, H., Song, L., Chen, J., Zhou, Y., Wang, G., Yuan, J., and Zhang, Q. Rethinking soft labels for knowledge distillation: A bias–variance tradeoff perspective. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=gIHd-5X324.

Zhu, Z., Dong, Z., Cheng, H., and Liu, Y. A good representation detects noisy labels. *arXiv preprint arXiv:2110.06283*, 2021a.

Zhu, Z., Liu, T., and Liu, Y. A second-order approach to learning with instance-dependent label noise. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10113–10123, 2021b.

Zhu, Z., Song, Y., and Liu, Y. Clusterability as an alternative to anchor points when learning with noisy labels. In *Proceedings of the 38th International Conference on Machine Learning*, ICML '21, 2021c.

Zhu, Z., Wang, J., and Liu, Y. Beyond images: Label noise transition matrix estimation for tasks with lower-quality features. *arXiv preprint arXiv:2202.01273*, 2022.

Appendix

The Appendix is organized as follows.

- Section A presents the full version of related works.
- Section B includes empirical validations of theoretical conclusions in Section 5.
- Section C discusses practical considerations of the robustness for LS and NLS.
- Section D shows additional experiments on synthetic dataset and UCI datasets.
- Section E illustrates the bias and variance trade-off when learning with LS and NLS from clean data.
- Section F includes omitted proofs for theoretical conclusions in the main paper.

A. Full Version of Related Works

Our work supplements to two lines of related works.

Learning with noisy labels Annotated labels from human labelers usually consists of an non-negligible amount of mis-labeled data samples. Making deep neural nets perform robust training on "noisily" labeled datasets remains a challenge. Classical approaches of learning with noisy labels assume the noisy labels are independent to features. They firstly estimate the noise transition matrix (Liu & Tao, 2015; Menon et al., 2015; Harish et al., 2016; Patrini et al., 2017; Zhu et al., 2021a;c; Yang et al., 2021; Cheng et al., 2022; Zhu et al., 2022), then proceed with a loss correction (Natarajan et al., 2013; Patrini et al., 2017; Liu & Tao, 2015) to mitigate label noise. Recent works mainly focus on: (1) proposing robust loss functions (Kim et al., 2019; Liu & Guo, 2020; Wei & Liu, 2021; Englesson & Azizpour, 2021b;a) to train deep neural nets directly without the knowledge of noise rates, or design a pipeline which dynamically select and train on "clean" samples with small loss (Jiang et al., 2018; Han et al., 2018; Yu et al., 2019; Yao et al., 2020a; Xia et al., 2021b); (2) hindering the memorization on noisy labels (Xia et al., 2020a; Liu et al., 2020; Cheng et al., 2021b; Liu et al., 2021; Wei et al., 2021; Bai et al., 2021; Liu et al., 2022; Yi et al., 2022); (3) sample-level re-weighting to mitigate the impacts of wrong labels (Liu & Tao, 2016; Majidi et al., 2021; Kumar & Amid, 2021; Liu & Wang, 2021). More recently, several approaches target at addressing more challenging noise settings, such as group/instance-dependent label noise (Cheng et al., 2021a; Wang et al., 2021; Berthon et al., 2021; Zhu et al., 2021b; Dawson & Polikar, 2021; Jiang et al., 2022), or considering more practical applications such as open-set data (Xia et al., 2020b; Wei et al., 2021; Xia et al., 2021a), partial label learning (Feng et al., 2020; Lv et al., 2021; Wang et al., 2022), samples with multiple noisy annotations (Wei et al., 2022a;c).

Understanding the effect of label smoothing Learning with one-hot labels is prone to over-fitting, soft label learning then naturally draws attentions of machine learning researchers. Successful applications of soft label learning include the label distribution learning (Geng, 2016) which provides an instance with description degrees of all the labels. Label smoothing (LS) (Szegedy et al., 2016) is another arising learning paradigm that uses positively weighted average of both the hard training labels and uniformly distributed soft labels. Empirical studies have demonstrated the effectiveness of LS in improving the model performance (Pereyra et al., 2017; Szegedy et al., 2016; Vaswani et al., 2017; Chorowski & Jaitly, 2017) and model calibration (Müller et al., 2019). However, knowledge distilling a teacher network (trained on smoothed labels) into a student network is much less effective (Müller et al., 2019). Later, generalization effects of more advanced forms of label smoothing was studied, such as structural label smoothing (Li et al., 2020). More recently, it was shown that an appropriate label smoothing regularizer with reduced label variance boosts the convergence (Xu et al., 2020). When label noise presents, (Liu, 2021) gives theoretical justifications for the memorizing effects of label smoothing. And the effectiveness of label smoothing in mitigating label noise is investigated in (Lukasik et al., 2020).

B. Empirical Validations of Main Theorems

In this section, we empirically validate our main theoretical conclusions in Section 5, i.e, the connection between LS/NLS and popular methods.

We compare the unified setting (GLS) with backward correction (Natarajan et al., 2013), forward correction (Patrini et al., 2017) and peer loss (Liu & Guo, 2020) on CIFAR-10 dataset. To approximate the performance of backward/forward Loss Correction, we adopt GLS with smooth rate $\frac{\epsilon}{(\epsilon-1)}$. As for the approximation of peer loss, we choose $\ell(\mathbf{f}(X), \widetilde{Y}) -$

$\ell(\mathbf{f}(X), \widetilde{Y}^{\text{GLS},r=0.5})$ which is equivalent to NLS when $r \to -\infty$. Experiment results in Table 7 on CIFAR-10 under symmetric noise settings demonstrate that the equivalent forms of GLS are robust to label noise.

Table 7. Comparison of test accuracies on CIFAR-10 under symmetric label noise.

Method	*CIFAR-10, Symmetric*		
	$\varepsilon = 0.2$	$\varepsilon = 0.4$	$\varepsilon = 0.6$
Backward T (Patrini et al., 2017)	84.79	83.40	71.52
Forward T (Patrini et al., 2017)	84.85	**83.98**	73.97
GLS form	**87.33**	81.73	**75.80**
Peer Loss (Liu & Guo, 2020)	**90.21**	**86.40**	**79.64**
GLS form	88.98	85.05	76.66

Explanation of the performance gap In practice, we adopt the same hyper-parameter setting as used for all other smooth rates for GLS form (VL, LS and NLS). Loss corrections will firstly warm-up with the cross-entropy loss, estimate the noise transition matrix with this pre-trained model, and then proceed to train with the backward/forward corrected loss. Peer loss functions adopt a dynamical adjustment for learning rate. The warming up, estimation error of noise transition matrix as well as the special hyper-parameter settings explain performance gaps.

C. Practical Consideration of LS and NLS

In the main paper, we theoretically show when we should adopt NLS and LS. In this section, we discuss more practical considerations, including the optimal smoothing parameter, how to reduce the impacts of bias terms, and multi-class extensions.

C.1. The optimal smoothing parameter

In practice, we don't have access to noise rates e_i. Our work does not intend to particularly focus on the noise rate estimation. For readers interested in the noise rate estimation, please refer to (Liu & Tao, 2015; Menon et al., 2015; Harish et al., 2016; Patrini et al., 2017; Yao et al., 2020b; Zhu et al., 2021c). To estimate $r_{\text{opt}} = \frac{r^*-2e}{1-2e}$, one can simply assume $r^* \to 0$. And the noise rate e is estimable by a large family of noise estimation methods mentioned above. Our practical observations show that NLS with a CE warm-up is not sensitive to the negative smooth rate, for example, on CIFAR-10 and CIFAR-100 synthetic noisy datasets, $r < -1.0$ frequently achieves best results (see Table 3 in the main paper). Our current contribution focuses on understanding the generalized label smoothing, and we prefer leaving the task of identifying the optimal smooth rate to future works.

C.2. Making LS and NLS more robust to label noise

There is a line of related works targeting at distinguishing clean labels from the noisy labels. Current literature in selecting clean samples from noisily labeled dataset is based on the empirical evidence that samples with noisy/wrong labels have a larger loss than clean ones. For interested readers, please refer to (Han et al., 2018; Jiang et al., 2018; Yu et al., 2019; Yao et al., 2020a; Wei et al., 2020; Northcutt et al., 2021). Compared with the risk minimization over the clean data distribution $(X, Y) \sim \mathcal{D}$, learning directly with GLS on the noisy distribution $(X, \widetilde{Y}) \sim \widetilde{\mathcal{D}}$ will result in an extra term $(e_1 - e_0) \cdot (1 - r) \cdot \mathbb{E}_{(X,Y=1)\sim\mathcal{D}}[\ell(\mathbf{f}(\mathbf{X}), 0) - \ell(\mathbf{f}(\mathbf{X}), 1)]$ compared to the clean scenario. Empirically, we can estimate the bias term, perform a bias correction by subtracting the estimated bias term from the objective function in Eqn. (3).

Suppose we have access to a clean distribution $\mathcal{D}_{\text{clean}}$ which consists of selected clean samples. Denote the estimated noise rates as $\hat{e}_i$, when $e_\Delta \neq 0$, in order to make LS/NLS be more robust to label noise and fit on the optimal distribution Y^*, we improve by performing a model confidence correction on the dominating class through:

$$\min_{f\in\mathcal{F}} \ \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}} \ \left[\ell\big(\mathbf{f}(X), \widetilde{Y}^{\text{GLS},r}\big)\right] - (\hat{e}_1 - \hat{e}_0) \cdot (1 - r) \cdot \mathbb{E}_{(X,Y=1)\sim\mathcal{D}_{\text{clean}}} \underbrace{\left[\ell\big(\mathbf{f}(X), 0\big) - \ell\big(\mathbf{f}(X), 1\big)\right]}_{\text{confidence correction}}.$$

D. Additional Experiment Results and Details

In this section, we include more experiment results, observations and details for learning with LS/NLS.

D.1. Experiment details on CIFAR-10, CIFAR-100

We firstly introduce experiment details on CIFAR-10 dataset adopted in our experiment designs.

Training settings of clean CIFAR-10 dataset (Krizhevsky et al., 2009) We adopted ResNet34 (He et al., 2016), trained for 200 epochs with batch-size 128, SGD (Robbins & Monro, 1951) optimizer with Nesterov momentum of 0.9 and weight decay 1e-4. The learning rate of first 100 epochs is 0.1. Then it multiples with 0.1 for every 50 epochs.

Generating noise labels on CIFAR datasets We adopt symmetric noise model which generates noisy labels by randomly flipping the clean label to the other possible classes with probability ϵ. And we set $\epsilon = 0.2, 0.4, 0.6$ for CIFAR-10, $\epsilon = 0.4, 0.6$ for CIFAR-100. We also make use of asymmetric noise model. The asymmetric noise is generated by flipping the true label to the next class with probability ϵ. We set $\epsilon = 0.2, 0.3$ for CIFAR-10.

Training settings of synthetic noisy CIFAR datasets The generation of symmetric noisy dataset is adopted from (Cheng et al., 2021a). The symmetric noise rates are $[0.2, 0.4, 0.6]$. We choose two methods to train LS and NLS.

- **Direct training:** this setting is the same as training on clean CIFAR-10 dataset.
- **Warm-up:** in this case, we firstly train a ResNet34 model with Cross-Entropy loss for 120 epochs. For this warm-up, the only difference in hyper-parameter setting is the learning rate, where the initial learning rate is 0.1 and it multiplies 0.1 for every 40 epochs. After the warm-up, LS/NLS loads the same pre-trained model and trains for 100 epochs with learning rate 1e-6.

D.2. Why NLS is overlooked?

When learning from a relative large scale dataset, NLS tends to push the model become overly confident early in the training. The poor performances of NLS (direct-train) in Table 8 explain why NLS is neglected. When there is no warm-up, training NLS directly without warming up will reach a $88\% - 92\%$ test accuracy on the clean data. The performance will degrade much more significantly than LS when the noise level is high or $|r|$ is large. In Table 8, we provide the comparisons between direct-train and warm-up in several settings. The improvement bring by a warm-up procedure becomes much more significantly in the high noise regime. NLS makes the classifier be overly confident at the early training which results in converging to a bad local optimum (without CE warm-up, NLS frequently results in a worse performance in CIFAR-10 and CIFAR-100). Since the model will usually fit on the clean data first, then over-fits on the noisy ones (Liu et al., 2020), a large number of approaches (such as Loss corrections (Patrini et al., 2017), Peer Loss (Liu & Guo, 2020), etc) adopt a CE warm-up firstly. Note that there is no difference in the computing costs between NLS (with CE warmup) and CE loss, proceeding with NLS to enhance the model confidence makes NLS much more competitive in the high noise regime, also gives practical insights on how to make NLS work better when learning with clean data.

Table 8. Test accuracies of GLS on assymetric noisy CIFAR-10 and symmetric CIFAR-100 (left/right denotes direct train / warm-up).

Smooth Rate	*CIFAR-10 Asymmetric*		*CIFAR-100 Symmetric*	
	$\varepsilon = 0.2$	$\varepsilon = 0.3$	$\varepsilon = 0.4$	$\varepsilon = 0.6$
$r = 0.8$	87.89 / 90.51	86.38 / 87.97	54.78 / 51.27	40.21 / 39.80
$r = 0.6$	89.14 / 90.55	85.97 / 88.01	52.83 / 52.88	39.64 / 40.57
$r = 0.4$	88.23 / **90.61**	86.95 / 88.04	51.40 / 54.36	38.29 / 41.63
$r = -0.4$	19.71 / 89.60	21.86 / **88.42**	40.30 / 56.97	31.35 / 43.91
$r = -0.8$	- / 89.02	- / 88.28	22.63 / 57.45	26.75 / 44.19
$r = -1.0$	- / 88.68	- / 88.29	- / 57.53	- / 44.59
$r = -2.0$	- / 88.86	- / 88.13	- / 58.21	- / 45.47
$r = -4.0$	- / 89.80	- / 88.20	- / **58.47**	- / 46.86
$r = -6.0$	- / 90.02	- / 88.18	- / 57.87	- / **47.18**

D.3. Experiment details on synthetic datasets and UCI

We introduce experiment details on synthetic datasets and UCI datasets adopted in our experiment designs.

Generation of synthetic dataset In the synthetic (Type 1) dataset, we generate 500 points for both classes. Class +1 distributes inside the circle with radius 0.25. Class -1 generates by randomly sampling 500 data points in the annulus with inner radius 0.28 and outer radius 0.45. As for synthetic (Type 2) dataset, we uniformly assign labels for 50% samples in the annulus (with inner radius 0.22, outer radius 0.31) based on Type 1 dataset.

Generating noisy labels on synthetic datasets and UCI datasets Note that these datasets are all binary classification datasets, each label in the training and validation set is flipped to the other class with probability e, and we set $e = 0.1, 0.4$ for synthetic Type 1 dataset, $e = 0.1, 0.3$ for synthetic Type 2 dataset.

Training settings of synthetic datasets For both types of synthetic datasets, we adopted a three-layer ReLU Multi-Layer Perceptron (MLP), trained for 200 epochs with batch-size 128 and Adam (Kingma & Ba, 2014) optimizer. The initial learning rate is 0.1, and it multiplies 0.1 for every 40 epochs.

Training settings of UCI datasets (Dua & Graff, 2017) We adopted (Liu & Guo, 2020) a two-layer ReLU Multi-Layer Perceptron (MLP) for classification tasks on multiple UCI datasets, trained for 1000 episodes with batch-size 64 and Adam (Kingma & Ba, 2014) optimizer. We report the best performance for each smooth rate under a set of learning rate settings, $[0.0007, 0.001, 0.005, 0.01, 0.05]$.

D.4. Additional experiment on r^* and r_{opt}

r^* and r_{opt} on synthetic dataset We generate 2D (binary) synthetic dataset by randomly sampling two circularly distributed classes. The inner annulus indicates one class (blue), while the outer annulus denotes the other class (red). Clearly, the generated synthetic dataset is well-separable (Type 1) and we hold 20% data samples for performance comparison. The noise transition matrix takes a symmetric form with noise rate e_i for both classes. To simulate the scenario where the clean data may not be perfectly separated due to a non-negligible amount of uncertainty samples clustering at the decision boundary, we flip the label of 50% samples near the intersection of two annulus to the other class (Type 2). As specified in Table 9, $r^* = [0.1, 0.4]$ for Type 1 data and $r^* = [0.0, 0.2]$ for Type 2 data. With the presence of label noise, the distribution of r_{opt} shifts from non-negative ones to negative values. Even though NLS fails to outperform LS on clean data, we observe that NLS is less sensitive to noisy labels. Data with high level noise rates clearly favor NLS with a low smooth rate!

Table 9. Test accuracies of GLS on clean and noisy synthetic data. We report best test accuracy for each method. r_{opt} and the corresponding test accuracy are highlighted (green: NLS; red: CE or LS).

Method	*Synthetic data (Type 1)*			*Synthetic data (Type 2)*		
	$e_i = 0$	$e_i = 0.2$	$e_i = 0.4$	$e_i = 0$	$e_i = 0.2$	$e_i = 0.4$
LS	0.896	0.878	0.786	0.894	0.848	0.842
Vanilla Loss	0.889	0.882	0.806	0.894	0.875	0.868
NLS	0.893	0.885	0.825	0.883	0.884	0.875
$r_{\text{opt}} =$	[0.1, 0.4]	-0.2	-0.4	[0, 0.2]	-0.3	-0.5

r^* and r_{opt} on more UCI datasets We further test the performance of generalized label smoothing on 7 more UCI datasets (Heart, Breast 1, Breast 2, Diabetes, German, Image and Waveform). Our observation remains unchanged: there exists a general trend that with the increasing of noise rates, NLS becomes much more competitive than LS. Here, we attach the results of 4 additional UCI datasets for illustration.

Table 10. Test accuracy comparisons on clean and noisy UCI datasets (Image, Waveform, Heart, Banana) with best two smooth rates (green: NLS; red: CE or LS).

Smooth Rate	*Image*					*Waveform*				
	$e_i = 0$	$e_i = 0.1$	$e_i = 0.2$	$e_i = 0.3$	$e_i = 0.4$	$e_i = 0$	$e_i = 0.1$	$e_i = 0.2$	$e_i = 0.3$	$e_i = 0.4$
$r = 0.8$	0.993	0.983	0.973	0.946	0.875	0.939	0.935	0.931	0.927	0.885
$r = 0.6$	0.993	0.987	0.970	0.939	0.869	0.943	0.943	0.943	0.929	0.901
$r = 0.4$	0.997	0.980	0.973	0.939	0.865	0.941	0.937	0.943	0.931	0.905
$r = 0.2$	0.993	0.993	0.966	0.936	0.875	0.941	0.935	0.933	0.931	0.913
$r = 0.0$	0.990	0.976	0.963	0.929	0.865	0.945	0.935	0.937	0.933	0.911
$r = -0.2$	0.912	0.96	0.953	0.919	0.872	0.937	0.939	0.939	0.933	0.907
$r = -0.4$	0.882	0.923	0.953	0.936	0.872	0.925	0.937	0.939	0.933	0.917
$r = -0.8$	0.842	0.882	0.926	0.933	0.872	0.921	0.925	0.939	0.931	0.923
$r = -1.0$	0.832	0.869	0.909	0.929	0.882	0.921	0.923	0.933	0.929	0.907
$r = -2.0$	0.818	0.815	0.889	0.909	0.906	0.911	0.913	0.921	0.927	0.911
Smooth Rate	*Twonorm*					*Banana*				
	$e_i = 0$	$e_i = 0.1$	$e_i = 0.2$	$e_i = 0.3$	$e_i = 0.4$	$e_i = 0$	$e_i = 0.1$	$e_i = 0.2$	$e_i = 0.3$	$e_i = 0.4$
$r = 0.8$	0.990	0.990	0.986	0.982	0.968	0.896	0.893	0.876	0.847	0.790
$r = 0.6$	0.990	0.989	0.987	0.981	0.972	0.903	0.881	0.876	0.855	0.811
$r = 0.4$	0.990	0.990	0.987	0.983	0.971	0.900	0.887	0.874	0.859	0.807
$r = 0.2$	0.990	0.989	0.986	0.985	0.969	0.896	0.894	0.876	0.856	0.810
$r = 0.0$	0.990	0.989	0.987	0.985	0.973	0.897	0.881	0.871	0.849	0.833
$r = -0.4$	0.986	0.988	0.988	0.986	0.972	0.847	0.874	0.859	0.853	0.840
$r = -0.6$	0.986	0.988	0.987	0.984	0.974	0.845	0.864	0.861	0.859	0.837
$r = -1.0$	0.986	0.986	0.988	0.985	0.977	0.796	0.812	0.852	0.854	0.811
$r = -2.0$	0.986	0.986	0.986	0.986	0.978	0.759	0.764	0.819	0.852	0.819
$r = -4.0$	0.986	0.986	0.986	0.986	0.983	0.718	0.723	0.738	0.787	0.813
$r = -8.0$	0.986	0.986	0.986	0.985	0.986	0.703	0.700	0.699	0.735	0.735

The noisy labels are generated by a symmetric noise transition matrix with noise rate $e_i = [0.1, 0.2, 0.3, 0.4]$. As highlighted in Table 10, r_{opt} appears with positive values when the data is clean (same as r^*) or of a low noise rate. With the increasing of noise rates, NLS becomes more competitive than LS. We color-code different noise regimes where either LS (red-ish) or NLS (green-ish) outperforms the other. Clearly, there is a separation of the favored smoothing rate for different noise scenarios (upper left & low noise for LS, bottom right & high noise for NLS).

r^* and r_{opt} on AGNews We next provide an additional empirical justification of Theorem 3.1. Note that when we have access to r^*, Theorem 3.6 reveals what smooth rate recovers the performance on the clean data when learning with noisy labels. We adopt an NLP dataset AGNews (Zhang et al., 2015) for illustration. In Table 11, we do observe that r_{opt} achieves the best performance for most noise settings.

Table 11. Test accuracy comparisons on clean and symmetric noisy AGNews dataset. Highlighted numbers indicate the best performance under each ϵ.

Smooth Rate	AGNews (4 classes)				
	$\epsilon = 0$	$\epsilon = 0.1$	$\epsilon = 0.2$	$\epsilon = 0.3$	$\epsilon = 0.4$
$r = 0.4$	86.33	85.55	83.93	82.29	79.80
$r = 0.2$	87.79	86.99	85.67	83.47	81.04
$r = 0.0$	**88.20**	87.79	86.80	85.24	82.39
$r = -0.15$	85.04	**88.00**	87.47	85.83	83.09
$r = -0.2$	84.08	87.30	87.50	85.85	83.34
$r = -0.36$	81.39	84.47	**87.75**	86.14	83.62
$r = -0.4$	80.76	83.99	87.28	86.36	83.96
$r = -0.6$	77.62	80.80	84.68	**87.26**	84.37
$r = -0.67$	76.70	79.91	83.87	87.21	84.58
$r = -1.14$	72.38	74.84	78.28	82.45	**86.43**
$r = r_{\text{opt}} = \frac{(K-1)r^* - K\epsilon}{(K-1) - K\epsilon}$	**88.20**	**88.00**	**87.75**	87.21	**86.43**

D.5. Additional experiment results on model confidence

NLS improves model confidence on Synthetic Type 2 dataset In this case, the clean data that are close to decision boundary distributes randomly. In Figure 6-7, the colored bands depict the different levels of prediction probabilities. When the smooth rate increases from negative to positive, more samples fall in the orange and light blue band which indicates uncertain predictions. When the smooth rate increases from negative to positive, learning with smoothed labels will result in more uncertain predictions. With the increasing of noise rates ($e_i = 0 \to 0.4$), Learning with a fixed smooth rate generally becomes less confident on its predictions. Thus, a smaller smooth rate is required when the noise rate increases.

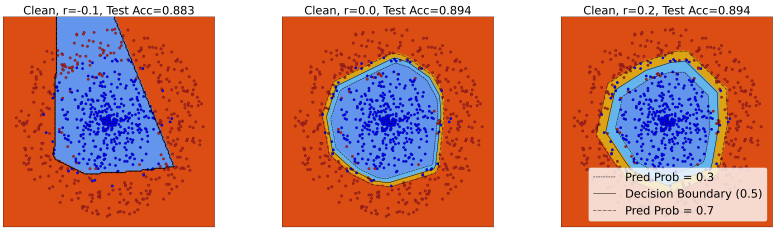

Figure 6. Model confidence visualization of NLS, VL and LS on synthetic data (Type 2) with the clean data. $r^* \in [0, 0.2]$. (left: NLS; middle: Vanilla Loss; right: LS).

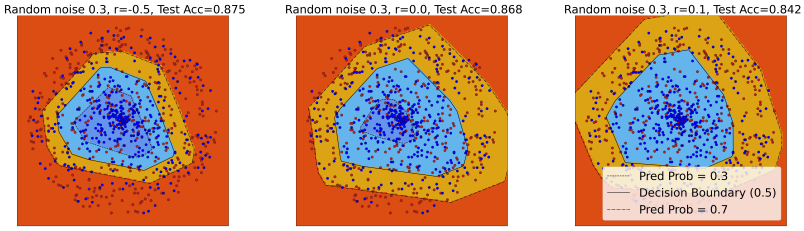

Figure 7. Model confidence visualization of NLS, VL and LS on synthetic data (Type 2) with noise rate $e_i = 0.3$. $r_{\text{opt}} = -0.5$. (left: NLS; middle: Vanilla Loss; right: LS).

D.6. Effect of LS and NLS on pre-logits

We visualise the pre-logits of a ResNet-34 for three classes on CIFAR-10. We adopt the method from (Müller et al., 2019) which illustrates how representations differ between penultimate layers of networks trained with different smooth rates in GLS. In Figure 8, NLS makes the model f be confident on her predictions and the distances between three clusters are clearly larger than those appeared in Vanilla Loss and LS.

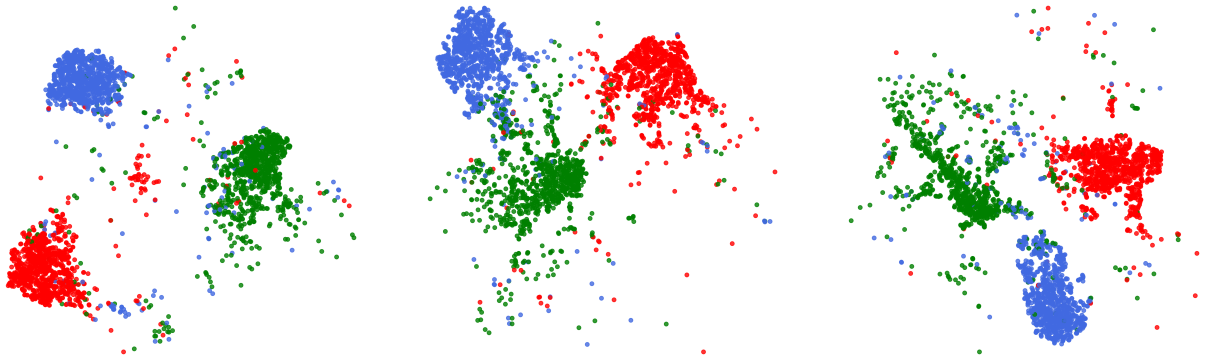

Figure 8. Effect of GLS on pre-logits (left: NLS; middle: Vanilla Loss; right: LS; trained with symmetric 0.2 noisy CIFAR-10 training dataset).

E. Bias and Variance Trade-off of Learning with Smoothed Labels

Denote $\hat{f}_{\mathrm{H}}$, $\hat{f}_{\mathrm{S}}$ as pre-trained models on the training dataset D w.r.t. hard labels and soft labels, respectively. The vector form of the prediction w.r.t. sample x given by $\hat{f}_{\mathrm{H}}$ and $\hat{f}_{\mathrm{S}}$ are $\hat{\mathbf{f}}_{\mathbf{H}}(x; D)$ and $\hat{\mathbf{f}}_{\mathbf{S}}(x; D)$. For the ease of presentation, we relate notations with subscript H/S to hard/soft labels without further explanation. Given the sample x and the one-hot label $\mathbf{y}$, we denote the averaged model prediction by:

$$\bar{\mathbf{f}}_{\mathbf{H}}(x; D) := \frac{1}{Z_{\mathrm{H}}} \exp^{\mathbb{E}_D \log(\hat{\mathbf{f}}_{\mathbf{H}}(x;D))}, \quad \bar{\mathbf{f}}_{\mathbf{S}}(x; D) := \frac{1}{Z_{\mathrm{S}}} \exp^{\mathbb{E}_D \log(\hat{\mathbf{f}}_{\mathbf{S}}(x;D))},$$

where $Z_{\mathrm{H}}, Z_{\mathrm{S}}$ are normalization constants. The bias of model prediction is defined as the KL divergence D_{KL} between target distribution (one-hot encoded vector form) $\mathbf{y}$ and the averaged model prediction.

$$\mathrm{Bias}_{\mathrm{H}} := \mathbb{E}_{x,\mathbf{y}}\left[\mathbf{y} \log \frac{\mathbf{y}}{\bar{\mathbf{f}}_{\mathbf{H}}(x; D)}\right], \quad \mathrm{Bias}_{\mathrm{S}} := \mathbb{E}_{x,\mathbf{y}}\left[\mathbf{y} \log \frac{\mathbf{y}}{\bar{\mathbf{f}}_{\mathbf{S}}(x; D)}\right].$$

While the variance of model prediction measures the expectation of KL divergence between the averaged model prediction and model prediction over D:

$$\mathrm{Var}_{\mathrm{H}} := \mathbb{E}_D\left[\mathbb{E}_{x,\mathbf{y}}\left[\bar{\mathbf{f}}_{\mathbf{H}}(x; D) \log\left(\frac{\bar{\mathbf{f}}_{\mathbf{H}}(x; D)}{\hat{\mathbf{f}}_{\mathbf{H}}(x; D)}\right)\right]\right], \quad \mathrm{Var}_{\mathrm{S}} := \mathbb{E}_D\left[\mathbb{E}_{x,\mathbf{y}}\left[\bar{\mathbf{f}}_{\mathbf{S}}(x; D) \log\left(\frac{\bar{\mathbf{f}}_{\mathbf{S}}(x; D)}{\hat{\mathbf{f}}_{\mathbf{S}}(x; D)}\right)\right]\right].$$

Empirical observation from (Zhou et al., 2021) shows that the variance brought by learning with positive soft labels given by a teacher's model (Hinton et al., 2015) is less than the direct training w.r.t hard labels. As an extension, we are interested in how LS/NLS interferes with the bias and variance of model prediction.

Bias and variance of LS/NLS on clean dataset We introduce our empirical observation regarding the role of LS/NLS in bias and variance trade-off in Figure 9. We select nine smooth rates of LS/NLS for illustration. Each smooth rate setting of LS/NLS trains on the CIFAR-10 dataset for 5 times with different data augmentations. To estimate the variance and bias of pre-trained models, we adopt the implementation in (Yang et al., 2020). Empirical results show that learning directly with a larger positive smooth rate typically results in lower variance and higher bias. In Figure 9, we can observe almost constant bias values and very low variance for NLS. This is best explained by the warm-up of pre-trained models and the fact that NLS pushes the classifier to give confident predictions. As for LS, with the increase of smooth rate, the overall bias has an increasing tendency while the variance has the decreasing pattern. Especially when the smooth rate approaches to 1, i.e., $r = 0.9$, the variance is close to 0.

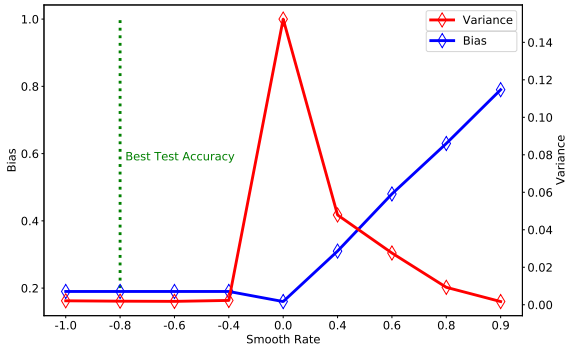

Figure 9. Bias and variance of pre-trained LS/VL/NLS models on clean CIFAR-10 test dataset.

F. Omitted Proofs

We observe that NLS connects to a special case of label smoothing regularization (Szegedy et al., 2016). We highlight this in Theorem F.1.

Theorem F.1. $\forall r \in [0, 1]$, *NLS with smooth rate* $-r$ *is a special form of label smoothing regularization:*

$$\min_{f\in\mathcal{F}} \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\ell(\mathbf{f}(X), \widetilde{Y}^{GLS,-r})\Big] = \min_{f\in\mathcal{F}} \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[2\cdot\ell(\mathbf{f}(X), \widetilde{Y}) - \ell(\mathbf{f}(X), \widetilde{Y}^{GLS,r})\Big].$$

F.1. Proof of Theorem F.1

Before we prove Theorem F.1, we first introduce Lemma F.2.

Lemma F.2. $\forall (x, \mathbf{y}^{GLS,r})$, $\ell\big(\mathbf{f}(x), \mathbf{y}^{GLS,r}\big) = \big(1-\frac{r}{2}\big)\cdot\ell\big(\mathbf{f}(x), y\big) + \frac{r}{2}\cdot\ell\big(\mathbf{f}(x), 1-y\big)$.

Proof of Lemma F.2

Proof. For CE loss, due to its linear property w.r.t. the label, we directly have:

$$\ell(\mathbf{f}(x), \mathbf{y}^{\text{GLS},r}) = \ell\big(\mathbf{f}(x), (1-r)\cdot\mathbf{y} + \frac{r}{2}\cdot\mathbf{1}\big) = \big(1-\frac{r}{2}\big)\cdot\ell\big(\mathbf{f}(x), y\big) + \frac{r}{2}\cdot\ell\big(\mathbf{f}(x), 1-y\big).$$

□

Proof of Theorem F.1

Proof. Based on Lemma F.2, with a bit of math, for NLS, we have:

$$\begin{aligned}
&\min_{f\in\mathcal{F}} \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\ell\big(\mathbf{f}(X), \widetilde{Y}^{\text{GLS},-r}\big)\Big] \\
=&\min_{f\in\mathcal{F}} \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\big(1+\frac{r}{2}\big)\cdot\ell\big(\mathbf{f}(X), \widetilde{Y}\big) - \frac{r}{2}\cdot\ell\big(\mathbf{f}(X), 1-\widetilde{Y}\big)\Big] \\
=&\min_{f\in\mathcal{F}} \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\big[\big(1+\frac{r}{2}\big) + \big(1-\frac{r}{2}\big)\big]\cdot\ell\big(\mathbf{f}(X), \widetilde{Y}\big) - \big[\big(1-\frac{r}{2}\big)\cdot\ell\big(\mathbf{f}(X), \widetilde{Y}\big) + \frac{r}{2}\cdot\ell\big(\mathbf{f}(X), 1-\widetilde{Y}\big)\big]\Big] \\
=&\min_{f\in\mathcal{F}} \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[2\cdot\ell\big(\mathbf{f}(X), \widetilde{Y}\big) - \ell\big(\mathbf{f}(X), \widetilde{Y}^{\text{GLS},r}\big)\Big].
\end{aligned}$$

□

F.2. Proof of Theorem 3.2

Proof.

$$
\begin{aligned}
Eqn.3 =& \min_{f\in\mathcal{F}} \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\underbrace{\big(1-\frac{r}{2}\big)}_{:=c_1}\cdot\ell\big(\mathbf{f}(X),\widetilde{Y}\big)+\underbrace{\frac{r}{2}}_{:=c_2}\cdot\ell\big(\mathbf{f}(X),1-\widetilde{Y}\big)\Big]\\
=& \min_{f\in\mathcal{F}} \mathbb{E}_{X,Y=0}\Big[\mathbb{P}(\widetilde{Y}=0|Y=0)\cdot\Big(c_1\cdot\ell\big(\mathbf{f}(X),0\big)+c_2\cdot\ell\big(\mathbf{f}(X),1\big)\Big)\\
&+\mathbb{P}(\widetilde{Y}=1|Y=0)\cdot\Big(c_1\cdot\ell\big(\mathbf{f}(X),1\big)+c_2\cdot\ell\big(\mathbf{f}(X),0\big)\Big)\Big]\\
&+\mathbb{E}_{X,Y=1}\Big[\mathbb{P}(\widetilde{Y}=0|Y=1)\cdot\Big(c_1\cdot\ell\big(\mathbf{f}(X),0\big)+c_2\cdot\ell\big(\mathbf{f}(X),1\big)\Big)\\
&+\mathbb{P}(\widetilde{Y}=1|Y=1)\cdot\Big(c_1\cdot\ell\big(\mathbf{f}(X),1\big)+c_2\cdot\ell\big(\mathbf{f}(X),0\big)\Big)\Big]\\
=& \min_{f\in\mathcal{F}} \mathbb{E}_{X,Y=0}\Big[\big[(1-e_0)\cdot c_1+e_0\cdot c_2\big]\cdot\ell\big(\mathbf{f}(X),0\big)+\big[(1-e_0)\cdot c_2+e_0\cdot c_1\big]\cdot\ell\big(\mathbf{f}(X),1\big)\Big]\\
&+\mathbb{E}_{X,Y=1}\Big[\big[(1-e_1)\cdot c_1+e_1\cdot c_2\big]\cdot\ell\big(\mathbf{f}(X),1\big)+\big[(1-e_1)\cdot c_2+e_1\cdot c_1\big]\cdot\ell\big(\mathbf{f}(X),0\big)\Big]\\
=& \min_{f\in\mathcal{F}} \mathbb{E}_{X,Y=0}\Big[\big[(1-e_0)\cdot c_1+e_0\cdot c_2\big]\cdot\ell\big(\mathbf{f}(X),0\big)+\big[(1-e_0)\cdot c_2+e_0\cdot c_1\big]\cdot\ell\big(\mathbf{f}(X),1\big)\Big]\\
&+\mathbb{E}_{X,Y=1}\Big[\big[(1-e_0)\cdot c_1+e_0\cdot c_2\big]\cdot\ell\big(\mathbf{f}(X),1\big)+\big[(1-e_0)\cdot c_2+e_0\cdot c_1\big]\cdot\ell\big(\mathbf{f}(X),0\big)\Big]\\
&+\mathbb{E}_{X,Y=1}\Big[e_\Delta\cdot(c_2-c_1)\cdot\ell\big(\mathbf{f}(X),1\big)-e_\Delta\cdot(c_2-c_1)\cdot\ell\big(\mathbf{f}(X),0\big)\Big]\\
=& \min_{f\in\mathcal{F}} \mathbb{E}_{(X,Y)\sim\mathcal{D}}\Big[\big[(1-e_0)\cdot c_1+e_0\cdot c_2\big]\cdot\ell\big(\mathbf{f}(X),Y\big)+\big[(1-e_0)\cdot c_2+e_0\cdot c_1\big]\cdot\ell\big(\mathbf{f}(X),1-Y\big)\Big]\\
&-e_\Delta\cdot(c_1-c_2)\cdot\mathbb{E}_{X,Y=1}\Big[\ell\big(\mathbf{f}(X),1\big)-\ell\big(\mathbf{f}(X),0\big)\Big]\\
=& \min_{f\in\mathcal{F}} \mathbb{E}_{(X,Y)\sim\mathcal{D}}\Big[(c_1+c_2)\cdot\ell\big(\mathbf{f}(X),Y\big)\Big]\\
&+\big[(1-e_0)\cdot c_2+e_0\cdot c_1\big]\cdot\mathbb{E}_{(X,Y)\sim\mathcal{D}}\Big[\ell\big(\mathbf{f}(X),1-Y\big)-\ell\big(\mathbf{f}(X),Y\big)\Big]\\
&-e_\Delta\cdot(c_1-c_2)\cdot\mathbb{E}_{X,Y=1}\Big[\ell\big(\mathbf{f}(X),1\big)-\ell\big(\mathbf{f}(X),0\big)\Big]\\
=& \min_{f\in\mathcal{F}} \mathbb{E}_{(X,Y)\sim\mathcal{D}}\Big[(c_1+c_2)\cdot\ell\big(\mathbf{f}(X),Y^*\big)\Big]\\
&+\big[-\frac{r^*}{2}+(1-e_0)\cdot c_2+e_0\cdot c_1\big]\cdot\mathbb{E}_{(X,Y)\sim\mathcal{D}}\Big[\ell\big(\mathbf{f}(X),1-Y\big)-\ell\big(\mathbf{f}(X),Y\big)\Big]\\
&-e_\Delta\cdot(c_1-c_2)\cdot\mathbb{E}_{X,Y=1}\Big[\ell\big(\mathbf{f}(X),1\big)-\ell\big(\mathbf{f}(X),0\big)\Big]\\
=& \min_{f\in\mathcal{F}} \underbrace{\mathbb{E}_{(X,Y)\sim\mathcal{D}}\Big[\ell\big(\mathbf{f}(X),Y^*\big)\Big]}_{\text{True Risk}}\underbrace{+\lambda_1\cdot\mathbb{E}_{(X,Y)\sim\mathcal{D}}\Big[\ell\big(\mathbf{f}(X),1-Y\big)-\ell\big(\mathbf{f}(X),Y\big)\Big]}_{\text{M-Inc1}}\\
&\underbrace{+\lambda_2\cdot\mathbb{E}_{X,Y=1}\Big[\ell\big(\mathbf{f}(X),0\big)-\ell\big(\mathbf{f}(X),1\big)\Big]}_{\text{M-Inc2}}.
\end{aligned}
$$

□

F.3. Proof of Theorem 3.4

Proof. With the Rademacher bound on the maximal deviation between risks and empirical ones, for $\forall f\in\mathcal{F}$ and with probability at least $1-\delta$, we have:

$$
\max_{f\in\mathcal{F}}|R^r_{\text{emp}}(f)-R^r_{\text{exp}}(f)|\leq 2\mathfrak{R}(\ell^{\text{GLS},r}\circ\mathcal{F})+\Big(\overline{\ell^{\text{GLS},r}}-\underline{\ell^{\text{GLS},r}}\Big)\cdot\sqrt{\frac{\log(1/\delta)}{2N}},
$$

where we define $\ell^{\text{GLS},r}\big(\mathbf{f}(x), \mathbf{y}^{\text{GLS},r}\big) = \big(1-\frac{r}{2}\big)\cdot\ell\big(\mathbf{f}(x), y\big)+\frac{r}{2}\cdot\ell\big(\mathbf{f}(x), 1-y\big)$, and $\mathfrak{R}$ indicates the Rademacher complexity. If ℓ is L-Lipshitz for every y, then for any $\mathbf{f}_1(x), \mathbf{f}_2(x)$, we have: $|\ell(\mathbf{f}_1(x), y) - \ell(\mathbf{f}_2(x), y)| \leq L|[\mathbf{f}_1(x)]_y - [\mathbf{f}_2(x)]_y|$. $\ell^{\text{GLS},r}$ is also L^r-Lipshitz such that for CE loss, we have:

$$
\begin{aligned}
&\Big|\big(1-\frac{r}{2}\big)\cdot\ell\big(\mathbf{f}_1(x), y\big)+\frac{r}{2}\cdot\ell\big(\mathbf{f}_1(x), 1-y\big)-\big(1-\frac{r}{2}\big)\cdot\ell\big(\mathbf{f}_2(x), y\big)-\frac{r}{2}\cdot\ell\big(\mathbf{f}_2(x), 1-y\big)\Big| \\
=&\Big|\big(1-\frac{r}{2}\big)\cdot\ell\big(\mathbf{f}_1(x), y\big)+\frac{r}{2}\cdot\ell\big(\mathbf{1}-\mathbf{f}_1(x), y\big)-\big(1-\frac{r}{2}\big)\cdot\ell\big(\mathbf{f}_2(x), y\big)-\frac{r}{2}\cdot\ell\big(\mathbf{1}-\mathbf{f}_2(x), y\big)\Big| \\
=&\Big|\big(1-\frac{r}{2}\big)\cdot\big(\ell\big(\mathbf{f}_1(x), y\big)-\ell\big(\mathbf{f}_2(x), y\big)\big)+\frac{r}{2}\cdot\big(\ell\big(\mathbf{1}-\mathbf{f}_1(x), y\big)-\ell\big(\mathbf{1}-\mathbf{f}_2(x), y\big)\big)\Big| \\
\leq&\Big|\big(1-\frac{r}{2}\big)\cdot\big(\ell\big(\mathbf{f}_1(x), y\big)-\ell\big(\mathbf{f}_2(x), y\big)\big)\Big|+\Big|\frac{r}{2}\cdot\big(\ell\big(\mathbf{1}-\mathbf{f}_1(x), y\big)-\ell\big(\mathbf{1}-\mathbf{f}_2(x), y\big)\big)\Big| \\
\leq&\underbrace{(1+\frac{|r|-r}{2})L}_{\text{defined as } L^r}|\mathbf{f}_1(x)-\mathbf{f}_2(x)|.
\end{aligned}
$$

Note that $(1-\frac{r}{2}) \geq \frac{r}{2}$ so we need to concentrate on the term $\ell(\mathbf{f}(x), y)$, what is more, $\underline{\ell}(\mathbf{f}(x), y) = \overline{\ell}(\mathbf{f}(x), 1-y)$. We then have:

$$
\overline{\ell^{\text{GLS},r}} = (1-\frac{r}{2})\cdot\overline{\ell}+\frac{r}{2}\cdot\underline{\ell};\quad \underline{\ell^{\text{GLS},r}} = (1-\frac{r}{2})\cdot\underline{\ell}+\frac{r}{2}\cdot\overline{\ell};
$$

Thus we have:

$$
\begin{aligned}
\overline{\ell^{\text{GLS},r}}-\underline{\ell^{\text{GLS},r}} &= (1-\frac{r}{2})\cdot\overline{\ell}+\frac{r}{2}\cdot\underline{\ell}-(1-\frac{r}{2})\cdot\underline{\ell}-\frac{r}{2}\cdot\overline{\ell} \\
&= (1-\frac{r}{2})\cdot\big(\overline{\ell}-\underline{\ell}\big)-\frac{r}{2}\cdot\big(\overline{\ell}-\underline{\ell}\big) \\
&= (1-r)\cdot\big(\overline{\ell}-\underline{\ell}\big).
\end{aligned}
$$

Thus, we finally have:

$$
\max_{f\in\mathcal{F}}|R^r_{\text{emp}}(f)-R^r_{\text{exp}}(f)| \leq (2+|r|-r)\cdot L\cdot\mathfrak{R}(\mathcal{F})+(1-r)\cdot\big(\overline{\ell}-\underline{\ell}\big)\cdot\sqrt{\frac{\log(1/\delta)}{2N}}.
$$

□

F.4. Proof of Proposition 5.1

Proof. The risk minimization of backward correction is equivalent to:

$$
\mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\ell^{\leftarrow}\big(\mathbf{f}(X),\widetilde{Y}\big)\Big] = \mathbb{E}_{(X,Y)\sim\mathcal{D}}\Big[\ell\big(\mathbf{f}(X),Y\big)\Big]. \quad \text{(By Theorem 1 in (Patrini et al., 2017))}
$$

The risk minimization of forward correction is equivalent to:

$$
\mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\ell^{\rightarrow}\big(\mathbf{f}(X),\widetilde{Y}\big)\Big] = \mathbb{E}_{(X,Y)\sim\mathcal{D}}\Big[\ell\big(\mathbf{f}(X),Y\big)\Big]. \quad \text{(By Theorem 2 in (Patrini et al., 2017))}
$$

Theorem 1 and 2 in (Patrini et al., 2017) demonstrate that forward and backward corrected losses equal the original loss ℓ computed on the clean data in expectation. Thus, for $r_{\text{LC}} = \frac{2e_0}{2e_0-1}$, by Theorem 3.2 (adopt $r^* = 0$), we have:

$$
\begin{aligned}
&\min_{f\in\mathcal{F}}\ \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\quad\Big[\ell\big(\mathbf{f}(X),\widetilde{Y}^{\text{GLS},r_{\text{LC}}}\big)\Big]+\lambda_{\text{LC}}\cdot\underbrace{\mathbb{E}_{X,Y=1}\Big[\ell\big(\mathbf{f}(X),1\big)-\ell\big(\mathbf{f}(X),0\big)\Big]}_{\text{Bias-LC}} \\
=&\min_{f\in\mathcal{F}}\ \mathbb{E}_{(X,Y)\sim\mathcal{D}}\Big[\ell\big(\mathbf{f}(X),Y\big)\Big]+\Big[e_0+(1-2e_0)\cdot\frac{r_{\text{LC}}}{2}\Big]\cdot\mathbb{E}_{(X,Y)\sim\mathcal{D}}\Big[\ell\big(\mathbf{f}(X),1-Y\big)-\ell\big(\mathbf{f}(X),Y\big)\Big] \\
&+e_\Delta\cdot(1-r_{\text{LC}})\cdot\mathbb{E}_{X,Y=1}\Big[\ell\big(\mathbf{f}(X),0\big)-\ell\big(\mathbf{f}(X),1\big)\Big]+\lambda_{\text{LC}}\cdot\mathbb{E}_{X,Y=1}\Big[\ell\big(\mathbf{f}(X),1\big)-\ell\big(\mathbf{f}(X),0\big)\Big] \\
=&\min_{f\in\mathcal{F}}\ \mathbb{E}_{(X,Y)\sim\mathcal{D}}\Big[\ell\big(\mathbf{f}(X),Y\big)\Big]+e_\Delta\cdot\Big(\frac{1}{1-2e_0}-\frac{1}{1-2e_0}\Big)\cdot\mathbb{E}_{X,Y=1}\Big[\ell\big(\mathbf{f}(X),0\big)-\ell\big(\mathbf{f}(X),1\big)\Big] \\
=&\min_{f\in\mathcal{F}}\ \mathbb{E}_{(X,Y)\sim\mathcal{D}}\Big[\ell\big(\mathbf{f}(X),Y\big)\Big].
\end{aligned}
$$

Thus,

$$\begin{aligned}&\min_{f\in\mathcal{F}}\ \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\ell^{\leftarrow}\big(\mathbf{f}(X),\widetilde{Y}\big)\Big]=\min_{f\in\mathcal{F}}\ \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\ell^{\rightarrow}\big(\mathbf{f}(X),\widetilde{Y}\big)\Big]\\=&\min_{f\in\mathcal{F}}\ \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\ell\big(\mathbf{f}(X),\widetilde{Y}^{\mathrm{GLS},r_{\mathrm{LC}}}\big)\Big]+\lambda_{\mathrm{LC}}\cdot\underbrace{\mathbb{E}_{X,Y=1}\Big[\ell\big(\mathbf{f}(X),1\big)-\ell\big(\mathbf{f}(X),0\big)\Big]}_{\text{Bias-LC}}.\end{aligned}$$

□

F.5. Proof of Theorem 5.2

Proof. Based on Proposition 5.1, when $e_\Delta=0$, $\lambda_{\mathrm{LC}}=0$, we directly have:

$$\begin{aligned}&\min_{f\in\mathcal{F}}\ \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\ell^{\leftarrow}\big(\mathbf{f}(X),\widetilde{Y}\big)\Big]=\min_{f\in\mathcal{F}}\ \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\ell^{\rightarrow}\big(\mathbf{f}(X),\widetilde{Y}\big)\Big]\\=&\min_{f\in\mathcal{F}}\ \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\ell\big(\mathbf{f}(X),\widetilde{Y}^{\mathrm{GLS},r_{\mathrm{LC}}}\big)\Big].\end{aligned}$$

□

F.6. Proof of Theorem 5.3

Proof. Note that

$$\min_{f\in\mathcal{F}}\ \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\ell_{\mathrm{CL}}\big(\mathbf{f}(X),\widetilde{Y}\big)\Big]=\min_{f\in\mathcal{F}}\ \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\ell\big(\mathbf{f}(X),\widetilde{Y}\big)-\ell\big(\mathbf{f}(X),1-\widetilde{Y}\big)\Big].$$

We have:

$$\begin{aligned}&\min_{f\in\mathcal{F}}\ \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\ell\big(\mathbf{f}(X),\widetilde{Y}^{\mathrm{GLS},r_{\mathrm{CL}}}\big)\Big]\\=&\min_{f\in\mathcal{F}}\ \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\big(1-\frac{r_{\mathrm{CL}}}{2}\big)\cdot\ell\big(\mathbf{f}(X),\widetilde{Y}\big)+\frac{r_{\mathrm{CL}}}{2}\cdot\ell\big(\mathbf{f}(X),1-\widetilde{Y}\big)\Big]\\\Leftrightarrow&\min_{f\in\mathcal{F}}\ \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\ell\big(\mathbf{f}(X),\widetilde{Y}\big)+\frac{r_{\mathrm{CL}}}{2-r_{\mathrm{CL}}}\cdot\ell\big(\mathbf{f}(X),1-\widetilde{Y}\big)\Big].\end{aligned}$$

When $r_{\mathrm{CL}}\to-\infty$, we have $\frac{r_{\mathrm{CL}}}{2-r_{\mathrm{CL}}}\to-1$. Thus,

$$\min_{f\in\mathcal{F}}\ \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\ell_{\mathrm{CL}}\big(\mathbf{f}(X),\widetilde{Y}\big)\Big]=\min_{f\in\mathcal{F}}\ \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\ell\big(\mathbf{f}(X),\widetilde{Y}^{\mathrm{GLS},r_{\mathrm{CL}}\to-\infty}\big)\Big].$$

□

F.7. Proof of Proposition 5.4

Proof. Note that:

$$\begin{aligned}&\mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\ell\big(\mathbf{f}(X),\widetilde{Y}\big)\Big]-\mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\ell\big(\mathbf{f}(X),\widetilde{Y}^{\mathrm{GLS},r}\big)\Big]\\=&\mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[1-\big(1-\frac{r}{2}\big)\cdot\ell\big(\mathbf{f}(X),\widetilde{Y}\big)-\frac{r}{2}\cdot\ell\big(\mathbf{f}(X),1-\widetilde{Y}\big)\Big]\\=&\frac{r}{2}\cdot\mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\ell\big(\mathbf{f}(X),\widetilde{Y}\big)-\ell\big(\mathbf{f}(X),1-\widetilde{Y}\big)\Big].\end{aligned}$$

And we have:

$$
\begin{aligned}
&\mathbb{E}_{(X_i,\widetilde{Y}_i)\sim\widetilde{\mathcal{D}}}\Big[\ell\big(\mathbf{f}(X_1),\widetilde{Y}_2\big)\Big]\\
=&\mathbb{E}_X\Big[\mathbb{P}(\widetilde{Y}=0)\cdot\ell\big(\mathbf{f}(X),0\big)+\big(1-\mathbb{P}(\widetilde{Y}=0)\big)\cdot\ell\big(\mathbf{f}(X),1\big)\Big]\\
=&\mathbb{E}_{X,\widetilde{Y}=0}\Big[\mathbb{P}(\widetilde{Y}=0)\cdot\ell\big(\mathbf{f}(X),0\big)+\big(1-\mathbb{P}(\widetilde{Y}=0)\big)\cdot\ell\big(\mathbf{f}(X),1\big)\Big]\\
&+\mathbb{E}_{X,\widetilde{Y}=1}\Big[\mathbb{P}(\widetilde{Y}=0)\cdot\ell\big(\mathbf{f}(X),0\big)+\big(1-\mathbb{P}(\widetilde{Y}=0)\big)\cdot\ell\big(\mathbf{f}(X),1\big)\Big]\\
=&\mathbb{E}_{X,\widetilde{Y}=0}\Big[\mathbb{P}(\widetilde{Y}=0)\cdot\ell\big(\mathbf{f}(X),0\big)+\big(1-\mathbb{P}(\widetilde{Y}=0)\big)\cdot\ell\big(\mathbf{f}(X),1\big)\Big]\\
&+\mathbb{E}_{X,\widetilde{Y}=1}\Big[\big(1-\mathbb{P}(\widetilde{Y}=0)\big)\cdot\ell\big(\mathbf{f}(X),0\big)+\mathbb{P}(\widetilde{Y}=0)\cdot\ell\big(\mathbf{f}(X),1\big)\Big]\\
&+\big(1-2\cdot\mathbb{P}(\widetilde{Y}=0)\big)\cdot\mathbb{E}_{X,\widetilde{Y}=1}\Big[\ell\big(\mathbf{f}(X),1\big)-\ell\big(\mathbf{f}(X),0\big)\Big].
\end{aligned}
$$

Thus,

$$
\begin{aligned}
&\min_{f\in\mathcal{F}}\ \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\ell_{\text{PL}}\big(\mathbf{f}(X),\widetilde{Y}\big)\Big]=\min_{f\in\mathcal{F}}\ \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\ell\big(\mathbf{f}(X),\widetilde{Y}\big)-\ell\big(\mathbf{f}(X_1),\widetilde{Y}_2\big)\Big]\\
=&\min_{f\in\mathcal{F}}\ \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\ell\big(\mathbf{f}(X),\widetilde{Y}\big)\Big]-\mathbb{E}_{(X_i,\widetilde{Y}_i)\sim\widetilde{\mathcal{D}}}\Big[\ell\big(\mathbf{f}(X_1),\widetilde{Y}_2\big)\Big]\\
=&\min_{f\in\mathcal{F}}\ \mathbb{E}_{X,\widetilde{Y}=0}\Big[\ell\big(\mathbf{f}(X),0\big)\Big]+\mathbb{E}_{X,\widetilde{Y}=1}\Big[\ell\big(\mathbf{f}(X),1\big)\Big]\\
&-\mathbb{E}_{X,\widetilde{Y}=0}\Big[\mathbb{P}(\widetilde{Y}=0)\cdot\ell\big(\mathbf{f}(X),0\big)+\big(1-\mathbb{P}(\widetilde{Y}=0)\big)\cdot\ell\big(\mathbf{f}(X),1\big)\Big]\\
&-\mathbb{E}_{X,\widetilde{Y}=1}\Big[\big(1-\mathbb{P}(\widetilde{Y}=0)\big)\cdot\ell\big(\mathbf{f}(X),0\big)+\mathbb{P}(\widetilde{Y}=0)\cdot\ell\big(\mathbf{f}(X),1\big)\Big]\\
&-\big(1-2\cdot\mathbb{P}(\widetilde{Y}=0)\big)\cdot\mathbb{E}_{X,\widetilde{Y}=1}\Big[\ell\big(\mathbf{f}(X),1\big)-\ell\big(\mathbf{f}(X),0\big)\Big]\\
=&\min_{f\in\mathcal{F}}\ \mathbb{E}_{X,\widetilde{Y}=0}\Big[\big(1-\mathbb{P}(\widetilde{Y}=0)\big)\cdot\big[\ell\big(\mathbf{f}(X),0\big)-\ell\big(\mathbf{f}(X),1\big)\big]\Big]\\
&+\mathbb{E}_{X,\widetilde{Y}=1}\Big[\big(1-\mathbb{P}(\widetilde{Y}=0)\big)\cdot\big[\ell\big(\mathbf{f}(X),1\big)-\ell\big(\mathbf{f}(X),0\big)\big]\Big]\\
&-\big(1-2\cdot\mathbb{P}(\widetilde{Y}=0)\big)\cdot\mathbb{E}_{X,\widetilde{Y}=1}\Big[\ell\big(\mathbf{f}(X),1\big)-\ell\big(\mathbf{f}(X),0\big)\Big]\\
=&\min_{f\in\mathcal{F}}\ \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\big(1-\mathbb{P}(\widetilde{Y}=0)\big)\cdot\big[\ell\big(\mathbf{f}(X),\widetilde{Y}\big)-\ell\big(\mathbf{f}(X),1-\widetilde{Y}\big)\big]\Big]\\
&-\big(1-2\cdot\mathbb{P}(\widetilde{Y}=0)\big)\cdot\mathbb{E}_{X,\widetilde{Y}=1}\Big[\ell\big(\mathbf{f}(X),1\big)-\ell\big(\mathbf{f}(X),0\big)\Big].
\end{aligned}
$$

Thus, for $r_{\text{PL}}=2\cdot\mathbb{P}(\widetilde{Y}=1)$, $\lambda_{\text{PL}}=1-r_{\text{PL}}$, we have:

$$
\begin{aligned}
&\mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\ell_{\text{PL}}(\mathbf{f}(X),\widetilde{Y})\Big]-\left[\mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\ell(\mathbf{f}(X),\widetilde{Y})\Big]-\mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\ell\big(\mathbf{f}(X),\widetilde{Y}^{\text{GLS},r_{\text{PL}}}\big)\Big]\right]\\
=&\mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\big(1-\mathbb{P}(\widetilde{Y}=0)\big)\cdot\big[\ell\big(\mathbf{f}(X),\widetilde{Y}\big)-\ell\big(\mathbf{f}(X),1-\widetilde{Y}\big)\big]\Big]\\
&-\big(1-2\cdot\mathbb{P}(\widetilde{Y}=0)\big)\cdot\mathbb{E}_{X,\widetilde{Y}=1}\Big[\ell\big(\mathbf{f}(X),1\big)-\ell\big(\mathbf{f}(X),0\big)\Big]\\
&-\frac{r_{\text{PL}}}{2}\cdot\mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\ell\big(\mathbf{f}(X),\widetilde{Y}\big)-\ell\big(\mathbf{f}(X),1-\widetilde{Y}\big)\Big]\\
=&\mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\big(1-\mathbb{P}(\widetilde{Y}=0)-\mathbb{P}(\widetilde{Y}=1)\big)\cdot\big[\ell\big(\mathbf{f}(X),\widetilde{Y}\big)-\ell\big(\mathbf{f}(X),1-\widetilde{Y}\big)\big]\Big]\\
&-\big(2\cdot\mathbb{P}(\widetilde{Y}=1)-1\big)\cdot\mathbb{E}_{X,\widetilde{Y}=1}\Big[\ell\big(\mathbf{f}(X),1\big)-\ell\big(\mathbf{f}(X),0\big)\Big]\\
=&\lambda_{\text{PL}}\cdot\mathbb{E}_{X,\widetilde{Y}=1}\Big[\ell\big(\mathbf{f}(X),1\big)-\ell\big(\mathbf{f}(X),0\big)\Big].
\end{aligned}
$$

And we can conclude that:

$$\min_{f\in\mathcal{F}} \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\ell_{\text{PL}}(\mathbf{f}(X),\widetilde{Y})\Big] = \min_{f\in\mathcal{F}} \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\ell(\mathbf{f}(X),\widetilde{Y}) - \ell(\mathbf{f}(X),\widetilde{Y}^{\text{GLS},r_{\text{PL}}})\Big] + \lambda_{PL}\cdot \underbrace{\mathbb{E}_{X,\widetilde{Y}=1}\Big[\ell(\mathbf{f}(X),1)-\ell(\mathbf{f}(X),0)\Big]}_{\text{Bias-PL}}.$$

□

F.8. Proof of Theorem 5.5

Proof. When $\mathbb{P}(\widetilde{Y}=0)=\mathbb{P}(\widetilde{Y}=1)$, according to Proposition 5.4, we have $\lambda_{PL}=0$ and:

$$\begin{aligned}\min_{f\in\mathcal{F}} \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\ell_{\text{PL}}\big(\mathbf{f}(X),\widetilde{Y}\big)\Big] &= \min_{f\in\mathcal{F}} \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\ell\big(\mathbf{f}(X),\widetilde{Y}\big) - \ell\big(\mathbf{f}(X),\widetilde{Y}^{\text{GLS},r_{\text{PL}}}\big)\Big]\\ &= \min_{f\in\mathcal{F}} \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\frac{r_{\text{PL}}}{2}\cdot\ell\big(\mathbf{f}(X),\widetilde{Y}\big) - \frac{r_{\text{PL}}}{2}\cdot\ell\big(\mathbf{f}(X),1-\widetilde{Y}\big)\Big]\\ &\Leftrightarrow \min_{f\in\mathcal{F}} \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\ell\big(\mathbf{f}(X),\widetilde{Y}\big) - \ell\big(\mathbf{f}(X),1-\widetilde{Y}\big)\Big].\end{aligned}$$

When $r_{\text{PL}}\to-\infty$, we further have:

$$\begin{aligned}\min_{f\in\mathcal{F}} \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\ell\big(\mathbf{f}(X),\widetilde{Y}^{\text{GLS},r_{\text{PL}}}\big)\Big] &\Leftrightarrow \min_{f\in\mathcal{F}} \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\ell\big(\mathbf{f}(X),\widetilde{Y}\big) + \frac{r_{\text{CL}}}{2-r_{\text{CL}}}\cdot\ell\big(\mathbf{f}(X),1-\widetilde{Y}\big)\Big]\\ &\Leftrightarrow \min_{f\in\mathcal{F}} \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\ell\big(\mathbf{f}(X),\widetilde{Y}\big) - \ell\big(\mathbf{f}(X),1-\widetilde{Y}\big)\Big].\end{aligned}$$

Thus, Theorem 5.5 is proved. □

F.9. Proof of Theorem 3.3

Proof. Note that the optimal r that will cancel the impact of Term M-Inc1 is:

$$r_{\text{opt}} := \frac{r^*-2e}{1-2e}.$$

- When $e<\frac{r^*}{2}$, $r_{\text{opt}}>0$. In this case, learning LS with smooth rate r_{opt} results in:

$$\min_{f\in\mathcal{F}} \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\ell(\mathbf{f}(X),\widetilde{Y}^{\text{GLS},r=r_{\text{opt}}})\Big] = \min_{f\in\mathcal{F}} \mathbb{E}_{(X,Y)\sim\mathcal{D}}\Big[\ell\big(\mathbf{f}(X),Y^*\big)\Big],$$

  which yields $f^*_{\mathcal{D}}$;

- When $e=\frac{r^*}{2}$, $r_{\text{opt}}=0$. Learning with the Vanilla Loss yields $f^*_{\mathcal{D}}$ since:

$$\min_{f\in\mathcal{F}} \mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\ell(\mathbf{f}(X),\widetilde{Y})\Big] = \min_{f\in\mathcal{F}} \mathbb{E}_{(X,Y)\sim\mathcal{D}}\Big[\ell\big(\mathbf{f}(X),Y^*\big)\Big];$$

- Similarly, when $e>\frac{r^*}{2}$, learning NLS with $r=r_{\text{opt}}<0$ yields $f^*_{\mathcal{D}}$.

□

F.10. Proof of Theorem 3.6

Proof. Denote $p_i = \mathbb{P}(Y=i)$ as the clean label distribution, $\tilde{p}_i = \mathbb{P}(\widetilde{Y}=i)$ as the clean label distribution. Let $\epsilon' = \frac{K\cdot\epsilon}{K-1}$, we have:

$$
\begin{aligned}
&\mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[(1-r)\cdot\ell\big(\mathbf{f}(X),\widetilde{Y}\big)\Big] + \mathbb{E}_X\Big[\sum_{i\in[K]}\frac{r}{K}\cdot\ell\big(\mathbf{f}(X),i\big)\Big] \\
=&\Bigg[\sum_{i\in[K]}\mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}},Y=i}\Big[(1-r)\cdot\ell\big(\mathbf{f}(X),\widetilde{Y}\big)\Big]\Bigg] + \mathbb{E}_X\Big[\sum_{i\in[K]}\frac{r}{K}\cdot\ell\big(\mathbf{f}(X),i\big)\Big] \\
=&\Bigg[(1-r)\cdot\sum_{i\in[K]}\mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}},Y=i}\Big[\sum_{j\in[K]}T_{i,j}\cdot\ell\big(\mathbf{f}(X),\widetilde{Y}=j\big)\Big]\Bigg] + \mathbb{E}_X\Big[\sum_{i\in[K]}\frac{r}{K}\cdot\ell\big(\mathbf{f}(X),i\big)\Big] \\
=&\Bigg[(1-r)\cdot\sum_{i\in[K]}\mathbb{E}_{X,Y=i}\Big[(1-\epsilon')\cdot\ell\big(\mathbf{f}(X),i\big)+\sum_{j\in[K]}\frac{\epsilon'}{K}\cdot\ell\big(\mathbf{f}(X),j\big)\Big]\Bigg] + \mathbb{E}_X\Bigg[\sum_{i\in[K]}\frac{r}{K}\cdot\ell\big(\mathbf{f}(X),i\big)\Bigg] \\
=&\Bigg[(1-r)\cdot\sum_{i\in[K]}\mathbb{E}_{X,Y=i}\Big[\big(1-\epsilon'\big)\cdot\ell\big(\mathbf{f}(X),i\big)\Big]\Bigg] + \mathbb{E}_X\Bigg[\Big[\frac{(1-r)\cdot\epsilon'}{K}+\frac{r}{K}\Big]\sum_{j\in[K]}\ell\big(\mathbf{f}(X),j\big)\Bigg] \\
=&\Bigg[\underbrace{(1-r)\cdot\big(1-\epsilon'\big)}_{:=c_3}\mathbb{E}_{(X,Y)\sim\mathcal{D}}\Big[\ell\big(\mathbf{f}(X),Y\big)\Big]\Bigg] + \mathbb{E}_X\Bigg[\underbrace{\Big[\frac{(1-r)\cdot\epsilon'}{K}+\frac{r}{K}\Big]}_{:=c_4}\sum_{j\in[K]}\ell\big(\mathbf{f}(X),j\big)\Bigg] \\
=&\Bigg[\frac{c_3}{1-r^*}\cdot\mathbb{E}_{(X,Y)\sim\mathcal{D}}\Big[\ell\big(\mathbf{f}(X),Y^*\big)-\frac{r^*}{K}\sum_{j\in[K]}\ell\big(\mathbf{f}(X),j\big)\Big]\Bigg] + \Bigg[c_4\cdot\mathbb{E}_X\Big[\sum_{j\in[K]}\ell\big(\mathbf{f}(X),j\big)\Big]\Bigg] \\
=&\underbrace{\Bigg[\frac{c_3}{1-r^*}\cdot\mathbb{E}_{(X,Y)\sim\mathcal{D}}\Big[\ell\big(\mathbf{f}(X),Y^*\big)\Big]\Bigg]}_{\text{True Risk}} + \underbrace{\Bigg[\Big(c_4-\frac{c_3\cdot r^*}{(1-r^*)\cdot K}\Big)\cdot\mathbb{E}_X\Big[\sum_{j\in[K]}\ell\big(\mathbf{f}(X),j\big)\Big]\Bigg]}_{\text{M-Inc1}}.
\end{aligned}
$$

Adopting $r_{\text{opt}} = \frac{r^*-\epsilon'}{1-\epsilon'}$, with a bit of math, the weight of Term M-Inc1 becomes 0 and

$$
\begin{aligned}
&\mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[\ell\big(\mathbf{f}(X),Y^{\text{GLS},r_{\text{opt}}}\big)\Big] \\
=&\mathbb{E}_{(X,\widetilde{Y})\sim\widetilde{\mathcal{D}}}\Big[(1-r_{\text{opt}})\cdot\ell\big(\mathbf{f}(X),\widetilde{Y}\big)\Big] + \mathbb{E}_X\Big[\sum_{i\in[K]}\frac{r_{\text{opt}}}{K}\cdot\ell\big(\mathbf{f}(X),i\big)\Big] \\
=&\Bigg[\frac{c_3}{1-r^*}\cdot\mathbb{E}_{(X,Y)\sim\mathcal{D}}\Big[\ell\big(\mathbf{f}(X),Y^*\big)\Big]\Bigg] \\
\Leftrightarrow&\mathbb{E}_{(X,Y)\sim\mathcal{D}}\Big[\ell\big(\mathbf{f}(X),Y^*\big)\Big].
\end{aligned}
$$

□