Understanding Instance-Level Label Noise: Disparate Impacts and Treatments

Yang Liu [1]

Abstract

This paper aims to provide understandings for the effect of an over-parameterized model, e.g. a deep neural network, memorizing instance-dependent noisy labels. We first quantify the harms caused by memorizing noisy instances, and show the *disparate impacts* of noisy labels for sample instances with different representation frequencies. We then analyze how several popular solutions for learning with noisy labels mitigate this harm at the instance level. Our analysis reveals that existing approaches lead to *disparate treatments* when handling noisy instances. While higher-frequency instances often enjoy a high probability of an improvement by applying these solutions, lower-frequency instances do not. Our analysis reveals new understandings for when these approaches work, and provides theoretical justifications for previously reported empirical observations. This observation requires us to rethink the distribution of label noise across instances and calls for different treatments for instances in different regimes.

1. Introduction

A salient feature of an over-parameterized model, e.g. a deep neural network, is its ability to memorize examples (Zhang et al., 2016; Neyshabur et al., 2017), and the memorization has proven to benefit the generalization performance (Arpit et al., 2017; Feldman, 2020; Feldman & Zhang, 2020). Nonetheless, the potential existence of label noise, combined with the memorization effect, might lead to detrimental consequence (Song et al., 2020; Yao et al., 2020a; Cheng et al., 2020b; Chen et al., 2019; Han et al., 2020; Song et al., 2020). In light of the reported empirical evidence of harms caused by over-memorizing noisy labels, we set out to understand this effect theoretically. Built on a recent analytical framework (Feldman, 2020), we demonstrate the varying effects of memorizing noisy labels associated with instances that sit at the different spectra of the instance distribution.

Soon since the above negative effect was empirically shown, learning with noisy labels has been recognized as a challenging and important task. The literature has observed growing interests in proposing defenses, see Natarajan et al. (2013); Liu & Tao (2016); Menon et al. (2015); Liu & Guo (2020); Lukasik et al. (2020) and many more. The second contribution of this paper is to build an analytical framework to gain new understandings of how the existing solutions fare (Section 5). While most existing theoretical results focus on the setting where label noise is homogeneous across training examples and focus on the distribution-level analysis, ours invests on the instance-level and aims to quantify when these existing approaches work and when they fail for different regimes of instances. Our result points out that while noisy labels for highly frequent instances contribute more to the drop of generalization power, they are also easier cases to fix with. We further highlight the need for taking additional care of long-tail examples (Zhu et al., 2014), where we prove existing solutions can have a substantial probability of failing. Our results call for immediate attention to a hybrid treatment of noisy instances.

To facilitate the understanding of our results, we outline the main contributions below with pointers:

- We extend an analytical framework to quantify the effects of memorizing noisy labels (Theorem 4 - 6).
- We highlight the scenarios when existing popular robust learning methods succeed or fail at the instance level (Section 4.4 & 5). We provide the conditions under which the existing approaches improve over memorizing the noisy labels (Theorem 7 & 11 and their corollaries), and when not (Theorem 8 & 12 and their corollaries).
- Our results in Section 5 help explain some empirical observations reported in the literature, including i) peer loss (Liu & Guo, 2020) induces confident prediction (Lemma 3), and ii) when peer loss and label smoothing (Lukasik et al., 2020) could perform better than loss correction (Natarajan et al., 2013; Patrini et al., 2017), which uses explicit knowledge of the noise rates (Sec-

[1]Department of Computer Science and Engineering, University of California, Santa Cruz, CA, USA. Correspondence to: Yang Liu <yangliu@ucsc.edu>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021.

tion 5.2 & 5.3) - in contrast, peer loss does not require this knowledge.

Due to space limit, all proofs can be found in the Appendix.

1.1. Related works

There have been substantial discussions on the memorization effects of deep neural networks, and how memorization relates to generalization (Zhang et al., 2016; Neyshabur et al., 2017; Arpit et al., 2017; Feldman, 2020; Feldman & Zhang, 2020). Most relevant to us, recent works have reported negative consequences of memorizing noisy labels, and have proposed corresponding fixes (Natarajan et al., 2013; Liu & Tao, 2016; Liu & Guo, 2020; Song et al., 2020; Yao et al., 2020a; Cheng et al., 2020b; Chen et al., 2019; Han et al., 2020; Song et al., 2020). Different solutions seem to be effective when guarding different type of noise, but there lacks a unified framework to understand why one approach would work and when they would fail.

Recently, there is increasing attention on learning with instance-dependent noise, which proves to be a much more challenging case (Cheng et al., 2020b;a; Xia et al., 2020; Zhu et al., 2021a). Our work echoes this effort and emphasizes the instance-level understanding. This focus particularly suits a study with long-tail distributions of instances that appear with different frequencies, which is often shown to be the case with image datasets (Zhu et al., 2014).

Common solutions toward learning with noisy labels build around loss or label corrections (Natarajan et al., 2013; Patrini et al., 2017; Xia et al., 2019). More recently, light and easy-to-implement solutions are proposed too (Lukasik et al., 2020; Liu & Guo, 2020). We delve into three of them in Section 4. As an area with growing interests, there exist many other solutions - we will not have space to list all, but we want to mention the following two streams of efforts. *Sample cleaning*: Sample cleaning leverages the idea of detecting instance x whose label is corrupted ($\tilde{y} \neq y$) (Jiang et al., 2017; Han et al., 2018; Yu et al., 2019; Yao et al., 2020a; Wei et al., 2020; Cheng et al., 2020a). Then the training is mainly done with the selected clean instances, with the aid of processed information from the detected corrupted examples. *Robust loss function*: The literature has also observed the proposal of robust loss functions that perform well with dealing outlier noisy examples (Zhang & Sabuncu, 2018; Menon et al., 2019; Charoenphakdee et al., 2019; Wang et al., 2019).

1.2. Overview of the main results: Disparate impacts and treatments of label noise

Our first set of results, perhaps non-surprisingly, show the disparate impact of noisy labels at the instance level. The impact to the drop of generalization power is linearly dependent on the frequency of the instance and its labels' associated noise rate:

Theorem 1 (Disparate Impacts, Informal). *For an instance x that appears l times (l-appearance instance) in the training data (with n samples), a model h memorizing its l noisy labels leads to the following order of individual excessive generalization error:*

$$\Omega\left(\frac{l^2}{n^2}\cdot \textit{(label noise rate at } x\textit{)}\right)$$

Our discussions then move to how the existing treatments fare at the instance level. We will introduce a memorization paradox in Section 4.4 to highlight a common pitfall when analyzing the performance of existing algorithmic treatments at the population level, when a deep model is considered and is able to memorize training examples. Probably more alarmingly, we then provide a set of instance-level analysis to show the disparate treatments of several existing learning with noisy label solutions:

Theorem 2 (Disparate Treatments, Informal). *For an instance x that appears l times in the training data (l-appearance instance), **when l is large (high-frequency instance)**, with high probability, performing loss correction (Natarajan et al., 2013; Patrini et al., 2017) and using peer loss correction (Liu & Guo, 2020) on x improves generalization performance compared to memorizing the noisy labels. **When l is small (long-tail instance)**, with non-negligible probability, both loss correction and peer loss incur higher prediction errors than memorizing the noisy labels.*

2. Formulation

To reuse the main analytical framework built in Feldman (2020), we largely follow their notations. In the clean setting, a training dataset $S = \{(x_1, y_1), ..., (x_n, y_n)\}$ is available. Each x indicates a feature vector and each y is an associated label. Denote by X the space of x and Y the space for y. Jointly (x, y) are drawn from an unknown distribution $\mathcal{P}$ over $X \times Y$. Specifically, x is sampled from a distribution $\mathcal{D}$, and the true label y for x is specified by a function $f : X \to Y$ *drawn from a distribution* $\mathcal{F}$.

The learner's algorithm $\mathcal{A}$, as a function of the training data S, returns a distribution of classifiers or functions $h : X \to Y$. By this, we consider a randomized algorithm that would potentially lead to the deployment of a randomized classifier. We define the following generalization error $\mathsf{err}_{\mathcal{P}}(\mathcal{A}, S) := \mathbb{E}_{h\sim\mathcal{A}(S)}[\mathsf{err}_{\mathcal{P}}(h)]$, where $\mathsf{err}_{\mathcal{P}}(h) := \mathbb{E}_{\mathcal{P}}[\mathbb{1}(h(x) \neq y)]$ and $\mathbb{1}(\cdot)$ is the indicator function. When there is no confusion, we shall use x, y to denote the random variables generating these quantities when used in a probability measure. To better and clearly demonstrate the main message of this paper, we consider discrete domains of X and Y such that $|X| = N, |Y| = m$. Our model,

as well as the main generalization results, can mostly extend to a setting with continuous X (Section 4, (Feldman, 2020)). We briefly discuss it after we introduce the following process to capture the generation of each instance x: We follow Feldman (2020) to characterize an unstructured discrete domain of classification problems:

- Let $\pi = \{\pi_1, ..., \pi_N\}$ denote the priors for each $x \in X$.
- For each $x \in X$, sample a quantity p_x independently and uniformly from the set π.
- Then the resulting probability mass function of x is given by $D(x) = \frac{p_x}{\sum_{x \in X} p_x}$ - this forms the distribution $\mathcal{D}$ that x will be drawn from.

For the case with continuous X, instead of assuming a prior π over each x in a finite X, it is assumed there is a prior π defined over N mixture models. Each x has a certain probability of being drawn from each model and then will realize according to the generative model. Each of the generative models captures similar but non-identical examples. With the above generation process, denote by $\mathbb{P}[\cdot|S]$ the marginal distribution over $\mathcal{P}$ conditional on S, we further define the following conditional generalization error (on the realization of the training data S):

$$\mathsf{err}(\pi, \mathcal{F}, \mathcal{A}|S) := \mathbb{E}_{\mathcal{P} \sim \mathbb{P}[\cdot|S]} \left[\mathsf{err}_{\mathcal{P}}(\mathcal{A}, S)\right].$$

l-appearance instances: We denote by $X_{S=l}$ the set of xs that appeared exactly $l \geq 1$ times in the dataset S. The difference in l helps us capture the imbalance of the distribution of instances. Later we show that the handling of instances with different frequencies matters differently.

2.1. Noisy labels

We consider a setting where the training labels are noisy. Suppose for each training instance (x, y), instead of observing the true label y, we observe a noisy copy of it, denoting by $\tilde{y}$. Each $\tilde{y}$ is generated according to the following model:

$$T_{k,k'}(x) := \mathbb{P}[\tilde{y} = k'|y = k, x], k', k \in Y. \quad (1)$$

We will denote by $T(x) \in \mathbb{R}^{m \times m}$ the noise transition matrix with the (k, k')-entry defined by $T_{k,k'}(x)$. Each of the above noisy label generation is independent across different x. We have access to the above noisy dataset $\tilde{S} := \{(x_1, \tilde{y}_1), ..., (x_n, \tilde{y}_n)\}$.

An x that appears l times in the dataset will have l independently generated noisy labels. One can think of these as l similar data instances, with each of them equipped with a single noisy label collected independently. For instance, Figure 1 shows a collection of 10 similar "Cats" from the CIFAR-10 dataset (Krizhevsky et al., 2009). On top of each image, we show a "noisy" label collected from Amazon Mechanical Turk. Approximately, one can view each row as a x with 5 appearances, with each of the instances associated with a potentially corrupted label.

Figure 1. Sample examples of "Cats" in CIFAR-10 with noisy labels on top. Examples are taken from (Zhu et al., 2021b).

$T(x)$ varies across the dataset S, and possibly that $T(x)$ would even be higher for low-frequency/rare instances, due to the inherent difficulties in recognizing and labeling them.

Note the noisy label distribution $\mathbb{P}[\tilde{y}|x]$ has a larger entropy due to the additional randomness introduced by $T(x)$ and is therefore harder to fit. As we shall see later, this fact poses additional challenges, especially for the long-tail instances that have an insufficient number of observations.

3. Impacts of Memorizing Noisy Labels

In this section, we discuss the impacts of noisy labels when training a model that can memorize examples. Our analysis builds on a recent generalization bound for studying the memorization effects of an over-parameterized model.

3.1. Generalization error

Denote by $\bar{\pi}^N$ the resulting marginal distribution over x: $\bar{\pi}^N(\alpha) := \mathbb{P}[D(x) = \alpha]$. $\bar{\pi}^N$ controls the true frequency of generating instances. Define the following quantity:

$$\tau_l := \frac{\mathbb{E}_{\alpha \sim \bar{\pi}^N}[\alpha^{l+1} \cdot (1-\alpha)^{n-l}]}{\mathbb{E}_{\alpha \sim \bar{\pi}^N}[\alpha^{l} \cdot (1-\alpha)^{n-l}]}. \quad (2)$$

Intuitively, τ_l quantifies the "importance weight" of the l-appearance instances. Theorem 2.3 of Feldman (2020) provides the following generalization error of an algorithm $\mathcal{A}$:

Theorem 3 (Theorem 2.3, (Feldman, 2020))**.** *For every learning algorithm $\mathcal{A}$ and every dataset $S \in (X \times Y)^n$:*

$$\mathsf{err}(\pi, \mathcal{F}, \mathcal{A}|S) \geq opt(\pi, \mathcal{F}|S) + \sum_{l \in [n]} \tau_l \cdot \sum_{x \in X_{S=l}} \mathbb{P}_{h \sim \mathcal{A}}[h(x) \neq y], \quad (3)$$

where in above, $opt(\pi, \mathcal{F}|S) := \min_{\mathcal{A}} \mathsf{err}(\pi, \mathcal{F}, \mathcal{A}|S)$ is the minimum achievable generalization error.

We will build our results and discussions using this generalization bound. Our discussion will focus on how label noise

can disrupt the training of a model through the changes of the following **Excessive Generalization Error:**

$$\mathsf{err}^{+}(\mathcal{P},\mathcal{A}|S) := \sum_{l\in[n]} \tau_l \sum_{x\in X_{S=l}} \mathbb{P}_{h\sim\mathcal{A}(S')}[h(x)\neq y], \tag{4}$$

Note though the input of the algorithm $\mathcal{A}$ is the noisy dataset S', we are interested in the distribution conditional on the clean dataset S - this is the true distribution that we aim for h to generalize to. On the other hand, the distribution induced by S' will necessarily encode bias to the clean distribution that we are interested in, when some labels are indeed different from the true ones. Even though we do not have access to S, the above "true generalization error" is well-defined for our analysis, and nicely encodes three quantities that are of primary interests to our study:

- τ_l: the "importance weight" of the l-appearance instances.
- l: the frequency of instances that categorizes how popular a particular instance x is in the dataset.
- $\sum_{x\in X_{Z=l}} \mathbb{P}_{h\sim\mathcal{A}(S')}[h(x)\neq y]$: the accumulative generalization error h makes for l-appearance instances.

We will also denote by

$$\mathsf{err}^{+}_{l}(\mathcal{P},\mathcal{A},x|S) := \tau_l \cdot \mathbb{P}_{h\sim\mathcal{A}(S')}[h(x)\neq y] \tag{5}$$

the **Individual Excessive Generalization Error** caused by a $x \in X_{S=l}$. Easy to see that $\mathsf{err}^{+}(\mathcal{P},\mathcal{A}|S) = \sum_x \mathsf{err}^{+}_{l}(\mathcal{P},\mathcal{A},x|S)$.

3.2. Importance of memorizing an l-appearance instance

Clearly Eqn. (4) informs us that different instance contributes differently to the generalization error. It was proved in Feldman (2020) even a single-appearance instance $x \in X_{S=1}$ (i.e., $l = 1$) will contribute to the increase of generalization error at the order of $\Omega(\frac{1}{n})$: when $\pi_{\max} := \max_{j\in[N]} \pi_j \leq 1/200$, we have

$$\tau_1 \geq \frac{1}{7n} \cdot \mathsf{weight}\left(\pi, \left[\frac{1}{2n}, \frac{1}{n}\right]\right),$$

where $\mathsf{weight}\big(\pi, \big[\beta_1,\beta_2\big]\big)$ is the expected fraction of distribution D contributed by frequencies in the range $[\beta_1,\beta_2]$:

$$\mathsf{weight}\big(\pi, \big[\beta_1,\beta_2\big]\big) := \mathbb{E}\left[\sum_{x\in X} D(x)\cdot \mathbb{1}\left(D(x)\in[\beta_1,\beta_2]\right)\right]$$

The expectation is w.r.t. $D(x) \sim \pi$ (and followed by the normalization procedure). We next first generalize the above lower bound to τ_l for an arbitrary l:

Theorem 4. *For sufficiently large n, N, when $\pi_{max} \leq \frac{1}{20}$:*

$$\tau_l \geq 0.4 \cdot \frac{l(l-1)}{n(n-1)} \cdot \mathit{weight}\left(\pi, \left[\frac{2}{3}\frac{l-1}{n-1}, \frac{4}{3}\frac{l}{n}\right]\right) \tag{6}$$

We observe that $\frac{l(l-1)}{n(n-1)} = O(\frac{l^2}{n^2})$. For instance:

- An $l = O(n^{2/3})$-appearance instance will lead to an $\Omega(\frac{1}{n^{2/3}})$ order of impact.
- An $l = O(n^{3/4})$-appearance instance will lead to an $\Omega(\frac{1}{\sqrt{n}})$ order of impact.
- An $l = cn$-appearance (linear) instance will lead to an $\Omega(1)$ bound, a constant order of impact.

Secondly, for the $\mathsf{weight}\big(\pi, \big[\frac{2}{3}\frac{l-1}{n-1}, \frac{4}{3}\frac{l}{n}\big]\big)$ term, we have the frequency interval at the order of length $\frac{l}{n} - \frac{l-1}{n-1} = \frac{n-l}{n(n-1)} = O(\frac{1}{n})$. That is the weight term captures the frequency of an $O(\frac{1}{n})$ interval of the sample distribution. One might notice that there seems to be a disagreement with the reported result in Feldman (2020) when l is small (particularly when $l = 1$): when ignoring the weight term τ_l, an $O(\frac{1}{n})$ lower bound was reported, while ours leads to an $O(\frac{1}{n^2})$ one. This is primarily due to different bounding techniques we incurred. Our above bound suits the study of l that is on a higher order than $O(\frac{1}{n})$. For small l, we provide the following bound:

Theorem 5. *For sufficiently large n, N and $\pi_{max} \leq \frac{1}{20}$:*

$$\tau_l \geq 0.4\frac{l-1}{n-1} \cdot \frac{1}{1.1^l} \cdot \mathit{weight}\left(\pi, \left[0.7\frac{l-1}{n-1}, \frac{4}{3}\frac{l-1}{n-1}\right]\right)$$

When l is small, based on the above bound, we do see $\tau_l = \Omega(\frac{1}{n})$, while the weight constant again captures an $O(\frac{1}{n})$ interval of instances. Note that this bound becomes less informative as l grows, due to the increasing 1.1^l term. Also when $l = 1$, our bound becomes vacuous since $l - 1 = 0$.

3.3. Memorizing noisy labels

In order to study the negative effects of memorizing noisy labels, we first define the memorization of noisy labels. For an $x \in X_{S=l}$ and its associated l noisy labels, denote by $\tilde{\mathbb{P}}[\tilde{y} = k|x], k \in Y$ the empirical distribution of the l noisy labels: for instance when $l = 3$ and two noisy labels are 1, we have $\tilde{\mathbb{P}}[\tilde{y} = 1|x] = \frac{2}{3}$.

Definition 1 (Memorization of noisy labels)**.** *We call a model h memorizing noisy labels for instance x if $\mathbb{P}_{h\sim\mathcal{A}(S')}[h(x) = k] = \tilde{\mathbb{P}}[\tilde{y} = k|x]$.*

Note that the probability measure is over the randomness of the algorithm $\mathcal{A}$, as well as the potential randomness in h - practically one can sample a classification outcome based on the posterior prediction of $h(x)$. Effectively the assumption states that when a model, e.g. a deep neural network, memorizes all l noisy labels for instance x, its output will follow the same empirical distribution. It has been shown in the literature (Cheng et al., 2020b;a) that a fully memorizing neural network will be able to encode $\tilde{\mathbb{P}}[\tilde{y} = k|x]$ for each x. This is also what we observe empirically. In

Figure 2, we simulate a 2D example: there are two classes of instances. The outer annulus represents one class and the inner ball is the other. Given the plotted training data, we train a 2-layer neural network using the cross-entropy (CE) loss. On the left panel, we observe concentrated predictions from the trained model when labels are clean. However, the decision boundary (colored bands with different prediction probabilities) becomes less certain and more probabilistic with the addition of noisy labels, signaling that the neural network is memorizing a mixed distribution of noisy labels.

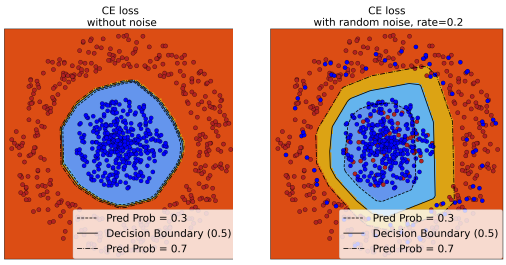

Figure 2. A 2D example illustrating the memorization of noisy labels. Left panel: Training with clean labels. Right panel: Training with random 20% noisy labels. Example with 40% noisy labels can be found in the Appendix.

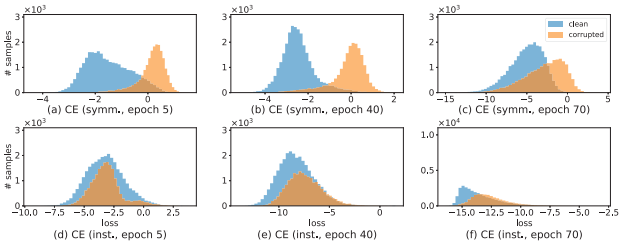

Figure 3. Memorization effects on CIFAR-10 with noisy labels.

We further illustrate this in Figure 3 where we train a neural network on the CIFAR-10 dataset with synthesized noisy labels. The top row simulated simple cases with instance-independent noise $T(x) \equiv T$, while the bottom one synthesized an instance-dependent case[1]. In each row, from Left to Right, we show the progressive changes in the distribution of losses[2] across different training epochs. Preferably, we would like the training to return two distributions of losses that are less overlapped so the model can better distinguish the clean (colored in blue) and corrupted instances (colored in orange). However, we do observe that in both cases, the neural network fails to separate the clean instances from the corrupted ones and memorizes a mixture of both.

This definition of memorization is certainly a simplification but it succinctly characterizes the situation when there are l similar but non-identical instances, the deep neural network would memorize the noisy label for each of them, which then results in memorizing each realized noisy label class a $\tilde{\mathbb{P}}[\tilde{y} = k|x]$ fraction of times. This definition would also require the instances (x's) to be rather independent, or each x's own label information is the most dominant one, which is likely to be true when N is large enough to separate X. Most of our observations would remain true as long as the memorization leads h to predict in the same direction of $\tilde{\mathbb{P}}[\tilde{y} = k|x]$. In particular, when h does not fully remember the empirical label distribution, we conjecture that our main results hold if the memorization preserves orders: for any two classes k, k', if $\tilde{\mathbb{P}}[\tilde{y} = k|x] > \tilde{\mathbb{P}}[\tilde{y} = k'|x]$, we require h to satisfy $\mathbb{P}_{h\sim\mathcal{A}(S')}[h(x) = k] > \mathbb{P}_{h\sim\mathcal{A}(S')}[h(x) = k']$. This simplification in Definition 1 greatly enables a clear presentation of our later analysis.

3.4. Impacts of memorizing noisy labels

Based on Theorem 4, we summarize our first observation that over-memorizing noisy labels for higher frequency instances leads to a bigger drop in the generalization power:

Theorem 6. *For $x \in X_{S=l}$ with true label y, h memorizing its l noisy labels leads to the following order of individual excessive generalization error* $\mathsf{err}_l^+(\mathcal{P}, \mathcal{A}, x|S)$*:*

$$\Omega\left(\frac{l^2}{n^2} \cdot \mathit{weight}\left(\pi, \left[\frac{2}{3}\frac{l-1}{n-1}, \frac{4}{3}\frac{l}{n}\right]\right) \cdot \sum_{k\neq y} \tilde{\mathbb{P}}[\tilde{y} = k|x]\right)$$

We would like to note that with large l, $\sum_{k\neq y} \tilde{\mathbb{P}}[\tilde{y} = k|x] \to \sum_{k\neq y} T_{y,k}(x)$ - not surprisingly, the higher probability an instance is observing a corrupted label, the higher generalization error it will incur. The bound informs us that over-memorizing high-frequency instances lead to a larger negative impact on the generalization. However, we shall see later the higher-frequency instances are in fact the easier ones to fix! On the other hand, memorizing the noisy labels for the lower frequency/appearance instances leads to a smaller drop in generalization performance. Nonetheless, they do incur non-negligible changes. For instance, misremembering a single instance with $l = 1$ leads to an $O(\frac{1}{n})$ increase in generalization error. Later we show a small l poses additional challenges in correcting the mistakes.

4. Learning with Noisy Labels

In this section, we quickly review a subset of popular and recently proposed solutions for learning with noisy labels.

4.1. Loss correction

Arguably one of the most popular approaches for correcting the effects of label noise is through loss correction using the knowledge of $T(x)$ (Natarajan et al., 2013; Liu & Tao, 2016;

[1] We defer the empirical details to the Appendix.

[2] To better visualize the separation of the instances, we follow Cheng et al. (2020a) to plot the distribution of a normalized loss by subtracting the CE loss with a normalization term $\sum_k \mathbb{P}[h(x) = k]/m$, resulting possibly negative losses on x-axis.

Patrini et al., 2017). Let's denote by $\ell : \mathbb{R}^m \times Y \to \mathbb{R}_+$ the underlying loss function we adopted for training a deep neural network. Denote by $\ell(h(x), y)$ the loss incurred by h on instance (x, y), and $\boldsymbol{\ell}(h(x)) = [\ell(h(x), y')]_{y' \in Y}$ the column vector form of the loss. We will assume each $T(x)$ is invertible that $T^{-1}(x)$ well exists. Loss correction is done via defining a surrogate loss function $\tilde{\ell}$ as follows:

$$\boldsymbol{\ell}_{\text{LC}}(h(x)) = T^{-1}(x) \cdot \boldsymbol{\ell}(h(x)). \tag{7}$$

The reason for performing the above correction is due to the following established unbiasedness property: Denote by $\tilde{\mathbf{y}}$ the one-hot encoding column vector form of the noisy label $\tilde{y}$: $\tilde{\mathbf{y}} := [0; ...; \underbrace{1}_{\tilde{y}'s \text{ position}}; ...; 0]$, we have:

Lemma 1 (Unbiasedness of ℓ_{LC}, (Natarajan et al., 2013))**.** $\mathbb{E}_{\tilde{y}|y}[\tilde{\mathbf{y}}^\top \cdot \boldsymbol{\ell}_{\text{LC}}(h(x))] = \ell(h(x), y)$.

The above lemma states that when conditioning on the distribution of $\tilde{y}|y$, $\boldsymbol{\ell}_{\text{LC}}(h(x))$ is unbiased in expectation w.r.t. the true loss $\ell(h(x), y)$. In Section 4.4, we explain how this unbiasedness is established for the binary classification setting. Based on Lemma 1, one can perform empirical risk minimization over $\sum_{i=1}^n \tilde{\mathbf{y}}_i^\top \cdot \boldsymbol{\ell}_{\text{LC}}(h(x_i))$, hoping the empirical sum will approximately converge to its expectation which will then equalize to the true empirical loss $\sum_{i=1}^n \ell(h(x_i), y_i)$. Of course, a commonly made assumption/requirement when applying this approach is that $T(x) \equiv T, \forall x$, and T can be estimated accurately enough. There exist empirical and extensive discussions on how to do so (Patrini et al., 2017; Xia et al., 2019; Yao et al., 2020b; Zhang et al., 2021; Li et al., 2021).

4.2. Label smoothing

Label smoothing has demonstrated its benefits in improving learning representation (Müller et al., 2019). A recent paper (Lukasik et al., 2020) has also proved the potential of label smoothing in defending training against label noise. Denote by $\mathbf{1}$ the all-one vector, and a smoothed and soft label is defined as $\mathbf{y}_{\text{LS}} := (1 - a) \cdot \tilde{\mathbf{y}} + \frac{a}{m} \cdot \mathbf{1}$, where $a \in [0, 1]$ is a smoothing parameter. That is, $\mathbf{y}_{\text{LS}}$ is defined as a linear combination of the noisy label $\tilde{y}$ and an uninformative and uniform label vector $\mathbf{1}$. Then each instance x will be evaluated using $\mathbf{y}_{\text{LS}}$: $\mathbf{y}_{\text{LS}}^\top \cdot \boldsymbol{\ell}(h(x))$.

Though label smoothing has shown promising advantages over loss correction, there are few theoretical understandings of why so, except for its high-level idea of being "conservative" when handling noisy labels.

4.3. Peer loss

Peer loss (Liu & Guo, 2020) is a different line of solution that promotes the use of multiple instances simultaneously while evaluating a particular noisy instance $(x, \tilde{y})$. A salient feature of peer loss is that the implementation of it does not require the knowledge of $T(x)$. The definition for peer loss has the following key steps:

- For each $(x, \tilde{y})$ we aim to evaluate, randomly drawn two other sample indices $p_1, p_2 \in [n]$.
- Pair x_{p_1} with $\tilde{y}_{p_2}$, define peer loss:

$$\ell_{\text{PL}}(h(x), \tilde{y}) := \ell(h(x), \tilde{y}) - \ell(h(x_{p_1}), \tilde{y}_{p_2}). \tag{8}$$

When $T(x) \equiv T$, it was proved in (Liu & Guo, 2020) that for binary classification with equal label prior, when ℓ is the 0-1 loss, minimizing peer loss returns the same minimizer of $\mathbb{E}[\ell(h(x), y)]$ on the clean distribution.

4.4. Memorization paradox

Some of the above approaches have established strong theoretical guarantees of recovering the optimal classifier in expectation when using only noisy training labels (Natarajan et al., 2013; Liu & Guo, 2020; Ma et al., 2020). Why would we need a different understanding? First of all, most theoretical results assumed away the outstanding challenges of having an unknown number and distribution of noise rate matrix $T(x)$ (either needed for estimation purpose or for handling them implicitly) and focused on a single transition matrix T. Secondly, the existing error analysis often focuses on the distribution level, while we would like to zoom in to each instance that occurs with a different frequency.

In addition, we now highlight a paradox introduced by a commonly made assumption that the noisy labels and the model's prediction t are conditionally independent given true label y: $\mathbb{P}[t, \tilde{y}|y] = \mathbb{P}[t|y] \cdot \mathbb{P}[\tilde{y}|y]$, or that t is simply deterministic that it does not encode the information of $\tilde{y}$. This assumption is often needed when evaluating the expected generalization error under noisy distributions. The independence can be justified by modeling $\tilde{y}$ as being conditionally independent of feature x, which the prediction t is primarily based on.

Let's take loss correction for an example. For a clear demonstration, let's focus on the binary case $y \in \{-1, +1\}$. Consider a particular x, define $e_-(x) := \mathbb{P}[\tilde{y} = +1|y = -1, x]$, $e_+(x) := \mathbb{P}[\tilde{y} = -1|y = +1, x]$ and $T(x)$:

$$T(x) := \begin{bmatrix} 1 - e_-(x) & e_-(x) \\ e_+(x) & 1 - e_+(x) \end{bmatrix} \tag{9}$$

Easy to verify its inverse is:

$$T^{-1}(x) = \frac{1}{1 - e_+(x) - e_-(x)} \begin{bmatrix} 1 - e_+(x) & -e_-(x) \\ -e_+(x) & 1 - e_-(x) \end{bmatrix} \tag{10}$$

For the rest of this section, without confusion, let's shorthand $e_+(x), e_-(x)$ as e_+, e_-. Then loss correction (Eqn.

(7)) takes the following form:

$$\ell_{\text{LC}}(h(x),-1) = \frac{(1-e_+)\cdot\ell(h(x),-1) - e_-\cdot\ell(h(x),+1)}{1-e_+-e_-}$$
$$\ell_{\text{LC}}(h(x),+1) = \frac{(1-e_-)\cdot\ell(h(x),+1) - e_+\cdot\ell(h(x),-1)}{1-e_+-e_-}$$

Consider the case with true label $y=+1$. The following argument establishes the unbiasedness of ℓ_{LC} (reproduced from Natarajan et al. (2013), with replacing and instantiating a prediction t with $h(x)$):

$$\begin{aligned}
&\mathbb{E}_{\tilde{y}|y=+1}[\ell_{\text{LC}}(h(x),\tilde{y})]\\
&= (1-e_+)\cdot\ell_{\text{LC}}(h(x),+1) + e_+\cdot\ell_{\text{LC}}(h(x),-1)\\
&= (1-e_+)\frac{(1-e_+)\cdot\ell(h(x),-1) - e_-\cdot\ell(h(x),+1)}{1-e_+-e_-}\\
&+ e_+\frac{(1-e_-)\cdot\ell(h(x),+1) - e_+\cdot\ell(h(x),-1)}{1-e_+-e_-}\\
&= \ell(h(x),+1) = \ell(h(x),y=+1)\,,
\end{aligned}$$

That is the conditional expectation of $\ell_{\text{LC}}(h(x),\tilde{y})$ recovers the true loss $\ell(h(x),y)$. Nonetheless, the first equality assumed the conditional independence between h and $\tilde{y}$, given y. When h is output from a deep neural network and memorizes all noisy labels $\tilde{y}$, the above independence condition can be challenged. As a consequence, it is unclear whether the classifiers that fully memorize the noisy labels would result in a lower empirical loss during training. We call the above observation the *memorization paradox*. We conjecture that this paradox leads to inconsistencies in previously observed empirical evidence, especially when training a deep neural network solution that memorizes labels well. In the next section, we will offer new explanations for how the proposed solutions actually fared when the trained neural network is able to memorize the examples.

5. How Do Solutions Fare at Instance Level?

In this section, we revisit how the above solutions offer fixes at the instance level and under what conditions they might fail to work. Unless stated otherwise, throughout the section, we focus on a particular instance $x \in X_{S=l}$ with true label y and l corresponding noisy labels $\tilde{y}$'s, one for each appearance. With limited space, the goal is to provide a template for carrying out further analysis for methods that are of individual interest. The three presented approaches were selected carefully as representatives for:

- Mainstream approaches (loss correction, label correction, loss reweighting etc) that use noise transition matrix T (**loss correction** in this paper).
- Robust losses that regularize against noisy outliers (**label smoothing** in this paper).
- More recent approaches that do not require the noise transition matrix (**peer loss** in this paper).

5.1. Loss correction

We start with loss correction and notice that the loss correction step is equivalent to the following "label correction" [3] procedure. Denote by $\mathbb{P}(\tilde{\mathbf{y}}) := [\mathbb{P}[\tilde{y}=k|x]]_{k\in Y}$ the vector form of the distribution of noisy label $\tilde{y}$, and $\mathbf{y}$ as the vector form of one-hot encoding of the true label y. As assumed earlier, the generation of noisy label $\tilde{y}$ follows: $\mathbb{P}(\tilde{\mathbf{y}}) := T^\top(x)\cdot\mathbf{y}$. When $T(x)$ is invertible (commonly assumed), we will have $\mathbf{y} = (T^{-1}(x))^\top\cdot\mathbb{P}(\tilde{\mathbf{y}})$, that is when $\mathbb{P}(\tilde{\mathbf{y}})$ is the true and exact posterior distribution of $\tilde{y}$, $(T^{-1}(x))^\top\cdot\mathbb{P}(\tilde{\mathbf{y}})$ recovers $\mathbf{y}$. Based on the above observation, easily we can show that (using linearity of expectation) loss correction effectively pushed h to memorize $(T^{-1}(x))^\top\cdot\mathbb{P}(\tilde{\mathbf{y}}) = \mathbf{y}$, i.e. the clean label:

$$\begin{aligned}
&\mathbb{E}_{\tilde{y}|y}[\tilde{\mathbf{y}}^\top\cdot\boldsymbol{\ell}_{\text{LC}}(h(x))]\\
=&\mathbb{E}_{\tilde{y}|y}[\tilde{\mathbf{y}}^\top\cdot T^{-1}(x)\cdot\boldsymbol{\ell}(h(x))]\\
=&\mathbb{E}_{\mathbf{y}'\sim(T^{-1}(x))^\top\cdot\mathbb{P}(\tilde{\mathbf{y}})}\left[(\mathbf{y}')^\top\cdot\boldsymbol{\ell}(h(x))\right]\\
=&\mathbf{y}^\top\cdot\boldsymbol{\ell}(h(x)) \qquad (11)
\end{aligned}$$

That is loss correction encourages h to memorize the true label $\mathbf{y}$, therefore reducing $\mathbb{P}[h(x)\neq y]$ to 0 to improve generalization.

The above is a clean case with accessing $\mathbb{P}(\tilde{\mathbf{y}})$, the exact noisy label distribution, which differs from the empirical noisy label distribution $\tilde{\mathbb{P}}[\tilde{y}=k|x]$ that a deep neural network can access and memorize. This is mainly due to the limited number, l, of noisy labels for an $x\in X_{S=l}$. Denote by $\tilde{\mathbb{P}}(\tilde{\mathbf{y}})$ the vector form of $\tilde{\mathbb{P}}[\tilde{y}=k|x]$. Let $\mathbf{y}_{\text{LC}}$ be the "corrected label" following from the distribution defined by $(T^{-1}(x))^\top\cdot\tilde{\mathbb{P}}(\tilde{\mathbf{y}})$:

$$\text{Corrected Label:}\quad \mathbf{y}_{\text{LC}} = (T^{-1}(x))^\top\cdot\tilde{\mathbb{P}}(\tilde{\mathbf{y}}).$$

Denote by $x(1),...,x(l)$ the l appearance of x, and $\tilde{y}(1),...,\tilde{y}(l)$ the corresponding noisy labels. Similar to Eqn. (11) we can show that:

$$\begin{aligned}
&\frac{1}{l}\sum_{i=1}^{l}\tilde{\mathbf{y}}^\top(i)\cdot\boldsymbol{\ell}_{\text{LC}}(h(x))\\
=&\mathbb{E}_{\tilde{\mathbb{P}}(\tilde{\mathbf{y}})}[\tilde{\mathbf{y}}^\top\cdot T^{-1}(x)\cdot\boldsymbol{\ell}(h(x))]\\
=&\mathbf{y}_{\text{LC}}^\top\cdot\boldsymbol{\ell}(h(x)) \qquad (12)
\end{aligned}$$

That is, the empirical loss for x with loss correction is equivalent with training using $\mathbf{y}_{\text{LC}}$! Next we will focus on the binary case: $T(x)$ is fully characterized and determined by $e_+(x), e_-(x)$ (Eqn. 9). We will follow the assumption made in the literature that $e_+(x)+e_-(x)<1$ (noisy labels

[3] Please note that our " label correction" definition differs from the existing ones in the literature.

are at least positively correlating with the true label). Easy to prove that the two entries of $\mathbf{y}_{\text{LC}}$ add up to 1[4]:

Lemma 2. $\mathbf{y}_{\text{LC}}[1] + \mathbf{y}_{\text{LC}}[2] = 1$.

However, it is possible that $(T^{-1}(x))^\top \cdot \tilde{\mathbb{P}}(\tilde{\mathbf{y}})$ is not a valid probability measure, in which case we will simply cap $\mathbf{y}_{\text{LC}}$ at either $[1; 0]$ or $[0; 1]$. Denote by y_{LC} the random variable drawn according to $\mathbf{y}_{\text{LC}}$. We again call h memorizing $\mathbf{y}_{\text{LC}}$ if $\mathbb{P}[h(x) = k] = \mathbb{P}[y_{\text{LC}} = k|x], \forall k \in Y$. Let's simplify our argument by assuming the following equivalence:

Assumption 1. *The trained model h using loss correction (minimizing Eqn. (12)) is able to memorize $\mathbf{y}_{\text{LC}}$.*

The first message we are ready to send is: **For an x with large l, with high probability, loss correction returns smaller generalization error than memorizing noisy labels.** Denote by $sgn(y)$ the sign function of y. For $x \in X_{S=l}$, consider a non-trivial case that $\tilde{\mathbb{P}}[\tilde{y} \neq y|x] > 0$[5]:

Theorem 7. *For an $x \in X_{S=l}$ with true label y, w.p. at least $1 - e^{-2l\left(1/2 - e_{sgn(y)}(x)\right)^2}$, h memorizing $\mathbf{y}_{\text{LC}}$ returns a lower error $\mathbb{P}[h(x) \neq y]$ than memorizing the noisy label s.t. $\mathbb{P}[h(x) = k] = \tilde{\mathbb{P}}[\tilde{y} = k|x]$.*

The above theorem implies when $l \geq \frac{\log 1/\delta}{2\left(\frac{1}{2} - e_+(x)\right)^2}$, memorizing $\mathbf{y}_{\text{LC}}$ improves the excessive generalization error with probability at least $1 - \delta$. As a corollary of Theorem 7:

Corollary 1. *For an $x \in X_{S=l}$ with true label y, w.p. at least $1 - e^{-2l\left(\frac{1}{2} - e_{sgn(y)}(x)\right)^2}$, performing loss correction for $x \in X_{S=l}$ improves the excessive generalization error $\mathsf{err}_l^+(\mathcal{P}, \mathcal{A}, x|S)$ by*

$$\Omega\left(\frac{l^2}{n^2} \cdot \textit{weight}\left(\pi, \left[\frac{2}{3}\frac{l-1}{n-1}, \frac{4}{3}\frac{l}{n}\right]\right)\right)$$

The above corollary is easily true due to definition of $\mathsf{err}_l^+(\mathcal{P}, \mathcal{A}, x|S)$, Theorem 6 & 7, as well as Assumption 1.

Our next message is: **For an x with small l, loss correction fails with a substantial probability.** Denote by $D_{\text{KL}}(\frac{1}{2}\|e)$ the Kullback-Leibler distance between two Bernoulli 0/1 random variables of parameter $1/2$ and e. Consider a non-trivial case that $\tilde{\mathbb{P}}[\tilde{y} \neq y|x] < 1$, we prove:

Theorem 8. *For an $x \in X_{S=l}$ with true label y, w.p. at least $\frac{1}{\sqrt{2l}} \cdot e^{-l \cdot D_{\text{KL}}\left(\frac{1}{2}\|e_{sgn(y)}(x)\right)}$, h memorizing $\mathbf{y}_{\text{LC}}$ returns a higher error $\mathbb{P}[h(x) \neq y]$ than memorizing noisy label $\tilde{\mathbf{y}}$.*

When l is small, the reported probability in Theorem 8 is a non-trivial one. Particularly, when $l \leq \frac{\log \frac{1}{\sqrt{2}\delta}}{D_{\text{KL}}(\frac{1}{2}\|e_{sgn(y)}(x))}$, with probability at least δ, memorizing $\mathbf{y}_{\text{LC}}$ (or performing loss correction) leads to worse generalization power.

[4]For binary labels $\{-1, +1\}$, the first entry of the vectors corresponds to -1 ($\mathbf{y}_{\text{LC}}[1]$), the second for $+1$ ($\mathbf{y}_{\text{LC}}[2]$).

[5]For trivial cases, our claims would simply be that loss correction performs equally well since the memorizing the noisy label is already equivalent with memorizing the true label.

5.2. Label smoothing

Denote by y_{LS} the "soft label" for the distribution vector $\mathbf{y}_{\text{LS}}$, and again we call h memorizing $\mathbf{y}_{\text{LS}}$ if $\mathbb{P}[h(x) = k] = \tilde{\mathbb{P}}[y_{\text{LS}} = k|x], \forall k \in Y$ - $\tilde{\mathbb{P}}[y_{\text{LS}} = k|x]$ denotes the empirical distribution for y_{LS} (empirical average of the soft label y_{LS}):

$$\tilde{\mathbb{P}}[y_{\text{LS}} = k|x] = (1 - a) \cdot \tilde{\mathbb{P}}[\tilde{y} = k|x] + \frac{a}{m}\,. \tag{13}$$

Consider the binary classification case. Denote the following event: $\mathcal{E}_+ := \{\tilde{\mathbb{P}}[\tilde{y} = +1|x] > \tilde{\mathbb{P}}[\tilde{y} = -1|x]\}$ and $\bar{\mathcal{E}}_+$ denotes the opposite event $\tilde{\mathbb{P}}[\tilde{y} = +1|x] < \tilde{\mathbb{P}}[\tilde{y} = -1|x]$. For a non-trivial case $\tilde{\mathbb{P}}[\tilde{y} \neq y|x] \in (0, 1)$:

Theorem 9. *For an $x \in X_{S=l}$ with true label $y = +1$, when $\mathcal{E}_+$ happens, h memorizing the smooth label $\mathbf{y}_{\text{LS}}$ leads to a higher error $\mathbb{P}[h(x) \neq y]$ than memorizing corrected label $\mathbf{y}_{\text{LC}}$. When $\bar{\mathcal{E}}_+$ happens, h memorizing $\mathbf{y}_{\text{LS}}$ has a lower error $\mathbb{P}[h(x) \neq y]$ than memorizing $\mathbf{y}_{\text{LC}}$.*

Similar result can be proved for the case with $y = -1$ but we will not repeat the details. Repeating the proofs for Theorem 8, we can similarly show that when l is small, there is a substantial probability that $\bar{\mathcal{E}}_+$ will happen $\left(\geq \frac{1}{\sqrt{2l}} \cdot e^{-l \cdot D_{\text{KL}}\left(\frac{1}{2}\|e_{sgn(y)}(x)\right)}\right)$, therefore label smoothing returns a better generalization power than loss correction. Intuitively, consider the extreme case with $l = 1$, and label smoothing has a certain correction power even when this single noisy label is wrong. On the other hand, loss correction (and $\mathbf{y}_{\text{LC}}$) would memorize this single noisy label for x. We view label smoothing as a safe way to perform label correction when l is small, and when the noise rate is excessively high such that $\mathbb{P}[\bar{\mathcal{E}}_+] > \mathbb{P}[\mathcal{E}_+]$.

5.3. Peer loss

The first message we send for peer loss is that: **For an x with larger l, peer loss extremizes h's prediction to the correct label with high probability.** We first show that peer loss explicitly regularizes h from memorizing noisy labels. We use the cross-entropy loss for ℓ in ℓ_{PL} (Eqn. (8)).

Lemma 3. *Denote by $\mathbb{Q}(x, \tilde{y})$ the joint distribution of $h(x)$ and $\tilde{y}$, $\mathbb{P}(x), \mathbb{P}(\tilde{y})$ the marginals of $x, \tilde{y}$, taking expectation of ℓ_{PL} over the training data distribution $\mathbb{P}(x, \tilde{y})$, one finds:*

$$\begin{aligned}\mathbb{E}_{\mathcal{P}}\left[\ell_{\text{PL}}(h(x), \tilde{y})\right] = {} & D_{\text{KL}}(\mathbb{Q}(x, \tilde{y}) \| \mathbb{P}(x, \tilde{y})) \\ & - D_{\text{KL}}(\mathbb{Q}(x, \tilde{y}) \| \mathbb{P}(x) \times \mathbb{P}(\tilde{y})).\end{aligned} \tag{14}$$

In above we use the standard notation D_{KL} for KL-divergence between two distributions. While minimizing $D_{\text{KL}}(\mathbb{Q}(x, \tilde{y}) \| \mathbb{P}(x, \tilde{y}))$ encourages h to reproduce $\mathbb{P}(x, \tilde{y})$ (the noisy distribution), the second term $D_{\text{KL}}(\mathbb{Q}(x, \tilde{y}) \| \mathbb{P}(x) \times \mathbb{P}(\tilde{y}))$ discourages h from doing so by incentivizing h to predict a distribution that is independent from $\tilde{y}$! This regularization power helps lead the training to generate more confident predictions, per a recent result:

Theorem 10. *[(Cheng et al., 2020a)] When minimizing* $\mathbb{E}\left[\ell_{\text{PL}}(h(x), \tilde{y})\right]$*, solutions satisfying* $\mathbb{P}[h(x) = k] > 0, \forall k \in Y$ *are not optimal.*

In the case of binary classification, the above theorem implies that we must have either $\mathbb{P}[h(x) = +1] \to 1$ or $\mathbb{P}[h(x) = -1] \to 1$. We provide an illustration of this effect in Figure 4. In sharp contrast to Figure 2, the decision boundaries returned by training using peer loss remain tight, despite high presence of label noise.

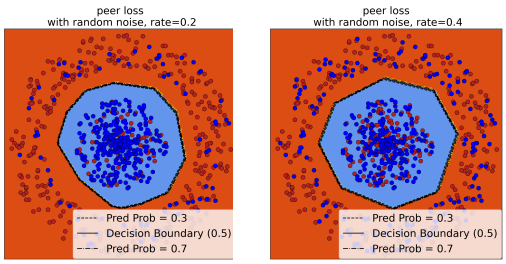

Figure 4. 2D example with peer loss: $20\%, 40\%$ random noise.

Now the question left is: is pushing to confident prediction in the right direction of correcting label noise? If yes, peer loss seems to be achieving the same effect as the loss correction approach, without knowing or using $T(x)$. Consider the binary classification case. To simplify our analysis, let's consider only a class-dependent noise setting that $e_+(x) \equiv e_+, e_-(x) \equiv e_-$. This simplification is needed due to the fact that the definition of peer loss requires drawing a global and random "peer sample". Technically we can revise peer loss to use "peer samples" that are similar enough so that they will likely to have the same $e_+(x), e_-(x)$. Denote the priors of the entire distribution by $p_+ := \mathbb{P}_{y' \in \mathcal{F}|S}[y' = +1] > 0, p_- := \mathbb{P}_{y' \in \mathcal{F}|S}[y' = -1] > 0$. When n is sufficiently large, we prove:

Theorem 11. *For an* $x \in X_{S=l}$ *with true label* y*, w.p. at least* $1 - e^{\frac{-2l}{p^2_{sgn(-y)} \cdot (1-e_+-e_-)^2}}$*, predicting* $\mathbb{P}[h(x) = y] = 1$ *leads to smaller training loss in* ℓ_{PL} *(with* ℓ*=CE loss).*

Using above theorem and Theorem 6, consider a non-trivial case $\tilde{\mathbb{P}}[\tilde{y} \neq y|x] > 0$ we have:

Corollary 2. *For an* $x \in X_{S=l}$ *with true label* y*, w.p. at least* $1 - e^{\frac{-2l}{p^2_{sgn(-y)} \cdot (1-e_+-e_-)^2}}$*, training using* ℓ_{PL} *improves the individual excessive generalization error* $\text{err}^+_l(\mathcal{P}, \mathcal{A}, x|S)$ *by:*

$$\Omega\left(\frac{l^2}{n^2}\textit{weight}\left(\pi, \left[\frac{2}{3}\frac{l-1}{n-1}, \frac{4}{3}\frac{l}{n}\right]\right) \cdot \sum_{k \neq y} \tilde{\mathbb{P}}[\tilde{y} = k|x]\right)$$

Both peer loss and loss correction implicitly use the posterior distribution of noisy labels to hopefully extremize $h(x)$ to the correct direction. The difference is peer loss extremizes even more ($\mathbb{P}[h(x) = k] \to 1$ for some k) when it is confident. Therefore, when there is sufficient information (i.e., l being large), peer loss tends to perform better than loss correction which needs explicit knowledge of the true transition matrix and performs a precise and exact bias correction step. This is also noted empirically in (Liu & Guo, 2020), and our results provide the theoretical justifications.

Similar to Theorem 8, when l is small, the power of peer loss does seem to drop: **For an** x **with small** l**, peer loss extremizes** h**'s prediction to the wrong label with a substantial probability.** Formally,

Theorem 12. *For an* $x \in X_{S=l}$ *with true label* y*, w.p. at least* $\frac{1}{\sqrt{2l}} e^{-l \cdot D_{\text{KL}}\left(\frac{1}{2} \| e_{sgn(y)}\right)}$*, predicting* $\mathbb{P}[h(x) = -y] = 1$ *leads to smaller training loss in* ℓ_{PL}*.*

For a non-trivial case that $\tilde{\mathbb{P}}[\tilde{y} \neq y|x] < 1$, this implies higher prediction error than memorizing the noisy labels.

6. Takeaways and Conclusion

We studied the impact of a model memorizing noisy labels. This paper proved the disparate impact of noisy labels at the instance level, and then the fact that existing treatments can often lead to disparate outcomes, with low-frequency instances being more likely to be mistreated. This observation is particularly concerning when a societal application is considered, and the low-frequency examples are drawn from a historically disadvantaged population (thus low presence in data). Specifically:

Frequent instance While high-frequent instances (large l) have a higher impact on the generalization bound, and misclassifying one such example would be more costly, our analysis shows that due to the existence of a good number of noisy labels, existing approaches can often counter the negative effects with high probability.

Long-tail instance For a rare instance x with small l, while missing it would incur a much smaller penalty in generalization, still, its impact is non-negligible. Moreover, due to the severely limited label information, we find the noise correction approaches would have a substantial probability of failing to correct such cases.

The above observations require us to rethink the distribution of label noise across instances and might potentially require different treatments for instances in different regimes.

Acknowledgments This work is partially supported by the National Science Foundation (NSF) under grant IIS-2007951 and the NSF FAI program in collaboration with Amazon under grant IIS-2040800. The author thanks Zhaowei Zhu, Xingyu Li and Jiaheng Wei for helps with Figure 1 - 4, as well as Lemma 3. The author thanks anonymous ICML reviewers for their comments that improved the presentation of the paper. The final draft also benefited substantially from discussions with Gang Niu.

References

Arpit, D., Jastrzębski, S., Ballas, N., Krueger, D., Bengio, E., Kanwal, M. S., Maharaj, T., Fischer, A., Courville, A., Bengio, Y., et al. A closer look at memorization in deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 233–242. JMLR. org, 2017.

Ash, R. Information theory. *Dover*, 1990.

Charoenphakdee, N., Lee, J., and Sugiyama, M. On symmetric losses for learning from corrupted labels. In *International Conference on Machine Learning*, pp. 961–970, 2019.

Chen, P., Liao, B. B., Chen, G., and Zhang, S. Understanding and utilizing deep neural networks trained with noisy labels. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 1062–1070. PMLR, 09–15 Jun 2019. URL http://proceedings.mlr.press/v97/chen19g.html.

Cheng, H., Zhu, Z., Li, X., Gong, Y., Sun, X., and Liu, Y. Learning with instance-dependent label noise: A sample sieve approach. *arXiv preprint arXiv:2010.02347*, 2020a.

Cheng, J., Liu, T., Ramamohanarao, K., and Tao, D. Learning with bounded instance-and label-dependent label noise. In *Proceedings of the 37th International Conference on Machine Learning*, ICML '20, 2020b.

Feldman, V. Does learning require memorization? a short tale about a long tail. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*, pp. 954–959, 2020.

Feldman, V. and Zhang, C. What neural networks memorize and why: Discovering the long tail via influence estimation. *NeurIPS*, 2020.

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.

Han, B., Niu, G., Yu, X., Yao, Q., Xu, M., Tsang, I., and Sugiyama, M. Sigua: Forgetting may make learning with noisy labels more robust. In *International Conference on Machine Learning*, pp. 4006–4016. PMLR, 2020.

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. *arXiv preprint arXiv:1712.05055*, 2017.

Jiang, L., Huang, D., Liu, M., and Yang, W. Beyond synthetic noise: Deep learning on controlled noisy labels. In *International Conference on Machine Learning*, pp. 4804–4815. PMLR, 2020.

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

Li, X., Liu, T., Han, B., Niu, G., and Sugiyama, M. Provably end-to-end label-noise learning without anchor points. *arXiv preprint arXiv:2102.02400*, 2021.

Liu, Q., Peng, J., and Ihler, A. T. Variational inference for crowdsourcing. *Advances in neural information processing systems*, 25:692–700, 2012.

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. and Guo, H. Peer loss functions: Learning from noisy labels without knowing noise rates. In *Proceedings of the 37th International Conference on Machine Learning*, ICML '20, 2020.

Lukasik, M., Bhojanapalli, S., Menon, A. K., and Kumar, S. Does label smoothing mitigate label noise? *arXiv preprint arXiv:2003.02819*, 2020.

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.

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.

Menon, A. K., Rawat, A. S., Reddi, S. J., and Kumar, S. Can gradient clipping mitigate label noise? In *International Conference on Learning Representations*, 2019.

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

Neyshabur, B., Bhojanapalli, S., McAllester, D., and Srebro, N. Exploring generalization in deep learning. In *Advances in neural information processing systems*, pp. 5947–5956, 2017.

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 *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

Song, H., Kim, M., Park, D., and Lee, J.-G. Prestopping: How does early stopping help generalization against label noise? 2020.

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.

Xia, X., Liu, T., Wang, N., Han, B., Gong, C., Niu, G., and Sugiyama, M. Are anchor points really indispensable in label-noise learning? *arXiv preprint arXiv:1906.00189*, 2019.

Xia, X., Liu, T., Han, B., Wang, N., Gong, M., Liu, H., Niu, G., Tao, D., and Sugiyama, M. Parts-dependent label noise: Towards instance-dependent label noise. *arXiv preprint arXiv:2006.07836*, 2020.

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.

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. *arXiv preprint arXiv:2006.07805*, 2020b.

Yu, X., Han, B., Yao, J., Niu, G., Tsang, I. W., and Sugiyama, M. How does disagreement help generalization against label corruption? *arXiv preprint arXiv:1901.04215*, 2019.

Zhang, C., Bengio, S., Hardt, M., Recht, B., and Vinyals, O. Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530*, 2016.

Zhang, Y., Niu, G., and Sugiyama, M. Learning noise transition matrix from only noisy labels via total variation regularization. *arXiv preprint arXiv:2102.02414*, 2021.

Zhang, Z. and Sabuncu, M. Generalized cross entropy loss for training deep neural networks with noisy labels. In *Advances in neural information processing systems*, pp. 8778–8788, 2018.

Zhu, X., Anguelov, D., and Ramanan, D. Capturing longtail distributions of object subcategories. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 915–922, 2014.

Zhu, Z., Liu, T., and Liu, Y. A second-order approach to learning with instance-dependent label noise. *CVPR*, 2021a.

Zhu, Z., Song, Y., and Liu, Y. Clusterability as an alternative to anchor points when learning with noisy labels. *ICML*, 2021b.