Understanding Difficulty-based Sample Weighting with a Universal Difficulty Measure

Xiaoling Zhou[1], Ou Wu[⋆1], Weiyao Zhu[1], and Ziyang Liang[1]

Center of Applied Mathematics, Tianjin University, China
{Xiaolingzhou,wuou}@tju.edu.cn
weiyaozhu042@outlook.com, lianggzy98@163.com

Abstract. Sample weighting is widely used in deep learning. A large number of weighting methods essentially utilize the learning difficulty of a training sample to calculate its weight. In this study, this scheme is called difficulty-based weighting. Two important issues arise when explaining this scheme. First, a universal difficulty measure that can be theoretically guaranteed for training samples does not exist. The learning difficulties of the samples are determined by multiple factors, including noise level, imbalance degree, margin, and uncertainty. Nevertheless, existing measures only consider a single factor or in part, but not in their entirety. Second, a comprehensive theoretical explanation is lacking with respect to demonstrating why difficulty-based weighting schemes are effective in deep learning. In this study, we theoretically prove that the generalization error of a sample can be used as a universal difficulty measure. Furthermore, we provide formal theoretical justifications on the role of difficulty-based weighting for deep learning, consequently revealing its positive influences on both the optimization dynamics and generalization performance of deep models, which is instructive to a number of weighting schemes under active research.

Keywords: Sample weighting · Learning difficulty · Generalization error · Deep learning interpretability.

1 Introduction

Treating each training sample unequally improves the learning performance. Two cues are typically considered in designing the weighting schemes of training samples [1]. The first cue is the application context of learning tasks. In applications such as medical diagnosis, samples with high gains/costs are assigned with high weights [2]. The second cue is the characteristics of the training data [3,4,5]. For example, samples with low confidence or noisy labels are assigned with low weights [6]. Characteristic-aware weighting has attracted increasing attention owing to its effectiveness and universality.

Many existing characteristic-aware weighting methods are based on an intrinsic property of the training samples, i.e., their learning difficulty. Based on this property, the training samples can be divided into easy/hard or easy/medium/hard ones [7]. In some schemes, easy samples have higher weights than hard ones, which is called the easy-first mode [3,6,8]. For example, Curriculum learning [9] is motivated by human

⋆ Corresponding author.

learning that easy samples should be learned first, which is verified to be effective on noisy datasets. In some other schemes, hard samples are assigned with high weights, which is called the hard-first mode [4,10,11]. For example, Lin et al. [10] proposed Focal Loss in object detection, which significantly improves the detection performance.

Despite the empirical success of various difficulty-based weighting methods, the process of how difficulty-based weighting positively influences the deep learning models remains unclear. Three recent studies have attempted to investigate the influence of weights on deep learning. Byrd and Lipton [12] empirically found that sample weights affect deep learning by influencing the implicit bias of gradient descent[1]. However, their conclusions were drawn only through experimental observations rather than theoretical analyses. Based on their finding, Xu et al. [13] dedicated to studying how the theoretical understandings for the implicit bias of gradient descent adjust to the weighted empirical risk minimization (ERM) setting. They concluded that assigning high weights to samples with small margins may accelerate optimization. In addition, they established a generalization bound for models that implement learning by using importance weighting. However, their theoretical analyses are only based on the margin-based difficulty measure, resulting in their conclusion being limited and inaccurate in some cases (discussed in Section 4). Unlike studies in view of the implicit bias of gradient descent for ERM, Zhou et al. [1] conjectured that the optimal weight is calculated by the likelihood ratio $p_{opt}[d(x)]/p_{tr}[d(x)]$, where $p_{opt}[d(x)]$ and $p_{tr}[d(x)]$ are the densities of the optimal and real difficulty distributions of the training data. However, how to derive the optimal difficulty distribution is not involved in their study.

Besides the margin-based difficulty measure considered by Xu et al. [13], the existing difficulty measures can be roughly divided into the following five categories.

- Prediction-based measures. This category directly uses the loss [3,6,8] or the predicted probability of the ground truth [4,14] as the difficulty measures. Their intuition is that a large loss (a small probability) indicates a large learning difficulty.
- Gradient-based measures. This category applies the loss gradient in the measurement of the samples' learning difficulty [11,15]. The intuition is that the larger the norm of the gradient, the harder the sample.
- Category proportion-based measures. This category is mainly utilized in imbalanced learning [10], where the category proportion measures the samples' difficulty. The intuition is that the smaller the proportion of a category, the larger the learning difficulty of samples in this category [10,16].
- Margin-based measures. The term "margin" refers to the distance from the sample to the oracle classification boundary. The motivation is that the smaller the margin, the larger the learning difficulty of a sample [17].
- Uncertainty-based measures. This category uses the uncertainty of a sample to measure the difficulty. Aguilar et al. [18] identified hard samples based on epistemic uncertainty and leveraged the Bayesian Neural Network [19] to infer it.

Varying difficulty measures have a significant impact on a difficulty-based weighting strategy. According to the underlying motivations of the five categories of measures,

[1] The implicit bias of gradient descent studies why over-parameterized models are biased toward solutions that generalize well [20].

there are four main factors that greatly influence samples' learning difficulty, including noise level [6,8], imbalance degree [10,16], margin [17], and uncertainty [18]. However, each measure mentioned above only considers a single factor or in part and comes from heuristic inspirations rather than theoretical verification, hindering the application scope of these measures. For example, samples with large margins may also be hard-to-classify in some cases (e.g., with heterogeneous samples in their neighbors). It is desirable to theoretically explore a universal measure capturing all four factors mentioned above, which is essential for our understanding of difficulty-based weighting.

In this study, the manner of how difficulty-based weighting affects deep model training is deeply investigated. First, our analyses support that the generalization error of a training sample can be regarded as a universal difficulty measure for capturing all the four factors described above. Second, based on this unified measure, we characterize the role of difficulty-based weighting on the implicit bias of gradient descent, especially for the convergence speed. Third, two new generalization bounds are constructed to demonstrate the explicit relationship between the sample weights and the generalization performance. The two bounds illuminate a new explanation for existing weighting strategies and characterize the optimal difficulty distribution of the training set formally. Our study takes the first step of constructing a formal theory for difficulty-based sample weighting. In summary, our contributions are threefold.

- We theoretically prove that the generalization error captures four main factors influencing the samples' learning difficulty, indicating that it can be used as a universal difficulty measure.
- We reveal how the difficulty-based sample weighting influences the optimization dynamics and generalization performance for deep learning. Our results indicate that assigning high weights to hard samples may accelerate the convergence. To enhance the generalization performance, a tradeoff between increasing the weights of certain samples and keeping the test and the weighted training distributions close should be found.
- We bring to light the characteristics of a good set of weights from multiple perspectives, illuminating the deep understanding of numerous weighting strategies.

2 Preliminaries

2.1 Description of Symbols

Let $\mathcal{X}$ denote the input space and $\mathcal{Y}$ a set of classes. We assume that the training and test samples are drawn *i.i.d* according to some distributions $\mathcal{D}^{tr}$ and $\mathcal{D}^{te}$ over $\mathcal{X} \times \mathcal{Y}$. The training set is denoted as $T = \{\boldsymbol{x}, y\} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{n}$ that contains n training samples, where $\boldsymbol{x}_i$ denotes the i-th sample's feature, and y_i is the associated label. Let d_i and $w(d_i)$ be the learning difficulty and the difficulty-based weight of $\boldsymbol{x}_i$. The learning difficulty can be approximated by several values, such as loss, margin, and generalization error which will be explained in detail in Section 3.

The predictor is denoted by $f(\boldsymbol{\theta}, \boldsymbol{x})$ and $\mathcal{F} = \{f(\boldsymbol{\theta}, \cdot) | \boldsymbol{\theta} \in \boldsymbol{\Theta} \subset \mathbb{R}^{\mathrm{d}}\}$. For the sake of notation, we focus on the binary setting $y_i \in \{-1, 1\}$ with $f(\boldsymbol{\theta}, \boldsymbol{x}) \in \mathbb{R}$. However, as to be clarified later, our results can be easily extended to the multi-class

setting where $y_i \in \{1, 2, \cdots, C\}$. For the multi-class setting, we extend our setup using the softmax function where the logits are now given by $\{f_{y_j}(\boldsymbol{\theta}, \boldsymbol{x})\}_{j=1}^{C}$. Given a non-negative loss ℓ and a classifier $f(\boldsymbol{\theta}, \cdot)$, the empirical risk can be expressed as $\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{w}) = \frac{1}{n}\sum_{i=1}^{n} w(d_i)\,\ell(y_i, f(\boldsymbol{\theta}, \boldsymbol{x}_i))$. We focus particularly on the exponential loss $\ell(u) = \exp(-u)$ and logistic loss $\ell(u) = \log(1 + \exp(-u))$. Let $\nabla l(u)$ be the loss gradient and $f(\boldsymbol{x}|T)$ be the trained model on T. The margin is denoted as $\gamma_{i,T}(\boldsymbol{x}_i) = y_i f(\boldsymbol{\theta}, \boldsymbol{x}_i|T)$ for the binary setting, where it is equivalently denoted as $\gamma_{i,T}(\boldsymbol{x}_i) = f_{y_i}(\boldsymbol{\theta}, \boldsymbol{x}_i|T) - \max_{i \neq j} f_{y_j}(\boldsymbol{\theta}, \boldsymbol{x}_i|T)$ for the multi-class setting.

2.2 Definition of Generalization Error

Bias-variance tradeoff is a basic theory for the qualitative analysis of the generalization error [21]. This tradeoff is initially constructed via regression and mean square error, which is given by

$$\begin{aligned} Err &= \mathbb{E}_{\boldsymbol{x},y}\mathbb{E}_T[||y - f(\boldsymbol{x}|T)||_2^2] \\ &\approx \underbrace{\mathbb{E}_{\boldsymbol{x},y}[||y - \overline{f}(\boldsymbol{x})||_2^2]}_{Bias} + \underbrace{\mathbb{E}_{\boldsymbol{x},y}\mathbb{E}_T[||f(\boldsymbol{x}|T) - \overline{f}(\boldsymbol{x})||_2^2]}_{Variance}, \end{aligned} \tag{1}$$

where $\overline{f}(\boldsymbol{x}) = \mathbb{E}_T[f(\boldsymbol{x}|T)]$. Similarly, we define the generalization error of a single sample $\boldsymbol{x}_i$ as

$$\mathrm{err}_i = \mathbb{E}_T\left[\ell(f(\boldsymbol{x}_i|T), y_i)\right] \approx B(\boldsymbol{x}_i) + V(\boldsymbol{x}_i), \tag{2}$$

where $B(\boldsymbol{x}_i)$ and $V(\boldsymbol{x}_i)$ are the bias and variance of $\boldsymbol{x}_i$.

2.3 Conditions and Definitions

Our theoretical analyses rely on the understanding of implicit bias of gradient descent in deep learning. The gradient descent process is denoted as

$$\boldsymbol{\theta}_{t+1}(\boldsymbol{w}) = \boldsymbol{\theta}_t(\boldsymbol{w}) - \eta_t \nabla\mathcal{L}(\boldsymbol{\theta}_t[\boldsymbol{w}(\boldsymbol{d}[t])]), \tag{3}$$

where η_t is the learning rate which can be a constant or step-independent, $\nabla\mathcal{L}(\boldsymbol{\theta}_t[\boldsymbol{w}(\boldsymbol{d}[t])])$ is the gradient of $\mathcal{L}$, and $\boldsymbol{w}(\boldsymbol{d}[t])$ is the difficulty-based weight of difficulty $\boldsymbol{d}$ at time t. The weight may be dynamic with respect to time t if difficulty measures, such as loss [3] and predicted probability [4], are used. To guarantee the convergence of the gradient descent, two conditions following the most recent study [13] are shown below:

- The loss ℓ has an exponential tail whose definition is shown in Section A.1 in the supplementary file. Thus, $\lim_{u\to\infty}\ell(-u) = \lim_{u\to\infty}\nabla\ell(-u) = 0$.
- The predictor $f(\boldsymbol{\theta}, \boldsymbol{x})$ is α-homogeneous such that $f(c\cdot\boldsymbol{\theta}, \boldsymbol{x}) = c^\alpha f(\boldsymbol{\theta}, \boldsymbol{x}), \forall c > 0$.

It is easy to verify that losses, including the exponential loss, log loss, and cross-entropy loss, satisfy the first condition. The second condition implies that the activation functions are homogeneous such as ReLU and LeakyReLU, and bias terms are disallowed. In addition, we need certain regularities from $f(\boldsymbol{\theta}, \boldsymbol{x})$ to ensure the existence of critical points and the convergence of gradient descent.

- For $\forall \boldsymbol{x} \in \mathcal{X}$, $f(\boldsymbol{\theta}, \boldsymbol{x})$ is β-smooth and l-Lipschitz on $\mathbb{R}^{\mathrm{d}}$.

The above condition is a common technical assumption whose practical implications are discussed in Section A.2 in the supplementary file.

The generalization performance of deep learning models is measured by the generalization error of the test set, which is

$$\hat{\mathcal{L}}(f) = \mathbb{P}_{(\boldsymbol{x},y)\sim\mathcal{D}^{te}}\left[\gamma(\boldsymbol{x}) \leq 0\right]. \tag{4}$$

2.4 Experiment Setup

Illustrated experiments are performed to support or illuminate our theoretical analyses. For the simulated data (shown in Fig. S-8 in the supplementary file), the linear predictor is a regular regression model. The nonlinear predictor is a two-layer MLP with five hidden units and ReLU as the activation function. Exponential loss and standard normal initialization are utilized. CIFAR10 [23] is experimented with, and ResNet32 [24] is adopted as the baseline model. For the imbalanced data, the imbalance setting follows Ref. [10]. For the noisy data, symmetric and pair-flip label noises are used, and the noise setting follows Ref. [22]. The models are trained with a gradient descent by using 0.1 as the learning rate. The model uncertainty is approximated by the predictive variance of five predictions. To approximate the generalization error, we adopt the five-fold cross-validation method [25] to calculate the average learning error for each sample.

3 A Universal Difficulty Measure

As previously stated, four main factors pointed out by existing studies, namely, noise, imbalance, margin, and uncertainty, greatly impact the learning difficulty of samples. Nevertheless, existing measures [10,17,18] only consider one or part of them, and their conclusions are based on heuristic inspirations or empirical observations rather than theoretical certifications. We theoretically prove that the generalization error of a sample can be used as a universal difficulty measure capturing all four factors. Although generalization error is a well-established concept, this is the first time the relationship between generalization error and the four factors is built with formal theories. All proofs are presented in Section B in the supplementary file. Without increasing the ambiguity, the generalization error of samples is termed as error for brevity.

3.1 Noise Factor

Noise refers to data that is inaccurate in describing the scene [26,27]. Numerous studies devoted to reducing the influence of noisy samples on deep learning models and they intuitively consider noisy samples as hard ones without theoretical certification [6,28]. There are two kinds of noise, namely, feature noise [26] and label noise [27]. We offer two propositions to reveal the relationship between the generalization error and the noise factor. For feature noise, we offer the following proposition:

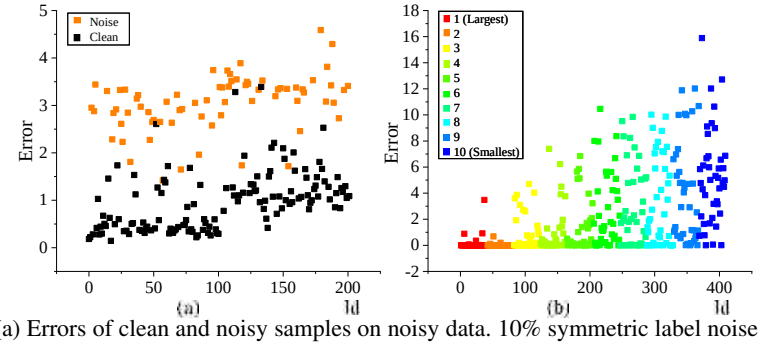

Fig. 1: (a) Errors of clean and noisy samples on noisy data. 10% symmetric label noise is added. (b) Errors of samples in ten categories on imbalanced data. The imbalance ratio is 10:1. Fifty samples are randomly selected to display for each category. CIFAR10 and ResNet32 are used. Other noise and imbalance settings following Ref. [22] are also experimented with, and the same conclusions are obtained.

Proposition 1. *Let $\Delta \boldsymbol{x}_i$ be the perturbation of sample $(\boldsymbol{x}_i, y_i)$, which is extremely small in that $o(\Delta \boldsymbol{x}_i)$ can be omitted. Let $\angle\varphi$ be the angle between the direction of $\Delta \boldsymbol{x}_i$ and the direction of $\mathbb{E}_T[f'(\boldsymbol{x}_i|T)]$. If $\mathbb{E}_T[f'(\boldsymbol{x}_i|T) \cdot \Delta \boldsymbol{x}_i] < 0$ (i.e., $\angle\varphi > 90°$), then the error of the noisy sample is increased relative to the clean one. Alternatively, the direction of the perturbation $\Delta \boldsymbol{x}_i$ and that of $\mathbb{E}_T[f'(\boldsymbol{x}_i|T)]$ are contradictory. Otherwise, if $\mathbb{E}_T[f'(\boldsymbol{x}_i|T) \cdot \Delta \boldsymbol{x}_i] > 0$, then $\angle\varphi < 90°$, and the error of the noisy sample is decreased.*

According to Proposition 1, feature noise can be divided into two categories. Noise increasing the error is called the adversarial type in this paper, which is frequently used in adversarial learning [29]; otherwise, it is a promoted type, referring to noise that decreases the errors of samples. Therefore, the variation of the error under feature noise depends on the noise type. For label noise, we offer the following proposition:

Proposition 2. *Let π be the label corruption rate (i.e., the probability of each label flipping to another one). Denote the predicted probability of the ground truth for the original sample as p. If $p > 0.5$, then the error of the noisy sample is larger than that of the clean one.*

This proposition implies that the errors of the samples with label noises are larger than those of the clean ones on average. Specifically, if a sample is more likely to be predicted correctly, its generalization error increases due to label noise. Let $\mathcal{L}^*$ be the global optimum of the generalization error of the clean dataset and y' be the corrupted label. When the noise in Proposition 2 is added, the empirical error is $\mathcal{L}' = (1-\pi)\,\mathcal{L}^* + \pi\mathcal{L}\left(f\left(\boldsymbol{x}\right), y'\right)$, where we have taken expectations over the noise. When $\pi \to 0$, the noise disappears, and the optimal generalization is attained. Proposition 2 is consistent with the empirical observation shown in Fig. 1(a), where the noisy samples have larger errors than the clean ones on average. In summary, the error embodies the noise factor.

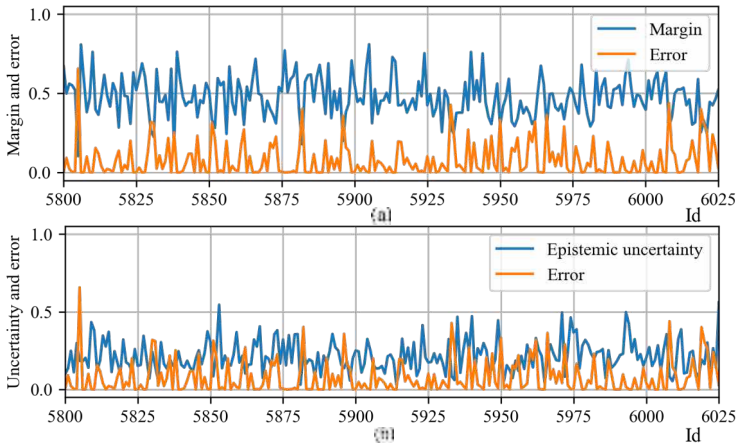

Fig. 2: (a) Correlation between error and the average margin. (b) Correlation between error and epistemic uncertainty. CIFAR10 and ResNet32 are used. All values are normalized.

3.2 Imbalance Factor

Besides noise, imbalance is another common deviation of real-world datasets. The category distribution of the samples in the training set is non-uniform. Various methods in deep learning solve this issue by assigning high weights to samples in tail categories (i.e., categories with a small number of samples), which are intuitively considered to be hard ones [4,10]. However, a theoretical justification about why these samples are hard lacks. We offer the following proposition.

Proposition 3. *If a predictor on an imbalanced dataset (the imbalance ratio $c_r > e:1$) is an approximate Bayesian optimal classifier (as the exponential loss is an approximation for the zero-one loss), which is to minimize the total risk, then the average probability of the ground truth of the samples in the large category is greater than that of the samples in the small category.*

The imbalance ratio is denoted by $c_r = \max\{\mathbb{c}_1, \mathbb{c}_2, \cdots, \mathbb{c}_C\} : \min\{\mathbb{c}_1, \mathbb{c}_2, \cdots, \mathbb{c}_C\}$ and $\mathbb{c}_k$ refers to the number of samples in the k-th category. With Proposition 3, it is easy to obtain Proposition S.1 shown in the supplementary file that the average error of the samples in the small category is larger than that of the samples in the large category, indicating that error captures the imbalance factor. This proposition is verified by experiments, as shown in Fig. 1(b). Tail categories contain a higher proportion of samples with larger errors. Therefore, samples with larger errors should be assigned with higher weights. Further experiments shown in Fig. 6 illustrate that the classification performance of the small category is improved by increasing its sample weights.

3.3 Margin Factor

The samples' margins measure the distances of the samples from the decision boundary [17]. Some studies in deep learning intuitively consider a small margin indicates a large learning difficulty and corresponds to low confidence of the prediction [13,17].

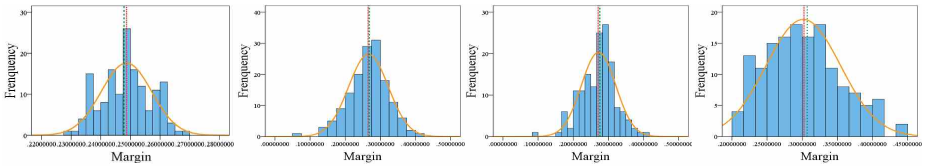

Fig. 3: The distributions of margins. Red and green lines refer to the mean of the distribution and the true margin.

However, there lacks a rigorous characterization of the relationship between the learning difficulty and margin. We offer the following proposition.

Proposition 4. *Let μ_i be the true margin of sample $\boldsymbol{x}_i$ corresponding to the oracle decision boundary. The condition is that the functional margins of a sample trained on random datasets obey a Gaussian distribution. In other words, for sample $\boldsymbol{x}_i$, its functional margin γ_i obey a Gaussian distribution $\mathcal{N}(\mu_i, \sigma_i^2)$. For sample $\boldsymbol{x}_j$, $\gamma_j \sim \mathcal{N}(\mu_j, \sigma_j^2)$. When the margin variances of the two samples are the same (i.e., $\sigma_i^2 = \sigma_j^2$), if $\mu_i \leq \mu_j$, then $\mathrm{err}_i \geq \mathrm{err}_j$. When the true margins of the two samples are the same (i.e., $\mu_i = \mu_j$), if $\sigma_i^2 \geq \sigma_j^2$, then $\mathrm{err}_i \geq \mathrm{err}_j$.*

Proposition 4 indicates that the true margin of a sample and error are negatively correlated when the margin variances of the samples are equal. By contrast, the margin variance and error are positively correlated when the true margins are equal. This finding corrects the current wisdom. Even a sample with a large true margin, as long as its margin variance is large, it may also have a large learning difficulty. The conclusion in which samples close to the oracle decision boundary are hard ones [13] is not completely correct. Indeed, the relation between the margin and error of sample $\boldsymbol{x}_i$ conforms $\mathrm{err}_i = \mathbb{E}_T[e^{-\gamma_i(T)}] = \exp(-\mu_i + \frac{1}{2}\sigma_i^2)$. For two samples $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, if $\mu_i < \mu_j$ and $\sigma_i^2 < \sigma_j^2$, then we cannot judge whether err_i is greater than err_j. As shown in Fig. 2(a), the average margin and error are negatively correlated for most samples, but it is not absolute, which accords with the above analyses.

Proposition 4 is based on the condition that the functional margins of a sample trained on random datasets obey a Gaussian distribution. We verify this condition via the Z-scores of Kurtosis and Skewness [30]. The margin distributions are shown in Fig. 3. More margin distributions and all Z-scores are shown in the supplementary file. As all Z-scores are in $[-1.96, 1.96]$, the margin distribution obeys the Gaussian distribution under the confidence level of 95%. In addition, the mean of the distribution and the true margin are very close, demonstrating the rationality of the assumed condition.

3.4 Uncertainty Factor

Uncertainties [31] in deep learning are classified into two types. The first type is aleatoric uncertainty (data uncertainty), which is caused by the noise in the observation data. Its correlation with error has been discussed in Section 3.1. The second type is epistemic uncertainty (model uncertainty) [18], indicating the consistency of multiple predictions. We give the analysis of the relationship between error and epistemic uncertainty.

Let T be a training set, and let $P(\boldsymbol{\theta}|T)$ be the distribution of the training models based on T. The predictive variance $Var(f(\boldsymbol{x}_i|\boldsymbol{\theta}_1), \cdots, f(\boldsymbol{x}_i|\boldsymbol{\theta}_K))$ plus a precision

constant is a typical manner of estimating epistemic uncertainty [32,33]. Take the mean square loss as an example[2], the epistemic uncertainty is

$$\widehat{\mathrm{Var}}\left(\boldsymbol{x}_i\right) := \tau^{-1} + \frac{1}{|K|}\sum\nolimits_k f(\boldsymbol{x}_i|\boldsymbol{\theta}_k)^\intercal f(\boldsymbol{x}_i|\boldsymbol{\theta}_k) - \mathbb{E}[f(\boldsymbol{x}_i|\boldsymbol{\theta}_k)]^\intercal \mathbb{E}[f(\boldsymbol{x}_i|\boldsymbol{\theta}_k)], \tag{5}$$

where τ is a constant. The second term on the right side of Eq. (5) is the second raw moment of the predictive distribution and the third term is the square of the first moment. When $K \to \infty$ and the constant term is ignored, Eq. (5) becomes

$$\widehat{\mathrm{Var}}\left(\boldsymbol{x}_i\right) := \int_{\boldsymbol{\theta}} ||f(\boldsymbol{x}_i|\boldsymbol{\theta}) - \mathbb{E}[f(\boldsymbol{x}_i|\boldsymbol{\theta}_k)]||_2^2 dP(\boldsymbol{\theta}|T). \tag{6}$$

If $P(\boldsymbol{\theta}|T)$ is approximated by the distribution of learned models on random training sets conforming to the Gaussian distribution $\mathcal{N}(T, \delta I)$, Eq. (6) is exactly the variance term of the error defined in Eq. (2) when the mean square loss is utilized.

As the bias term in error can capture the aleatoric uncertainty and the variance term captures the epistemic uncertainty, the overall relationship between uncertainty and error is positively correlated. Nevertheless, the relationship between epistemic uncertainty and error is not simply positively or negatively correlated. For some samples with heavy noises, their epistemic uncertainties will be small as their predictions remain erroneous. However, their errors are large due to their large bias. This phenomenon is consistent with the experimental results shown in Fig. 2(b). Epistemic uncertainty and error are positively correlated for some samples, and the two variables are negatively correlated for others.

3.5 Discussion about Generalization Error

The commonly used difficulty measures, such as loss [3] and predicted probability [4], are mainly related to the bias term. Shin et al. [27] pointed out that using only loss as the measure cannot distinguish clean and noisy samples, especially for asymmetric label noise. There are also a few studies utilizing variance [34]. Agarwal et al. [35] applied the variance of gradient norms as the difficulty measure. Actually, the role of both variance and bias terms should not be underestimated when measuring samples' learning difficulty. Our theoretical analyses support that generalization error, including both the two terms, captures four main factors that influence samples' learning difficulty, revealing that generalization error is a generic measure.

Existing studies generally utilize the average learning error to approximate generalization error, which is calculated by the K-fold cross-validation method [25]. Using this method, experiments under four typical scenarios in deep learning are conducted, as shown in Section E in the supplementary file, empirically demonstrating the superiority of generalization error as a universal difficulty measure. More efficient error calculation algorithms are supposed to be proposed which will be our future work.

4 Role of Difficulty-Based Weighting

Based on our universal difficulty measure (i.e., generalization error), the impacts of difficulty-based weighting on optimization dynamics and generalization performance

[2] For other losses, other methods can be used to calculate the predictive variance [25].

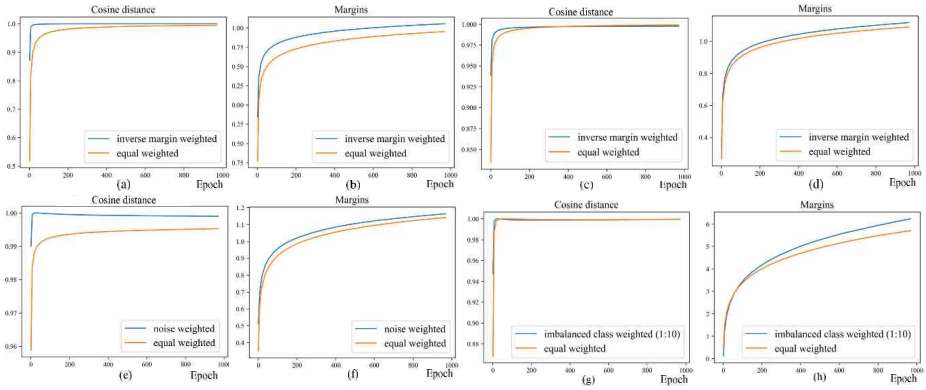

Fig. 4: "Cosine distance" means the cosine of the angle between the decision boundary (at that epoch) and the max-margin solution. (a), (b) Cosine distance and the average margin of equal weights and inverse margin weights using the linear predictor. (c), (d) Cosine distance and the average margin of equal weights and inverse margin weights using the nonlinear predictor. (e), (f) Cosine distance and the average margin of equal weights and increased weights for noisy samples using the linear predictor on noisy data. The noise ratio is 20%. (g), (h) Cosine distance and the average margin of equal weights and increased weights for samples in small categories using the nonlinear predictor on imbalanced data. More results are placed in the supplementary file.

in deep learning are investigated and well revealed. Compared with the most recent conclusions established only on the margin factor [13], our theoretical findings, which are based on our universal measure, are more precise and thus adapt to a wider range.

4.1 Effects on Optimization Dynamics

Linear Predictor We begin with the linear predictors allowing for a more refined analysis. The most recent study conducted by Xu et al. [13] inferred an upper bound of the convergence speed containing the term $D_{KL}(\boldsymbol{p}\|\boldsymbol{w})$, where D_{KL} is the Kullback-Leibler divergence [36] and $\boldsymbol{p}$ is the optimal dual coefficient vector which is a decreasing function of margin. A smaller value of $D_{KL}(\boldsymbol{p}\|\boldsymbol{w})$ means that the convergence may be accelerated. Therefore, they believe that the weights $\boldsymbol{w}$ should be consistent with $\boldsymbol{p}$. Alternatively, the samples with small functional margins will have large coefficients and thus should be assigned with large weights. However, samples with small margins may not be hard-to-classify, and the functional margin is not the true margin that corresponds to the oracle decision boundary. Therefore, their conclusion that samples close to the oracle decision boundary (hard-to-classify) should be assigned with large weights [13] cannot be well-drawn according to their inference.

We offer a more precise conclusion with our universal difficulty measure (i.e., generalization error). As before, we assume that the functional margin of a sample $\boldsymbol{x}_i$ obeys a Gaussian distribution $\mathcal{N}(\mu_i, \sigma_i^2)$, where μ_i and σ_i^2 are the true margin and margin variance. We offer the following proposition.

Proposition 5. *For two samples $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, if $\mathrm{err}_i \geq \mathrm{err}_j$, then we have:*

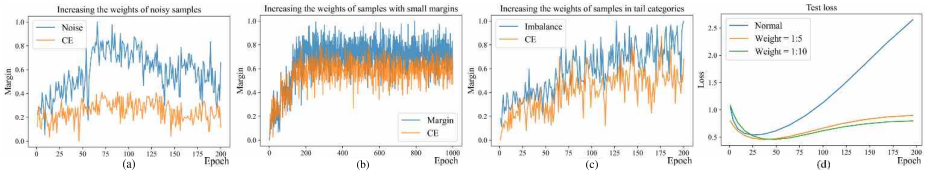

Fig. 5: (a)-(c) Normalized margin of increasing the weights of noisy samples/samples with small margins/samples in tail categories. CIFAR10 is used. 10% symmetric label noise is added. (d) Generalization error of the test set when the nonlinear model is trained with normal weights and increased weights for samples in small categories on simulated imbalanced data. The imbalance ratio is 10:1. The same conclusions can also be obtained on other noise and imbalance settings.

(1) When the optimal dual coefficient p_i of $\boldsymbol{x}_i$ on a random training set T is a linear function of its functional margin γ_i on T, if $\mu_i \leq \mu_j$, then $\mathbb{E}_T[p_i] \geq \mathbb{E}_T[p_j]$ (i.e., $\mathbb{E}_T[w_i] \geq \mathbb{E}_T[w_j]$);

(2) When the optimal dual coefficient p_i on a random training set T is a natural exponential function of its functional margin γ_i on T, $\mathbb{E}_T[p_i] \geq \mathbb{E}_T[p_j]$ (i.e., $\mathbb{E}_T[w_i] \geq \mathbb{E}_T[w_j]$) always holds. Notably, even when $\mu_i > \mu_j$, $\mathbb{E}_T[p_i] > \mathbb{E}_T[p_j]$ still holds.

The proof is presented in the supplementary file. $\mathbb{E}_T[p_i] > \mathbb{E}_T[p_j]$ implies that $w_i > w_j$ holds on average. The conclusion that samples with small true margins should be assigned with large weights [13] may not hold on some training sets when p_i is not a linear function of γ_i. A sample with a small true margin may have a smaller weight than a sample with a large true margin yet a large error. Thus, a more general conclusion when p_i is not a linear function of γ_i is that increasing the weights of hard samples (i.e., samples with large generalization errors) may accelerate the convergence, rather than just for samples with small margins. Other factors, including noise, imbalance, and uncertainty, also affect samples' learning difficulty. Notably, the weights of the hard samples should not be excessively increased, as to be explained in the succeeding section. We reasonably increase the weights of the hard samples shown in Figs. 4 and S-10 in the supplementary file indicating that the optimization is accelerated.

We also prove that the difficulty-based weights do not change the convergence direction to the max-margin solution shown in the supplementary file. As shown in Fig. 3, as training progresses, the cosine distance and margin both increase, indicating the direction of the asymptotic margin is that of the max-margin solution.

Nonlinear Predictor Analyzing the gradient dynamics of the nonlinear predictors is insurmountable. The main conclusion obtained by Xu et al. [13] can also be established for difficulty-based weights only if the bound of weights is larger than zero. However, their theorem has only been proven for binary cases as the employed loss is inapplicable in multi-class cases. We extend the theory to the multi-class setting with a regularization $\lambda||\boldsymbol{\theta}||^r$ on the cross-entropy loss denoted as $\mathcal{L}_\lambda$. Let $\boldsymbol{\theta}_\lambda(\boldsymbol{w}) \in \arg\min \mathcal{L}_\lambda(\boldsymbol{\theta}(\boldsymbol{w}))$. The dynamic regime for the nonlinear predictor can be described as follows:

Theorem 1. *Let $\boldsymbol{w} \in [b, B]^n$. Denote the optimal normalized margin as*

$$\gamma^* = \max_{\|\boldsymbol{\theta}(\boldsymbol{w})\| \leq 1} \min_i (f_{y_i}(\boldsymbol{\theta}(\boldsymbol{w}), \boldsymbol{x}_i) - \max_{j \neq i} (f_{y_j}(\boldsymbol{\theta}(\boldsymbol{w}), \boldsymbol{x}_i))). \tag{7}$$

Let $\overline{\boldsymbol{\theta}}_\lambda(\boldsymbol{w}) = \boldsymbol{\theta}_\lambda(\boldsymbol{w})/\|\boldsymbol{\theta}_\lambda(\boldsymbol{w})\|$. *Then, it holds that*

(1) Denote the normalized margin as

$$\gamma_\lambda(\boldsymbol{w}) = \min_i (f_{y_i}(\overline{\boldsymbol{\theta}}_\lambda(\boldsymbol{w}), \boldsymbol{x}_i) - \max_{j \neq i} f_{y_j}(\overline{\boldsymbol{\theta}}_\lambda(\boldsymbol{w}), \boldsymbol{x}_i)). \tag{8}$$

Then, $\gamma_\lambda(\boldsymbol{w}) \to \gamma^*$, *as* $\lambda \to 0$*;*

(2) There exists a $\lambda := \lambda(r, a, \gamma^*, \boldsymbol{w})$. *For* $\alpha \leq 2$, *let* $\boldsymbol{\theta}'(\boldsymbol{w})$ *denote a* α*-approximate minimizer of* $\mathcal{L}_\lambda$. *Thus,* $\mathcal{L}_\lambda(\boldsymbol{\theta}'(\boldsymbol{w})) \leq \alpha L_\lambda(\boldsymbol{\theta}_\lambda(\boldsymbol{w}))$. *Denote the normalized margin of* $\boldsymbol{\theta}'(\boldsymbol{w})$ *by* $\gamma'(\boldsymbol{w})$. *Then,*$\gamma'(\boldsymbol{w}) \geq \frac{\gamma^*}{10\alpha^{a/r}}$.

The proof is placed in the supplementary file. When λ is sufficiently small, the difficulty-based weighting does not affect the asymptotic margin. According to Theorem 1, the weights do affect the convergence speed. Even though $L_\lambda(\boldsymbol{\theta}_\lambda(\boldsymbol{w}))$ has not converged but close enough to its optimum, the corresponding normalized margin has a reasonable lower bound. A good set of weights can help the deep learning models achieve this property faster. However, the conditions in which a set of weights can accelerate optimization are not clearly illuminated. Notably, as shown in our experiments in Figs. 4 and S-10 in the supplementary file, assigning large weights for hard samples increases the convergence speed. The results on the multi-class cases (CIFAR10) indicate that assigning large weights on hard samples increases the margin, as shown in Figs. 5(a-c). However, some particular occasions of difficulty-based weights, such as Self-paced learning (SPL) [3], do not satisfy the bounding condition because the lower bounds of these weights are zero instead of a positive real number. This theorem requires further revision to accommodate this situation.

4.2 Effects on Generalization Performance

Besides optimization dynamics, we are concerned as to whether and how the difficulty-based weights affect the generalization performance. The generalization bound of Xu et al. [13] only considers importance weighting which is fixed. Thus it cannot explain why different weighting strategies are effective. In addition, they assume that the source and target distributions are unequal, restricting the application scope of their conclusion. The two generalization bounds we propose offer good solutions to these issues, illuminating how weighting strategies can be well designed and explained.

Let P_s and P_t be the source (training) and target (testing) distributions, with the corresponding densities of $p_s(\cdot)$ and $p_t(\cdot)$. Assume that the two distributions have the same support. The training and test samples are drawn *i.i.d* according to distributions P_s and P_t, respectively. Learning with sample weights $\boldsymbol{w}(\boldsymbol{x})$ is equivalent to learning with a new training distribution $\widetilde{P}_s$. The density of the distribution of the weighted training set $\widetilde{P}_s$ is denoted as $\widetilde{p}_s(\boldsymbol{x})$ and $\widetilde{p}_s(\boldsymbol{x}) \sim \boldsymbol{w}(\boldsymbol{x}) p_s(\boldsymbol{x})$. Pearson χ^2-divergence is used to measure the difference between $\widetilde{P}_s$ and P_t, i.e., $D_{\chi^2}(P_t \| \widetilde{P}_s) = \int [(d\widetilde{P}_s/dP_t)^2 - 1] d\widetilde{P}_s$. We consider depth-$q$ ($q \geq 2$) networks with the activation function ϕ. The binary setting is considered, in that the network computes a real value

$$f(\boldsymbol{x}) := \boldsymbol{W}_q \phi(\boldsymbol{W}_{q-1} \phi(\cdots \phi(\boldsymbol{W}_1 \boldsymbol{x}) \cdots)), \tag{9}$$

where $\phi(\cdot)$ is the element-wise activation function (e.g., ReLU). The training set contains n samples. Denote the generalization error for a network f as $\hat{\mathcal{L}}(f)$. The generalization performance of f with weights can be described as follows:

Theorem 2. *Suppose ϕ is 1-Lipschitz and 1-positive homogeneous. With a probability at least of $1-\delta$, we have*

$$\hat{\mathcal{L}}(f) \leq \underbrace{\frac{1}{n}\sum_{i=1}^{n}\frac{p_t(\boldsymbol{x}_i)}{\widetilde{p}_s(\boldsymbol{x}_i)}\mathbb{1}(y_i f(\boldsymbol{x}_i)<\gamma)}_{(I)} + \underbrace{\frac{L\cdot\sqrt{D_{\chi^2}\left(P_t\|\widetilde{P}_s\right)+1}}{\gamma\cdot q^{(q-1)/2}\sqrt{n}}}_{(II)} + \underbrace{\epsilon(\gamma,n,\delta)}_{(III)}, \tag{10}$$

where $\epsilon(\gamma,n,\delta)=\sqrt{\frac{\log\log_2\frac{4L}{\gamma}}{n}}+\sqrt{\frac{\log(1/\delta)}{n}}$ *and* $L:=\sup_{\boldsymbol{x}}\|\boldsymbol{x}\|$.

The proof is presented in the supplementary file. Compared with the findings of Xu et al. [13], the generalization bound we propose is directly related to the sample weights $\boldsymbol{w}(\boldsymbol{x})$ contained in $\widetilde{p}_s(\boldsymbol{x})$. In view of reducing the generalization error, a natural optimization strategy can be implemented as follows: 1) an optimal weight set $\boldsymbol{w}(\boldsymbol{x})$ (in $\widetilde{p}_s(\boldsymbol{x})$) is obtained according to decreasing the right side of Eq. (10) based on the current f; 2) f is then optimized under the new optimal weights $\boldsymbol{w}(\boldsymbol{x})$. Disappointingly, this strategy heavily relies on the current f, which is unstable. Given a fixed training set, f depends on random variables (denoted as $\mathcal{V}$) such as hyperparameters and initialization. To obtain a more stable weighting strategy, we propose the following proposition:

Proposition 6. *Suppose ϕ is 1-Lipschitz and 1-positive homogeneous. With a probability of at least $1-\delta$, we have*

$$\mathbb{E}_{\mathcal{V}}[\hat{\mathcal{L}}(f_{\mathcal{V}})] \leq \underbrace{\frac{1}{n}\sum_{i=1}^{n}\frac{p_t(\boldsymbol{x}_i)}{\widetilde{p}_s(\boldsymbol{x}_i)}\mathbb{E}_{\mathcal{V}}[\mathbb{1}(y_i f_{\mathcal{V}}(\boldsymbol{x}_i)<\gamma)]}_{(I)} + \underbrace{\frac{L\cdot\sqrt{D_{\chi^2}\left(P_t\|\widetilde{P}_s\right)+1}}{\gamma\cdot q^{(q-1)/2}\sqrt{n}}}_{(II)} + (III). \tag{11}$$

Accordingly, to decrease the generalization bound, (I) and (II) of Eq. (11) are supposed to be decreased. Samples with larger generalization errors will have larger values of $\mathbb{E}_{\mathcal{V}}[\mathbb{1}(y_i f_{\mathcal{V}}(\boldsymbol{x}_i)<\gamma)]$. The proof is placed in the supplementary file. Given that the optimal weight set (or the optimal $\widetilde{p}_s(\boldsymbol{x})$) should minimize the sum of (I) and (II), the following insights can be obtained:

- Conventional importance weighting only guarantees that (II) of Eq. (11) attains its minimum rather than the sum of (I) and (II). Consequently, although importance weighting is prevalent in previous studies, it may not be the ideal strategy.
- Increasing the weights of hard samples (i.e., samples with large $\mathbb{E}_{\mathcal{V}}[\mathbb{1}(y_i f_{\mathcal{V}}(\boldsymbol{x}_i)<\gamma)]$) will reduce the impact of these samples on (I) and thus may decrease (I). In this case, if $D_{\chi^2}(P_t\|\widetilde{P}_s)$ in (II) also decreases or its increase is relatively small, the value of $(I)+(II)$ will decrease, indicating that assigning large weights on hard samples (i.e., hard-first mode) takes effect, as shown in Fig. 5(d). The weights of hard samples cannot be increased arbitrarily as $D_{\chi^2}(P_t\|\widetilde{P}_s)$ may be large.
- When there are noises in the training set, the values of p_t for noisy samples are zero. Therefore, reducing the values of $\widetilde{p}_s$ (or sample weights) of noisy samples does not increase the generalization bound. Meanwhile, the values of $\widetilde{p}_s$ for clean samples will increase as $\sum_i \widetilde{p}_s(\boldsymbol{x}_i)=1$. Consequently, (I) will decrease. As decreasing the weights of noise samples makes P_t and $\widetilde{P}_s$ close, (II) will also decrease. Thus,

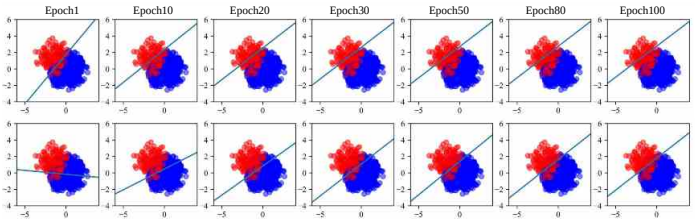

Fig. 6: Top: Equal weights of two categories. Bottom: Samples in the small (red) category are assigned with high weights, obtaining better performance for the small category. The imbalance ratio is 10:1. The same conclusion can also be obtained for other imbalance ratios.

increasing the weights of easy samples (i.e., easy-first mode) is suitable for noisy data, as shown in Figs. S-12 and S-13 in the supplementary file. This conclusion is in accordance with the observations from an empirical study conducted by Wu et al. [37] that the easy-first paradigm curriculum learning mainly takes effect in noisy scenarios.

– Zhou et al. [1] did not exhibit how to pursue the optimal difficulty distribution of the training samples. Proposition 6 offers a reasonable solution to this issue. Specifically, the optimal weight $\boldsymbol{w}^*$ can be obtained by minimizing the generalization bound (i.e., attaining the tradeoff between (I) and (II)). Meanwhile, the difficulty distribution corresponding to $\boldsymbol{w}^*$ is the optimal difficulty distribution.

It is worth mentioning that our conclusions are still insightful when $P_t = P_s$ while the conclusion of Xu et al. [13] assumes $P_t \neq P_s$. Apparently, even when $P_t = P_s$, assigning weights according to samples' learning difficulties is also beneficial as the tradeoff between (I) and (II) still takes effect. An illustrative example with a brief theoretical analysis is presented in the supplementary file to show how the weights of the easy and hard samples affect the sum of (I) and (II).

5 Discussion

Our theoretical analyses in Sections 3 and 4 provide reasonable answers to the concerns described in Section 1.

First, the generalization error has been theoretically guaranteed as a generic difficulty measure. It is highly related to noise level, imbalance degree, margin, and uncertainty, which significantly affect samples' learning difficulty. Consequently, two directions are worth further investigating. The first direction pertains to investigating a more efficient estimation method for the generalization error, enhancing its practicality. This will be our future work. As for the second direction, numerous existing and new weighting schemes can be improved or proposed using the generalization error as the difficulty measure. Our theoretical findings supplement or even correct the current understanding. For example, samples with large margins may also be hard-to-classify if they have large margin variances (e.g., with heterogeneous samples in their neighbors).

Second, the existing conclusions on convergence speed have been extended. For linear predictors, the existing conclusion is extended by considering our difficulty measure, namely, the generalization error. For the nonlinear predictors, the conclusion is

extended into multi-class cases. Furthermore, the explicit relationship between the generalization gap and sample weights has been established. Our theorem indicates that assigning weights on training samples according to the learning difficulty is also effective even when the source P_s and target distributions P_t are equal.

Our theoretical findings of the generalization bounds provide better explanations for existing weighting schemes. As discussed before, if heavy noise exists in the dataset, then the weights of the noisy samples should be decreased to better match the source and target distributions. The experiments on noisy data are shown in Figs. S-12 and S-13 in the supplementary file in which decreasing the weights of noisy samples obtains the best performance. In imbalanced learning, samples in small categories have higher generalization errors on average. Increasing the weights of the hard samples will not only accelerate the optimization but also improve the performance of the small category, as shown in Figs. 5(d) and 6. These high-level intuitions justify a number of difficulty-based weighting methods. Easy-first schemes, such as SPL [3], Superloss [6], and Curriculum learning [9], perform well on noisy data. Hard-first schemes, such as Focal Loss [4], Class-balance [10], and G-RW [16], are suitable for imbalanced data.

6 Conclusion

This study has theoretically investigated difficulty-based sample weighting. First, the generalization error is verified as a universal difficulty measure that can reflect four main factors influencing the learning difficulty of samples. Second, on the basis of the universal difficulty measure, the role of the difficulty-based weighting strategy for deep learning is characterized in terms of convergence dynamics and generalization performance. Theoretical findings are also presented. Increasing the weights of the hard samples may accelerate the optimization. A good set of weights should attain the trade-off between assigning large weights on certain samples and keeping the test and the weighted training distributions close. These findings enlighten the deep understanding and design of existing and future weighting schemes.

References

1. Zhou, X., Wu, O.: Which Samples Should be Learned First: Easy or Hard? arXiv preprint arXiv:2110.05481 (2021)
2. Khan, S.-H., Hayat, M., Bennamoun, M., Sohel, F.-A., Togneri, R.: Cost-sensitive learning of deep feature representations from imbalanced data. IEEE Transactions on Neural Networks and Learning Systems **29**(8), 3573–3587 (2018)
3. Kuma, M.-P., Packer, B., Koller, D.: Self-paced learning for latent variable models. In: NeurIPS, pp. 1–9. Curran Associates, America (2010)
4. Lin, T.-Y., Goyal, P., Girshick, R., He, K., Dollar, P.: Focal Loss for Dense Object Detection. IEEE Transactions on Pattern Analysis and Machine Intelligence **42**(2), 318–327 (2020)
5. Bengio, Y., Louradour, J., Collobert, R., Weston, J.: Curriculum learning. In: ICML, pp. 41–48. Association for Computing Machinery, America (2009)
6. Castells, T., Weinzaepfel, P., Revaud, J.: SuperLoss: A generic loss for robust curriculum learning. In: NeurIPS, pp. 1–12. NeurIPS foundation, America (2020)

7. Arpit, D., Jastrzębski, S., Ballas, N., Krueger, D., Bengio, E., Kanwal, M.-S., Maharaj, T., Fischer, A., Courville, A., Benglo, Y., Simon, L.-J.: A closer look at memorization in deep networks. In: ICML, pp. 350–359. IMLS, America (2017)
8. Wang, W., Feng, F., He, X., Nie, L., Chua, T.-S.: Denoising Implicit Feedback for Recommendation. In: WSDM, pp. 373–381. Association for Computing Machinery, America (2021)
9. Bengio, Y., Louradour, J., Collobert, R., Weston, J.: Curriculum learning. In: ICML, pp. 41–48. Association for Computing Machinery, America (2009)
10. Cui, Y., Jia, M., Lin, T.-Y., Song, Y., Belongie, S.: Class-Balanced Loss Based on Effective Number of Samples. In: CVPR, pp. 9260–9269. IEEE, America (2019)
11. Santiago, C., Barata, C., Sasdelli, M., Carneiro, G., Nascimento, J.-C.: LOW: Training deep neural networks by learning optimal sample weights. Pattern Recognit **110**(1), 107585 (2021)
12. Byrd, J., Lipton, Z.-C.: What is the effect of Importance Weighting in Deep Learning? In: ICML, pp. 1405–1419. Springer, Germany (2019)
13. Xu, D., Ye, Y., Ruan, C.: Understanding the role of importance weighting for deep learning. In: ICLR, pp. 1–20. ICLR foundation, America (2020)
14. Emanuel, B.-B., Tal, R., Nadav, Z., Asaf, N., Itamar, F., Matan, P., Lihi, Z.-M.: Asymmetric Loss For Multi-Label Classification. arXiv preprint arXiv:2009.14119 (2020)
15. Li, B., Liu, Y., Wang, X.: Gradient Harmonized Single-stage Detector. In: AAAI, pp. 8577–8584. AAAI Press, America (2019)
16. Zhang, S., Li, Z., Yan, S., He, X., Sun, J.: Distribution Alignment: A Unified Framework for Long-tail Visual Recognition. In: CVPR, pp. 2361–2370. IEEE, America (2021)
17. Zhang, J., Zhu, J., Niu, G., Han, B., Sugiyama, M., Kankanhalli, M.: Geometry-aware Instance-reweighted Adversarial Training. In: ICLR, pp. 1–29. America (2021)
18. Aguilar, E., Nagarajan, B., Khatun, R., Bolaños, M., Radeva, P.: Uncertainty modeling and deep learning applied to food image analysis. In: ICBM, pp. 3–16. Springer, Germany (2020)
19. Xiao, Y., Wang, W.-Y.: Quantifying uncertainties in natural language processing tasks. In: AAAI, pp. 7322–7329. AAAI Press, America (2019)
20. Soudry, D., Hoffer, E., Nacson, M.-S., Gunasekar, S., Srebro, N.: The implicit bias of gradient descent on separable data. Journal of Machine Learning Research **19**(1), 1–14 (2018)
21. Heskes, T.: Bias/Variance Decompositions for Likelihood-Based Estimators. Neural Computation **10**(6), 1425–1433 (1998)
22. Shu, J., Xie, Q., Yi, L., Zhao, Q., Zhou, S., Xu, Z., Meng, D.: Meta-weight-net: Learning an explicit mapping for sample weighting. In: NeurIPS, pp. 1–23. America (2019)
23. Alex, K., Hinton, G.: Learning multiple layers of features from tiny images. Handbook of Systemic Autoimmune **1**(4), 1–60 (2009)
24. He, K., Zhang, X., Ren S., Sun, J.: Deep Residual Learning for Image Recognition. In: CVPR, pp. 770–778. IEEE, America (2016)
25. Yang, Z., Yu, Y., You, C., Jacob, S., Yi, M.: Rethinking bias-variance trade-off for generalization of neural networks. In: ICML, pp. 10767–10777. IMLS, America (2020)
26. Wolterink, J.-M., Leiner, T., Viergever M.-A., Išgum, I.: Generative Adversarial Networks for Noise Reduction in Low-Dose CT. IEEE Trans Med Imaging **36**(12), 2536–2545 (2017)
27. Shin, W., Ha, J.-W., Li S., Cho, Y., Song, H.: Which Strategies Matter for Noisy Label Classification? Insight into Loss and Uncertainty. arXiv preprint arXiv:2008.06218 (2020)
28. Liu, E.-Z., Haghgoo, B., Chen, A.-S., Raghunathan, A., Koh, P.-W., Sagawa, S., Liang, P., Finn, C.: Just Train Twice: Improving Group Robustness without Training Group Information. arXiv preprint arXiv:2107.09044 (2021)
29. Lowd, D., Meek, C.: Adversarial learning. In: SIGKDD, pp. 641–647. Association for Computing Machinery, America (2005)
30. Ghasemi, A., Zahediasl, S.: Normality tests for statistical analysis: a guide for non-statisticians. International journal of endocrinology and metabolism **10**(2), 486–489 (2012)

31. Brando, A., Rodríguez-Serrano, J.-A., Ciprian, M., Maestre, R., Vitrià, J.: Uncertainty modelling in deep networks: Forecasting short and noisy series. In: ECML-PKDD, pp. 325–340. Springer, Germany (2019)
32. Gal, Y., Ghahramani, Z.: Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In: ICML, pp. 1050–1059. America (2016)
33. Abdar, M., Pourpanah, F., Hussain, S., Rezazadegan, D., Liu, L., Ghavamzadeh, M., Fieguth, P., Cao, X., Khosravi, A., Acharya, U.-R., Makarenkov, V., Nahavandi, S.: A review of uncertainty quantification in deep learning: Techniques, applications and challenges. Information Fusion **76**(1), 243–297 (2021)
34. Chang, H.-S., Erik, L.-M., McCallum A.: Active bias: Training more accurate neural networks by emphasizing high variance samples. In: NeurIPS, pp. 1003–1013. America (2017)
35. Agarwal, C., Hooker, S.: Estimating example difficulty using variance of gradients. arXiv preprint arXiv:2008.11600 (2020)
36. Erven, T.-.V., Harremos, P.: Rényi divergence and Kullback-Leibler divergence. IEEE Trans. Inform. Theory **60**(7), 3797–3820 (2014)
37. Wu, X., Dyer, E., Neyshabur, B.: When Do Curricula Work? In: ICLR, pp. 1–15. (2021)

Supplementary Materials for Understanding Difficulty-based Sample Weighting with a Universal Difficulty Measure

Xiaoling Zhou, Ou Wu, Weiyao Zhu, Ziyang Liang

Center of Applied Mathematics

A Supplementary Material for Section 2

A.1 Definition of the Exponential-tail Loss

Following Lyu and Li [1], there is a general definition of the exponential loss, where $\ell(u) = \exp(-f(u))$,

- f is smooth and $f'(u) \geq 0, \forall u$,
- there exists $c > 0$ such that $f'(u)\, u$ is non-decreasing for $u > c$ and $f'(u)\, u \to \infty$ as $u \to \infty$.

It is easy to verify that losses including the exponential loss, log loss, and cross-entropy loss satisfy the definition.

A.2 Practical Implications of the Third Condition

The third condition in Section 2.3 asserts the Lipschitz and smoothness properties. The Lipschitz condition is rather mild assumption for neural networks, and several recent paper are dedicated to obtaining the Lipschitz constant of certain deep learning models [2,3].

The β-smooth condition, on the other hand, is more technical-driven such that we can analyze the gradient descent. In practice, neural networks with ReLU activation do not satisfy the smoothness condition. However, there are smooth homogeneous activation functions, such as the quadratic activation $\sigma(x) = x^2$ and higher-order ReLU activation $\sigma(x) = \text{ReLU}(x)^c$ for $c > 2$. Still, in our experiments, we use ReLU as the activation function for its convenience.

B Supplementary Material for Section 3

In this section, we provide the omitted proofs and discussions of Section 3. Take the exponential loss $\ell = \exp(-y_i f(\boldsymbol{x}_i))$ as an example in the subsequent analyses. Let T be a random training set from some distributions over $\mathcal{X} \times \mathcal{Y}$ and let $f(\cdot|T)$ be the trained model on T. The generalization error of sample $\boldsymbol{x}_i$ is

$$\text{err}_i = \mathbb{E}_T\left[\ell\left(f\left(\boldsymbol{x}_i|T\right), y_i\right)\right] = \int_{T \in \mathcal{X}\times\mathcal{Y}} \exp(-y_i f\left(\boldsymbol{x}_i|T\right)) dP\left(T\right). \tag{A.1}$$

For the sake of notation, we focus on the binary setting $y \in \{-1, 1\}$. The positive samples are taken as examples in the succeeding discussion. It is easy to verify that our conclusions are also valid for multi-class setting and other loss unctions.

B.1 Proof of Proposition 1

In this section, we offer the proof of Proposition 1.

Proof. According to the definition of the generalization error, the error of a clean positive sample is

$$\mathrm{err}_i = \mathbb{E}_T \left[\ell \left(f\left(\boldsymbol{x}_i|T\right), y_i\right)\right] = \int_{T\in\mathcal{X}\times\mathcal{Y}} \exp(-f\left(\boldsymbol{x}_i|T\right))dP\left(T\right). \tag{A.2}$$

Denote the perturbation of $\boldsymbol{x}_i$ as $\Delta\boldsymbol{x}_i$, which is usually sufficient small. After adding feature noise, the sample becomes $\left(\boldsymbol{x}_i + \Delta\boldsymbol{x}_i, y_i = 1\right)$. Its error is

$$\mathrm{err}_{i'} = \mathbb{E}_T \left[\ell \left(f\left(\boldsymbol{x}_i + \Delta\boldsymbol{x}_i|T\right), y_i\right)\right] = \int_{T\in\mathcal{X}\times\mathcal{Y}} \exp(-f\left(\boldsymbol{x}_i + \Delta\boldsymbol{x}_i|T\right))dP\left(T\right). \tag{A.3}$$

Based on the definition of the exponential-tail loss, the Taylor expansion of f can be adopted. Let $f'(\boldsymbol{x}_i|T)$ denote the first-order derivative. As we mainly concern about the direction of $f'(\boldsymbol{x}_i|T)$ and the perturbation $\Delta\boldsymbol{x}_i$ is small, the first-order Taylor expansion can be adopted. Thus, we have

$$f\left(\boldsymbol{x}_i + \Delta\boldsymbol{x}_i|T\right) = f\left(\boldsymbol{x}_i|T\right) + f'\left(\boldsymbol{x}_i|T\right) \cdot \Delta\boldsymbol{x}_i + o\left(\Delta\boldsymbol{x}_i\right), \tag{A.4}$$

Here, $f(\boldsymbol{x}_i)$ is the output of the sigmoid layer, i.e., $f(\boldsymbol{x}_i) \in (0, 1)$. Applying the first-order Taylor expansion on $\exp(-x)$, we yield $\exp(-x) = 1 - x + R(x)$. Then, the generalization error turns to

$$\mathrm{err}_i = \mathbb{E}_T \left[\exp(-f\left(\boldsymbol{x}_i|T\right))\right] = \mathbb{E}_T \left[1 - f\left(\boldsymbol{x}_i|T\right) + R(f\left(\boldsymbol{x}_i|T\right))\right]. \tag{A.5}$$

After adding feature noise, its generalization error becomes

$$\begin{aligned} \mathrm{err}_{i'} &= \mathbb{E}_T \left[\exp(-f\left(\boldsymbol{x}_i + \Delta\boldsymbol{x}_i|T\right))\right] \\ &= \mathbb{E}_T \left[1 - f\left(\boldsymbol{x}_i + \Delta\boldsymbol{x}_i|T\right) + R(f\left(\boldsymbol{x}_i + \Delta\boldsymbol{x}_i|T\right))\right]. \end{aligned} \tag{A.6}$$

To compare the generalization error of the clean sample and the feature-noised sample, we separately study the first two terms and the residual term of the Taylor expansion shown in Formula (A.6).

$$\begin{aligned} &\mathrm{err}_i - \mathrm{err}_{i'} \\ =&\mathbb{E}_T[1 - f(\boldsymbol{x}_i|T) + R(f(\boldsymbol{x}_i|T))] \\ &- \mathbb{E}_T[1 - f(\boldsymbol{x}_i + \Delta\boldsymbol{x}_i|T) + R(f(\boldsymbol{x}_i + \Delta\boldsymbol{x}_i|T))] \\ =&\mathbb{E}_T[1 - f(\boldsymbol{x}_i|T) - 1 + f(\boldsymbol{x}_i + \Delta\boldsymbol{x}_i|T) \\ &+ R(f(\boldsymbol{x}_i|T)) - R(f(\boldsymbol{x}_i + \Delta\boldsymbol{x}_i|T))] \\ =&\mathbb{E}_T[1 - f(\boldsymbol{x}_i|T) - 1 + f(\boldsymbol{x}_i + \Delta\boldsymbol{x}_i|T)] \\ &+ \mathbb{E}_T[R(f(\boldsymbol{x}_i|T)) - R(f(\boldsymbol{x}_i + \Delta\boldsymbol{x}_i|T))]. \end{aligned} \tag{A.7}$$

For the first two terms,

$$\begin{aligned}&\mathbb{E}_T[1-f\left(\boldsymbol{x}_i|T\right)-1+f\left(\boldsymbol{x}_i+\Delta\boldsymbol{x}_i|T\right)]\\&=\mathbb{E}_T[1-f\left(\boldsymbol{x}_i|T\right)-1+f\left(\boldsymbol{x}_i|T\right)+f'\left(\boldsymbol{x}_i|T\right)\cdot\Delta\boldsymbol{x}_i+o(\Delta\boldsymbol{x}_i)]\\&=\mathbb{E}_T[f'\left(\boldsymbol{x}_i|T\right)\cdot\Delta\boldsymbol{x}_i+o(\Delta\boldsymbol{x}_i)].\end{aligned}\tag{A.8}$$

Comparing the residual term $R(x)=\exp(-x)+x-1$,

$$\begin{aligned}&\mathbb{E}_T[R(f(\boldsymbol{x}_i|T))-R(f(\boldsymbol{x}_i+\Delta\boldsymbol{x}_i|T))]\\&=\mathbb{E}_T[\exp(-f\left(\boldsymbol{x}_i|T\right))+f\left(\boldsymbol{x}_i|T\right)]\\&-\mathbb{E}_T[\exp(-f\left(\boldsymbol{x}_i+\Delta\boldsymbol{x}_i|T\right))+f\left(\boldsymbol{x}_i+\Delta\boldsymbol{x}_i|T\right)]\\&=\mathbb{E}_T[-f'\left(\boldsymbol{x}_i|T\right)\Delta\boldsymbol{x}_i-o(\Delta\boldsymbol{x}_i)]\\&+\mathbb{E}_T[\exp(-f\left(\boldsymbol{x}_i|T\right))-\exp(-f\left(\boldsymbol{x}_i+\Delta\boldsymbol{x}_i|T\right))].\end{aligned}\tag{A.9}$$

When $x\in(0,1)$, considering the relationship between the two functions $y=\exp(-x)$ and $y=-x+1+\frac{1}{e}$, $\exp(-f(\boldsymbol{x}_i|T))-\exp(-f(\boldsymbol{x}_i+\Delta\boldsymbol{x}_i|T))$ can be bounded.

For the upper bound, when $f(\boldsymbol{x}_i+\Delta\boldsymbol{x}_i|T)\geq f(\boldsymbol{x}_i|T)$,

$$\exp(-f(\boldsymbol{x}_i|T))-\exp(-f(\boldsymbol{x}_i+\Delta\boldsymbol{x}_i|T))\quad\leq f'\left(\boldsymbol{x}_i|T\right)\cdot\Delta\boldsymbol{x}_i+o(\Delta\boldsymbol{x}_i);\tag{A.10}$$

otherwise, when $f(\boldsymbol{x}_i+\Delta\boldsymbol{x}_i|T)<f(\boldsymbol{x}_i|T)$,

$$\exp(-f(\boldsymbol{x}_i|T))-\exp(-f(\boldsymbol{x}_i+\Delta\boldsymbol{x}_i|T))<0+o(\Delta\boldsymbol{x}_i).\tag{A.11}$$

Thus, we yield

$$e\mathrm{rr}_i-e\mathrm{rr}_{i'}\leq\mathcal{C}_1\mathbb{E}_T[f'(\boldsymbol{x}_i|T)\cdot\Delta\boldsymbol{x}_i+o(\Delta\boldsymbol{x}_i)],\tag{A.12}$$

where $\mathcal{C}_1\in[0,1]$.

For the lower bound, when $f(\boldsymbol{x}_i+\Delta\boldsymbol{x}_i|T)\geq f(\boldsymbol{x}_i|T)$,

$$\exp(-f(\boldsymbol{x}_i|T))-\exp(-f(\boldsymbol{x}_i+\Delta\boldsymbol{x}_i|T))\geq 0+o(\Delta\boldsymbol{x}_i);\tag{A.13}$$

when $f(\boldsymbol{x}_i+\Delta\boldsymbol{x}_i|T)<f(\boldsymbol{x}_i|T)$,

$$\exp(-f(\boldsymbol{x}_i|T))-\exp(-f(\boldsymbol{x}_i+\Delta\boldsymbol{x}_i|T))>f'\left(\boldsymbol{x}_i|T\right)\cdot\Delta\boldsymbol{x}_i+o(\Delta\boldsymbol{x}_i).\tag{A.14}$$

Thus, we yield

$$e\mathrm{rr}_i-e\mathrm{rr}_{i'}\geq\mathcal{C}_2\mathbb{E}_T[f'(\boldsymbol{x}_i|T)\cdot\Delta\boldsymbol{x}_i+o(\Delta\boldsymbol{x}_i)],\tag{A.15}$$

where $\mathcal{C}_2\in[0,1]$.

Obviously, $\mathcal{C}_2\leq\mathcal{C}_1$. We consider the most cases, where $\mathcal{C}_1\neq 0$ and $\mathcal{C}_2\neq 0$. The difference between the two errors satisfies the following formula:

$$\begin{aligned}\mathcal{C}_2\mathbb{E}_T[f'(\boldsymbol{x}_i|T)\cdot\Delta\boldsymbol{x}_i+o(\Delta\boldsymbol{x}_i)]&\leq e\mathrm{rr}_i-e\mathrm{rr}_{i'}\\&\leq\mathcal{C}_1\mathbb{E}_T[f'(\boldsymbol{x}_i|T)\cdot\Delta\boldsymbol{x}_i+o(\Delta\boldsymbol{x}_i)].\end{aligned}\tag{A.16}$$

Ignore the higher-order term $o(\Delta \boldsymbol{x}_i)$, we have

$$\mathcal{C}_2 \mathbb{E}_T[f'(\boldsymbol{x}_i|T) \cdot \Delta \boldsymbol{x}_i] \leq \mathrm{err}_i - \mathrm{err}_{i'} \leq \mathcal{C}_1 \mathbb{E}_T[f'(\boldsymbol{x}_i|T) \cdot \Delta \boldsymbol{x}_i]. \tag{A.17}$$

From the formula above, there are three cases. Let $\angle\varphi$ be the angle between the direction of $\Delta \boldsymbol{x}_i$ and the direction of $\mathbb{E}_T[f'(\boldsymbol{x}_i|T)]$. The cases are summarized as below.

1. If $\mathbb{E}_T[f'(\boldsymbol{x}_i|T)\, \Delta \boldsymbol{x}_i] > 0$, then $\angle\varphi < 90°$. In this case, the direction of the perturbation $\Delta \boldsymbol{x}_i$ and the direction of $\mathbb{E}_T[f'(\boldsymbol{x}_i|T)$ are consistent. Thus, the generalization error of the noisy sample is smaller than that of the clean one.
2. If $\mathbb{E}_T[f'(\boldsymbol{x}_i|T)\, \Delta \boldsymbol{x}_i] < 0$, then $\angle\varphi > 90°$. In this case, the direction of the perturbation $\Delta \boldsymbol{x}_i$ and the direction of $\mathbb{E}_T[f'(\boldsymbol{x}_i|T)$ are contradictory. Thus, the generalization error of the noisy sample is larger than that of the clean one.
3. If $\mathbb{E}_T[f'(\boldsymbol{x}_i|T)\, \Delta \boldsymbol{x}_i] = 0$, then $\angle\varphi = 90°$ or $\Delta \boldsymbol{x}_i = \mathbf{0}$. The generalization error does not change in this case.

Therefore, the change of the generalization error with feature noise is dependent on the angle between the direction of the perturbation $\Delta \boldsymbol{x}_i$ and the direction of $\mathbb{E}_T[f'(\boldsymbol{x}_i|T)]$.

B.2 Proof of Proposition 2

In this section, we offer the proof of Proposition 2.

Proof. Let π be the label corruption rate, that is, the probability of each label flipping to another one. When label noise is added, the generalization error of the sample $(\boldsymbol{x}_i, y_i = 1)$ after adding label noise (i.e., $(\boldsymbol{x}_i, y_i' = -1)$) becomes

$$\begin{aligned} \mathrm{err}_{i''} &= \mathbb{E}_T\left[\ell\left(f\left(\boldsymbol{x}_i|T\right), y_i'\right)\right] \\ &= \int_{T \in \mathcal{X} \times \mathcal{Y}} (1-\pi)\, e^{-f(\boldsymbol{x}_i|T)} + \pi e^{f(\boldsymbol{x}_i|T)} dP\left(T\right). \end{aligned} \tag{A.18}$$

The sign of the output $f(\boldsymbol{x}_i)$ indicates the predicted label. For samples which are classified correctly, $y_i f(\boldsymbol{x}_i) > 0$, otherwise, $y_i f(\boldsymbol{x}_i) < 0$. To clearly distinguish between correctly predicted and wrongly predicted samples, the absolute value of $f(\boldsymbol{x}_i)$ is utilized. Therefore, if a sample is correctly predicted, if loss is $e^{-|f(\boldsymbol{x}_i)|}$; otherwise, the loss is $e^{|f(\boldsymbol{x}_i)|}$. As the probability of a sample being correctly classified is p and $p > 0.5$, the generalization error of the original sample $(\boldsymbol{x}_i, y_i = 1)$ is

$$\begin{aligned} \mathrm{err}_i &= \mathbb{E}_T\left[\ell\left(f\left(\boldsymbol{x}_i|T\right), y_i\right)\right] \\ &= \int_{T \in \mathcal{X} \times \mathcal{Y}} p e^{-|f(\boldsymbol{x}_i|T)|} + (1-p) e^{|f(\boldsymbol{x}_i|T)|} dP\left(T\right). \end{aligned} \tag{A.19}$$

After flipping the label of the sample, its generalization error becomes

$$\begin{aligned} \mathrm{err}_{i''} &= \mathbb{E}_T\left[\ell\left(f\left(\boldsymbol{x}_i|T\right), y_i'\right)\right] \\ &= \int_{T \in \mathcal{X} \times \mathcal{Y}} (1-\pi)\, p e^{-|f(\boldsymbol{x}_i|T)|} + (1-\pi)\,(1-p) e^{|f(\boldsymbol{x}_i|T)|} dP\left(T\right) \\ &+ \int_{T \in \mathcal{X} \times \mathcal{Y}} \pi(1-p) e^{-|f(\boldsymbol{x}_i|T)|} + \pi p e^{|f(\boldsymbol{x}_i|T)|} dP\left(T\right). \end{aligned} \tag{A.20}$$

Comparing the two generalization errors above, we yield

$$\mathrm{err}_{i''} - \mathrm{err}_i = \pi(2p-1)\int_{T\in\mathcal{X}\times\mathcal{Y}} e^{|f(\boldsymbol{x}_i|T)|} - e^{-|f(\boldsymbol{x}_i|T)|} dP\left(T\right) > 0. \tag{A.21}$$

Therefore, for a sample that is more likely to be predicted correctly (i.e., the probability of being correctly classified p is greater than 0.5), its generalization error after adding label noise is larger than that of the original one. Therefore, the generalization errors of the samples with label noises are larger than those of the clean ones on the average.

B.3 Proof of Proposition 3

In this section, we offer the proof of Proposition 3.

Proof. Take the output of the sigmoid layer as the model's output $f(\boldsymbol{x})$ and $f(\boldsymbol{x}) \in (0,1)$. The disproved method is adopted to prove this proposition. We prove that if the average probability of ground truth of the large category, which contains the majority of samples, is smaller than that of the small category, then the classifier must not be the approximate Bayesian optimal classifier. There are two categories which are $y=1$ and $y=-1$. The numbers of samples in the two categories are $\mathbb{c}_1$ and $\mathbb{c}_2$. The condition is that $\mathbb{c}_1 > e\mathbb{c}_2$ ($c_\tau > e:1$), which means that the number of samples in the large category is e times the number of samples in the small category. Denote the average probabilities of ground truth for the large and small categories as f_1 and f_2, and $f_1 < f_2$. The total error is

$$\begin{aligned}\mathcal{L} &= \frac{1}{n}\left(\sum_{i=1}^{\mathbb{c}_1}\ell\left(f\left(\boldsymbol{x}_i\right),y_i\right) + \sum_{j=1}^{\mathbb{c}_2}\ell\left(f\left(\boldsymbol{x}_j\right),y_j\right)\right) \\ &= \frac{1}{n}\left[\mathbb{c}_1 e^{-f_1} + \mathbb{c}_2 e^{1-f_2}\right].\ (i)\end{aligned} \tag{A.22}$$

Denote $\frac{1}{n}\left[\mathbb{c}_1 e^{-f_2} + \mathbb{c}_2 e^{1-f_1}\right]$ as (ii). $(i)-(ii)$ gives

$$\begin{aligned}&\frac{1}{n}\left[\mathbb{c}_1 e^{-f_1} + \mathbb{c}_2 e^{1-f_2} - \mathbb{c}_1 e^{-f_2} - \mathbb{c}_2 e^{1-f_1}\right] \\ &= \frac{1}{n}\left[\mathbb{c}_1\left(e^{-f_1} - e^{-f_2}\right) + \mathbb{c}_2\left(e^{1-f_2} - e^{1-f_1}\right)\right] \\ &= \frac{1}{n}\left[\mathbb{c}_1\frac{e^{f_2}-e^{f_1}}{e^{f_1}e^{f_2}} + \mathbb{c}_2\frac{e\left(e^{f_1}-e^{f_2}\right)}{e^{f_1}e^{f_2}}\right] \\ &= \frac{1}{n}\frac{e^{f_2}-e^{f_1}}{e^{f_1}e^{f_2}}\left[\mathbb{c}_1 - \mathbb{c}_2 e\right] > 0\end{aligned} \tag{A.23}$$

Obviously, if the average probability of the ground truth for the large category is smaller than that of the small category, then the predictor is not an approximate Bayesian optimal classifier. Thus, if a predictor is an approximate Bayesian optimal classifier, the average probability of the ground truth for the large category is greater than that of the small category, that is $f_1 > f_2$. Proposition 3 holds.

Proposition 3 indicates that the average probability of the ground truth of samples in the large category f_1 is greater than that of samples in the small category f_2 for an approximate Bayesian optimal classifier. According to Proposition 3, it is natural to get the following proposition.

Proposition A.1. *The average generalization error* $\overline{\mathrm{err}}$ *of samples in the large category* $\overline{\mathrm{err}}_1$ *is larger than that of samples in the small category* $\overline{\mathrm{err}}_2$ *.*

The proof is shown below.

Proof. If the positive category is the large category, the average generalization error of samples in this category is

$$\overline{\mathrm{err}}_1 = \int_{T \in \mathcal{X} \times \mathcal{Y}} e^{-f_1} dP(T), \tag{A.24}$$

If the positive category is the small category, the average generalization error of samples in this category is

$$\overline{\mathrm{err}}_2 = \int_{T \in \mathcal{X} \times \mathcal{Y}} e^{-f_2} dP(T), \tag{A.25}$$

where $\overline{\mathrm{err}}_1$ denotes the average generalization error of samples in the large category, and $\overline{\mathrm{err}}_2$ denotes the average generalization error of samples in the small category. All sampled datasets are imbalanced, in which $c_r > e : 1$. The difference between the two average generalization errors equals to

$$\overline{\mathrm{err}}_1 - \overline{\mathrm{err}}_2 = \int_{T \in \mathcal{X} \times \mathcal{Y}} e^{-f_1} - e^{-f_2} dP(T). \tag{A.26}$$

From Proposition 3, we know that $f_1 > f_2$. Thus, $\overline{\mathrm{err}}_1 < \overline{\mathrm{err}}_2$, which means that the average generalization error of samples in the large category is smaller than that of samples in the small category.

B.4 Proof of Proposition 4

In this section, we offer the proof of Proposition 4.

Proof. According to the moment-generating function, there is

$$E\left[e^{tx}\right] = e^{t\mu + \frac{1}{2}\sigma^2 t^2}, \quad x \sim \mathcal{N}\left(\mu, \sigma^2\right). \tag{A.27}$$

When $t = -1$, it can be drawn that

$$E\left[e^{-x}\right] = e^{-\mu + \frac{1}{2}\sigma^2}, \quad x \sim \mathcal{N}\left(\mu, \sigma^2\right). \tag{A.28}$$

For sample $\boldsymbol{x}_i$, its generalization error and margin are denoted as err_i and γ_i, respectively. As the condition that the functional margins γ_i of sample $\boldsymbol{x}_i$ trained on random datasets obey the Gaussian distribution $\mathcal{N}$, and the mean of the distribution μ_i is the true margin corresponding to the oracle decision boundary. Although this condition is

intuitively, we verify it by a lot of experiments, in which the results are shown in Section E.1.

Thus, for samples $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$, there are $\gamma_1 \sim \mathcal{N}(\mu_1, \sigma_1^2)$ and $\gamma_2 \sim \mathcal{N}(\mu_2, \sigma_2^2)$. Based on the moment-generating function, we yield

$$\mathbb{E}_T\left[e^{-\gamma_1(T)}\right] = e^{-\mu_1+\frac{1}{2}\sigma_1^2}, \tag{A.29}$$

and

$$\mathbb{E}_T\left[e^{-\gamma_2(T)}\right] = e^{-\mu_2+\frac{1}{2}\sigma_2^2}. \tag{A.30}$$

Therefore, when $\sigma_1 = \sigma_2$, if $\mu_1 \leq \mu_2$, then we have

$$\mathbb{E}_T[e^{-\gamma_1(T)}] \geq \mathbb{E}_T[e^{-\gamma_2(T)}], \tag{A.31}$$

which indicates that $e\mathrm{rr}_1 > e\mathrm{rr}_2$.

For the second case, when $\mu_1 = \mu_2$, if $\sigma_1^2 \geq \sigma_2^2$, we obtain

$$\mathbb{E}_T[e^{-\gamma_1(T)}] \geq \mathbb{E}_T[e^{-\gamma_2(T)}], \tag{A.32}$$

which also indicates $e\mathrm{rr}_1 > e\mathrm{rr}_2$.

Thus, the true margin (the mean of the functional margin distribution) of a sample and its generalization error are negatively correlated when the margin variances of samples are equal, while the margin variance and the generalization error are positively correlated when the true margins are equal.

This proposition indicates that the conclusion of [4] that samples close to the oracle decision boundary are hard ones does not always hold. Even samples that have large true margins may have large errors.

B.5 Proof of Section 3.4

In this section, we analyze the relationship between the epistemic uncertainty and the generalization error. The epistemic uncertainties can be formulated as a probability distribution over model parameters. The aim is to optimize the parameters, i.e., $\boldsymbol{\theta}$ of a function $y = f_{\boldsymbol{\theta}}(x)$ that can produce the desired output. Then, we prove that the epidemic uncertainty is exactly the variance term in the generalization error following some reasonable conditions.

For a given dataset T over $\boldsymbol{\theta}$, the posterior distribution is $p(\boldsymbol{\theta}|T)$. A class label with regard to the $p(\boldsymbol{\theta}|T)$ for a given test sample $\boldsymbol{x}^*$ can be predicted:

$$p\left(y^* \mid \boldsymbol{x}^*, T\right) = \int p\left(y^* \mid \boldsymbol{x}^*, \boldsymbol{\theta}\right) p(\boldsymbol{\theta} \mid T) d\boldsymbol{\theta} \tag{A.33}$$

This process is called inference or marginalization. However, $p(\boldsymbol{\theta}|T)$ cannot be computed analytically, but it can be approximated by various methods.

The predictive variance $Var(f(\boldsymbol{x}_i|\boldsymbol{\theta}_1), \cdots, f(\boldsymbol{x}_i|\boldsymbol{\theta}_K))$ plus a precision constant is a typical manner to approximate the epistemic uncertainty [5,6]. Take the mean square loss as an example[1], the epistemic uncertainty is

$$\begin{aligned}\widehat{\mathrm{Var}}\left[\boldsymbol{x}_i\right] :=& \tau^{-1} + \frac{1}{|K|}\sum\nolimits_k f(\boldsymbol{x}_i|\boldsymbol{\theta}_k)^\intercal f(\boldsymbol{x}_i|\boldsymbol{\theta}_k) \\ &- E[f(\boldsymbol{x}_i|\boldsymbol{\theta}_k)]^\intercal E[f(\boldsymbol{x}_i|\boldsymbol{\theta}_k)],\end{aligned} \tag{A.34}$$

where τ is a precision constant. The second term in Eq. (A.34) is the second raw moment of the predictive distribution and the third term is the square of the first moment. When $K \to \infty$ and the constant term is ignored, Eq. (A.34) becomes

$$\widehat{\mathrm{Var}}\left[\boldsymbol{x}_i\right] := \int_{\boldsymbol{\theta}} ||f(\boldsymbol{x}_i|\boldsymbol{\theta}) - \overline{f}_{\boldsymbol{\theta}}(\boldsymbol{x}_i)||_2^2 dP(\boldsymbol{\theta}|T), \tag{A.35}$$

where $\overline{f}_{\boldsymbol{\theta}}(\boldsymbol{x}_i) = \mathbb{E}_{\boldsymbol{\theta}}[f(\boldsymbol{x}_i|\boldsymbol{\theta})]$. If $P(\boldsymbol{\theta}|T)$ is approximated by the distribution of the learned models on random training sets which conform to the Gaussian distribution $\mathcal{N}(T, \delta I)$, Eq. (A.35) becomes

$$\widehat{\mathrm{Var}}\left[\boldsymbol{x}_i\right] = \int_{T \in \mathcal{X}\times\mathcal{Y}} ||f(\boldsymbol{x}_i|T) - \overline{f}(\boldsymbol{x}_i)||_2^2 dP(T), \tag{A.36}$$

where $\overline{f}(\boldsymbol{x}_i) = \mathbb{E}_T[f(\boldsymbol{x}_i|T)]$. Formula (A.36) is exactly the variance term of the generalization error when the mean square loss is adopted.

C Supplementary Material for Section 4.1

In this section, we offer the omitted discussions and proofs of Section 4.1.

C.1 Linear Predictor

Proof of Proposition 5 In this section, we offer proof of Proposition 5.

Proof. Following [8], the decision boundaries of the linear predictors share certain characteristics with the support vector machine (SVM) since they rely on the same support vectors. As a matter of fact, the current understandings of the implicit bias of gradient descent are mostly established on the connection with the hard-margin SVM [9]. Denote $\boldsymbol{\theta}^*$ as the optimal solution to the hard-margin SVM:

$$\min_{\boldsymbol{\theta}\in\mathbb{R}^{\mathrm{d}}} \|\boldsymbol{\theta}\|_2 \quad \text{s.t.} \quad y_i f\left(\boldsymbol{\theta}, \mathbf{x}_i\right) \geq 1 \quad \forall i = 1, 2, \ldots, n. \tag{A.37}$$

Define the corresponding margin $\gamma^* = \gamma\left(\boldsymbol{\theta}^*\right) := \min_i y_i f\left(\boldsymbol{\theta}^*, \mathbf{x}_i\right)$.

The convergence speed of the linear predictors satisfies the following proposition:

[1] For other losses, there are other methods to calculate the predictive variance [7].

Proposition A.2. *(Proposition 1 of Xu et al. [4]) With a constant learning rate $\eta_t \leq \beta^{-1}$ and the normalized weights $\boldsymbol{w} \in [1/M, M]^n$, such that $\sum_{i=1}^n w_i = 1$, without loss of generality, it holds that:*

$$\left| \frac{\boldsymbol{\theta}_t(\mathbf{w})}{\|\boldsymbol{\theta}_t(\mathbf{w})\|_2} - \boldsymbol{\theta}^* \right| \lesssim \frac{\log n + D_{KL}\left(\boldsymbol{p} \| \mathbf{w}\right) + M}{\log t \cdot \gamma^*} \tag{A.38}$$

where $\boldsymbol{p} = [p_1, \ldots, p_n]$ characterizes the dual optimal for the hard-margin SVM such that $\boldsymbol{\theta}^* = \sum_{i=1}^n y_i \boldsymbol{x}_i \cdot p_i$ and satisfies: $p_i \geq 0$ and $\sum_{i=1}^n p_i = 1$. Here, D_{KL} is the Kullback-Leibler divergence.

The decrease of the upper bound in the right side of Formula (A.38) may accelerate the convergence speed. To reduce $D_{KL}\left(\boldsymbol{p} \| \boldsymbol{w}\right)$, samples with high dual coefficients (samples with small functional margins) should be assigned with large weights. The optimal dual coefficient vector is a monotonically decreasing function of the functional margin which is denoted as g. Thus, $p_i = g(\gamma_i)$. The concrete form of g is unknown. Here, we consider two typical types, including the linear and the exponential forms. The condition of Proposition 5 is that $\mathrm{err}_i \geq \mathrm{err}_j$.

First, we prove (1) in Proposition 5 that is when g is a linear function, if $\mu_i < \mu_j$, then $\mathbb{E}_T[p_i] > \mathbb{E}_T[p_j]$. It means that samples with small true margins should be assigned large weights on the average.

As before, we assume that the functional margins of random datasets obey a Gaussian distribution, thus, we have

$$\gamma_i \sim \mathcal{N}(\mu_i, \sigma_i^2), \tag{A.39}$$

and

$$\gamma_j \sim \mathcal{N}(\mu_j, \sigma_j^2). \tag{A.40}$$

As g is a linear decreasing function of the functional margin, we have $p_i = g(\gamma_i) = a\gamma_i + b$ ($a < 0$). Thus, the optimal dual coefficients of random datasets also obey a Gaussian distribution. There are

$$p_i \sim \mathcal{N}(a\mu_i + b, a^2\sigma_i^2), \tag{A.41}$$

and

$$p_j \sim \mathcal{N}(a\mu_j + b, a^2\sigma_j^2). \tag{A.42}$$

Then, if $\mu_i < \mu_j$, we obtain that $a\mu_i + b > a\mu_j + b$, i.e., $E[p_i] > E[p_j]$. Thus, (1) holds, that is, if p_i is a linear function of γ_i, samples close to the oracle decision boundary should be assigned with large weights on the average.

Next, we prove (2) in Proposition 5, in which the dual optimal p_i of sample $\boldsymbol{x}_i$ is a negative exponential function of the functional margin γ_i, i.e.,

$$p_i = ae^{-\gamma_i}, \tag{A.43}$$

where $a > 0$. According to the moment-generating function, we have

$$E[p_i] = aE[e^{-\gamma_i}] = ae^{-\mu_i + \frac{1}{2}\sigma_i^2} = a \cdot \mathrm{err}_i, \tag{A.44}$$

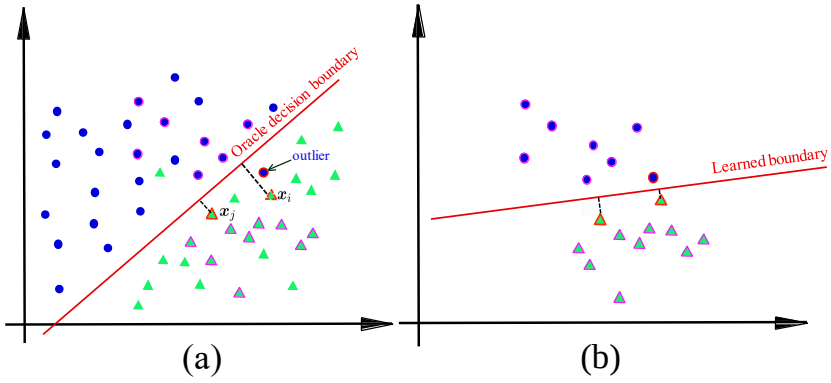

Fig. A-1: The situation where samples with small true margins have small weights on a specific training set. (a) shows the oracle decision boundary of the two categories. (b) shows the learned boundary on a sampled training set. Although μ_i is larger than μ_j shown in (a), the margin variance of sample $\boldsymbol{x}_i$ is large as there are heterogeneous nodes around it. Therefore, $\mathrm{err}_i > \mathrm{err}_j$ and $E[p_i] > E[p_j]$.

and

$$E[p_j] = aE[e^{-\gamma_j}] = ae^{-\mu_j + \frac{1}{2}\sigma_j^2} = a \cdot \mathrm{err}_j, \tag{A.45}$$

Thus, as $\mathrm{err}_i \geq \mathrm{err}_j$, $E[p_i] \geq E[p_j]$.

For the two samples described in Proposition 5, there are three typical situations.

- If $\sigma_i^2 = \sigma_j^2$, then $\mu_i \leq \mu_j$ as $\mathrm{err}_i \geq \mathrm{err}_j$.
- If $\mu_i = \mu_j$, then $\sigma_i^2 \geq \sigma_j^2$ as $\mathrm{err}_i \geq \mathrm{err}_j$.
- If $\mu_i > \mu_j$, then $\sigma_i^2 > \sigma_j^2$ as $\mathrm{err}_i > \mathrm{err}_j$.

In the third situation, although $\boldsymbol{x}_j$ has a smaller true margin μ_j, its generalization error is still smaller than that of $\boldsymbol{x}_i$ as $\boldsymbol{x}_j$ has a small margin variance. Fig. A-1 shows an illustrative example for the third case.

Convergence Direction of Linear Predictor For the convergence direction of the linear predictor with difficulty-based weights, we offer the following proposition:

Theorem A.1. *For the linear predictor with difficulty-based weights on separable data, if $\boldsymbol{\theta}^*$ is the L_2 max-margin vector (the solution to the hard margin SVM), we have* $\lim_{t\to\infty} \boldsymbol{\theta}_t \left[\boldsymbol{w}(\boldsymbol{d})\right] = \boldsymbol{\theta}^*$.

Theorem 1 indicates that the difficulty-based weighting scheme does not change the parameters' convergence direction to the max-margin solution. We give the proof sketch of it.

Proof. To simplify the notation, in this proof we assume that all labels are positive, this is true without loss of generality. Following our assumption shown in the body, the loss function $\mathcal{L}(\cdot)$ is a $\beta\sigma_{\max}^2$-smooth function, where $\sigma_{\max}(\mathbf{X})$ is the maximal singular

value of the data matrix $\mathbf{X} \in \mathbb{R}^{\mathbb{d} \times n}$ with $\mathbb{d}$ is the feature dimension. The Gradient Descent with a fixed learning rate η is used to achieve the minimum of the loss by the following schema

$$\boldsymbol{\theta}_{t+1}(\boldsymbol{w}) = \boldsymbol{\theta}_t(\boldsymbol{w}) - \eta \nabla \mathcal{L}(\boldsymbol{\theta}_t(\boldsymbol{w})). \tag{A.46}$$

And the gradient descent satisfies the following lemma.

Lemma A.1. *(Lemma 1 of Soudry et al. [8]) Let $\boldsymbol{\theta}_t$ be the iterates of gradient descent with $\eta < 2\beta^{-1}\sigma_{max}^{-2}(\boldsymbol{X})$ and any starting point $\boldsymbol{\theta}_0$. Under assumptions shown in the body and above, we have:*

1. $\lim_{t\to\infty} \mathcal{L}(\boldsymbol{\theta}_t(\boldsymbol{w})) = 0$,
2. $\lim_{t\to\infty} \|\boldsymbol{\theta}_t(\boldsymbol{w})\| = \infty$,
3. $\forall i : \lim_{t\to\infty} \boldsymbol{\theta}_t(\boldsymbol{w})^\intercal \boldsymbol{x}_i = \infty$.

According to Lemma A.1, $\forall i: \quad \boldsymbol{\theta}_t(\boldsymbol{w})^\intercal \boldsymbol{x}_i \to \infty$, if $\boldsymbol{\theta}_t(\boldsymbol{w}) / \|\boldsymbol{\theta}_t(\boldsymbol{w})\|$ converges to $\boldsymbol{\theta}_\infty$. We then have $\boldsymbol{\theta}_t(\boldsymbol{w}) = g_t \boldsymbol{\theta}_\infty + \boldsymbol{\rho}_t$, such that with $g_t \to \infty, \forall i, \boldsymbol{x}_i^\intercal \boldsymbol{\theta}_\infty > 0$. If the difficulty measure $\boldsymbol{d}$ is not a function of $\boldsymbol{\theta}_t$, the gradient can be written as

$$\begin{aligned} -\nabla \mathcal{L}(\boldsymbol{\theta_t}(\boldsymbol{w})) &= \sum\nolimits_{i=1}^{n} w(d_i) \exp(-\boldsymbol{\theta}_t^\intercal \boldsymbol{x}_i) \boldsymbol{x}_i \\ &= \sum\nolimits_{i=1}^{n} w(d_i) \exp(-g_t {\boldsymbol{\theta}_\infty}^\intercal \boldsymbol{x}_i) \exp(-{\boldsymbol{\rho}_t}^\intercal \boldsymbol{x}_i) \boldsymbol{x}_i, \end{aligned} \tag{A.47}$$

From the above formula, only samples with the least negative exponents which are called support vectors, and non-zero difficulty-based weights will contribute to the gradient. Thus, the difficulty-based weights influence the convergence value of the parameter. $\boldsymbol{\theta}_\infty$ will then be dominated by gradients of the samples with the least negative exponents and it will be the linear combination of support vectors with non-zero weights. So will its scaling $\boldsymbol{\theta}^* = \boldsymbol{\theta}_\infty / \min_n(\boldsymbol{\theta}_\infty^\intercal \boldsymbol{x}_n)$. Thus, we have

$$\begin{aligned} \boldsymbol{\theta}^* &= \sum\nolimits_{i=1}^{n} \alpha_i w(d_i) \boldsymbol{x}_i \\ \forall i\ (\alpha_i w(d_i) &\geq 0\ and\ \boldsymbol{\theta}^{*\intercal} \boldsymbol{x}_i = 1), \\ \text{or } (\alpha_i w(d_i) &= 0\ and\ \boldsymbol{\theta}^{*\intercal} \boldsymbol{x}_i > 1), \end{aligned} \tag{A.48}$$

which is exactly the KKT conditions for SVM. Thus, we know that the parameters converge to the max-margin solution. When the difficulty measure $\boldsymbol{d}$ is a function of $\boldsymbol{\theta}_t$, the loss gradient is

$$\begin{aligned} -\nabla \mathcal{L}(\boldsymbol{\theta}) &= -\sum_{i=1}^{n} w'(d_i) \exp(-\boldsymbol{\theta}_t^\intercal \boldsymbol{x}_i) - w(d_i) \exp(-\boldsymbol{\theta}_t^\intercal \boldsymbol{x}_i) \boldsymbol{x}_i \\ &= \sum\nolimits_{i=1}^{n} -w'(d_i) \exp(-g_t \boldsymbol{\theta}_\infty^\intercal \boldsymbol{x}_i) \exp(-\boldsymbol{\rho}_t^\intercal \boldsymbol{x}_i) \\ &\quad + w(d_i) \exp(-g_t \boldsymbol{\theta}_\infty^\intercal \boldsymbol{x}_i) \exp(-\boldsymbol{\rho}_t^\intercal \boldsymbol{x}_i) \boldsymbol{x}_i \end{aligned} \tag{A.49}$$

As $g_t \to \infty$ and the exponents become more negative, only those with the largest exponents will contribute to the gradient which is the same as the occasion where the weighting function does not require derivation. The difference is that even a sample's weight is zero, as long as the weighting function's derivative is non-zero, this sample still has an influence on the gradient when the weighting function requires derivation. Therefore, $\boldsymbol{\theta}_\infty$ will also be dominated by these gradients and will be the linear combination of support vectors. Therefore, the parameters still converge to the max-margin solution. In summary, Theorem A.1 is proved.

C.2 Nonlinear Predictor

This section provides proofs of Theorem 1 under the multi-class setting where the cross-entropy loss is used.

The only assumption we need to make is that the training data can be separated by f at some point during gradient descent. As stated in Section 2.1, under multi-class setting, $\mathcal{F} = \{f(\boldsymbol{\theta}, \cdot) | \boldsymbol{\theta} \in \boldsymbol{\Theta} \subset \mathbb{R}^C\}$, where C is the number of categories. Now, we focus on the λ-regularized cross-entropy loss, defined as

$$\mathcal{L}_\lambda(\boldsymbol{\theta}) \triangleq -\frac{1}{n}\sum_{i=1}^{n} w(d_i) \log\left(\frac{\exp\left(f_{y_i}(\boldsymbol{\theta}, \boldsymbol{x}_i)\right)}{\sum_{j=1}^{C}\exp\left(f_{y_j}(\boldsymbol{\theta}, \boldsymbol{x}_j)\right)}\right) + \lambda||\boldsymbol{\theta}||^r, \tag{A.50}$$

for fixed $r > 0$. First, we state two technical lemmas that characterize the loss function. From Xu et al. [4], we know the following two lemmas hold.

Lemma A.2. $\arg\min_{\boldsymbol{\theta}} \mathcal{L}_\lambda(\boldsymbol{\theta})$ *exists.*

Lemma A.3. *There exists a critical (stationary) point such that* $\lim_{\lambda\to 0} L_\lambda(\boldsymbol{\theta}^*; \boldsymbol{w}) = 0$*, and* $||\boldsymbol{\theta}_\lambda(\boldsymbol{w})|| \to \infty$ *is the critical point.*

Lemma A.2 indicates that $\mathcal{L}_\lambda$ indeed has a global minimizer and Lemma A.3 shows that as λ decrease, the norm of the solution $||\boldsymbol{\theta}_\lambda(\boldsymbol{w})||$ grows. Let $\boldsymbol{\theta}_\lambda(\boldsymbol{w}) \in \operatorname{argmin}\mathcal{L}_\lambda(\boldsymbol{\theta})$, the normalized margin is defined as:

$$\gamma_\lambda(\boldsymbol{w}) \triangleq \min_i \left(f_{y_i}\left(\overline{\boldsymbol{\theta}}_\lambda(\boldsymbol{w}), \boldsymbol{x}_i\right) - \max_{j\neq i}\left(f_{y_j}\left(\overline{\boldsymbol{\theta}}_\lambda(\boldsymbol{w}), \boldsymbol{x}_j\right)\right)\right), \tag{A.51}$$

where $\overline{\boldsymbol{\theta}}_\lambda(\boldsymbol{w}) = \boldsymbol{\theta}_\lambda(\boldsymbol{w})/\|\boldsymbol{\theta}_\lambda(\boldsymbol{w})\|$. The $||\cdot||$-max normalized margin is defined as:

$$\gamma^* = \max_{||\boldsymbol{\theta}||\leq 1} \min_i \left(f_{y_i}(\boldsymbol{\theta}, \boldsymbol{x}_i) - \max_{j\neq i}\left(f_{y_j}(\boldsymbol{\theta}, \boldsymbol{x}_j)\right)\right). \tag{A.52}$$

Proof of Theorem 1

Proof. To prove (1) in Theorem 1, the exponential scaling of the cross entropy is adopted. $\mathcal{L}_\lambda$ can be lower bounded that scales with $\exp(-||\boldsymbol{\theta}_\lambda(\boldsymbol{w})||\gamma_\lambda(\boldsymbol{w}))$ and it can also be upper bounded that scales with $\exp(-||\boldsymbol{\theta}_\lambda(\boldsymbol{w})||\gamma^*)$. By Lemma A.3, a large enough $||\boldsymbol{\theta}_\lambda(\boldsymbol{w})||$ can be taken, so the gap $\gamma^* - \gamma_\lambda(\boldsymbol{w})$ vanishes. The weight $\boldsymbol{w} \in [b, B]^n$ and $b > 0$. Then, for any $M > 0$,

$$\begin{aligned}
\mathcal{L}_\lambda(M\boldsymbol{\theta}) &\triangleq \frac{1}{n}\sum_{i=1}^{n} -w(d_i)\log\frac{\exp\left(M^a f_{y_i}(\boldsymbol{\theta}, \boldsymbol{x}_i)\right)}{\sum_{j=1}^{C}\exp\left(M^a f_{y_j}(\boldsymbol{\theta}, \boldsymbol{x}_i)\right)} + \lambda M^r||\boldsymbol{\theta}||^r \\
&= \frac{1}{n}\sum_{i=1}^{n} -w(d_i)\log\frac{1}{1+\sum_{y_j\neq y_i}\exp\left(M^a\left(f_{y_j}(\boldsymbol{\theta}, \boldsymbol{x}_i) - f_{y_i}(\boldsymbol{\theta}, \boldsymbol{x}_i)\right)\right)} + \lambda M^r||\boldsymbol{\theta}||^r \\
&\leq B\log\left(1 + (C-1)\exp\left(-M^a\gamma_{\boldsymbol{\theta}}(\boldsymbol{w})\right)\right) + \lambda M^r||\boldsymbol{\theta}||^r,
\end{aligned} \tag{A.53}$$

where $\gamma_{\boldsymbol{\theta}}(\boldsymbol{w}) \triangleq \min_i \left(f_{y_i}\left(\overline{\boldsymbol{\theta}}(\boldsymbol{w}), \boldsymbol{x}_i\right) - \max_{y_j \neq y_i} f_{y_j}\left(\overline{\boldsymbol{\theta}}(\boldsymbol{w}), \boldsymbol{x}_i\right)\right)$ and $\overline{\boldsymbol{\theta}}(\boldsymbol{w}) = \boldsymbol{\theta}(\boldsymbol{w})/||\boldsymbol{\theta}(\boldsymbol{w})||$. Below we calculate the lower bound of the loss function. According to

$$\begin{aligned} &\sum_{y_j \neq y_i} \exp\left(M^a\left(f_{y_j}\left(\boldsymbol{\theta}(\boldsymbol{w}), \boldsymbol{x}_i\right) - f_{y_i}\left(\boldsymbol{\theta}(\boldsymbol{w}), \boldsymbol{x}_i\right)\right)\right) \\ \geq &\max \exp\left(M^a\left(f_{y_j}\left(\boldsymbol{\theta}(\boldsymbol{w}), \boldsymbol{x}_i\right) - f_{y_i}\left(\boldsymbol{\theta}(\boldsymbol{w}), \boldsymbol{x}_i\right)\right)\right) \\ = &\exp\left(-M^a \gamma_\theta(\boldsymbol{w})\right), \end{aligned} \tag{A.54}$$

a lower bound can be obtained:

$$\mathcal{L}_\lambda\left(M\boldsymbol{\theta}(\boldsymbol{w})\right) \geq \frac{b}{n}\log\left(1 + \exp\left(-M^a \gamma_\theta(\boldsymbol{w})\right)\right) + \lambda M^r ||\boldsymbol{\theta}(\boldsymbol{w})||^r. \tag{A.55}$$

Given $M = ||\boldsymbol{\theta}_\lambda(\boldsymbol{w})||$ and $\boldsymbol{\theta} = \boldsymbol{\theta}^*$, noting that $||\boldsymbol{\theta}^*|| \leq 1$, the upper bound of the loss function is

$$\mathcal{L}_\lambda\left(\boldsymbol{\theta}^* || \boldsymbol{\theta}_\lambda(\boldsymbol{w})||\right) \leq B\log\left(1 + (C-1)\exp\left(-||\boldsymbol{\theta}_\lambda(\boldsymbol{w})||^a \gamma^*\right)\right) + \lambda ||\boldsymbol{\theta}_\lambda(\boldsymbol{w})||^r. \tag{A.56}$$

Next, we lower bound $\mathcal{L}_\lambda\left(\boldsymbol{\theta}_\lambda(\boldsymbol{w})\right)$ by applying Formula (A.55).

$$\mathcal{L}_\lambda\left(\boldsymbol{\theta}_\lambda(\boldsymbol{w})\right) \geq \frac{b}{n}\log\left(1 + \exp\left(-||\boldsymbol{\theta}_\lambda(\boldsymbol{w})||^a \gamma_\lambda(\boldsymbol{w})\right)\right) + \lambda ||\boldsymbol{\theta}_\lambda(\boldsymbol{w})||^r. \tag{A.57}$$

With $\mathcal{L}_\lambda(\boldsymbol{\theta}) \leq \mathcal{L}_\lambda\left(\boldsymbol{\theta}^* || \boldsymbol{\theta}_\lambda(\boldsymbol{w})||\right)$, the following inequality can be drew:

$$\begin{aligned} &nB\ \log\left(1 + (C-1)\exp\left(-||\boldsymbol{\theta}_\lambda(\boldsymbol{w})||^a \gamma^*\right)\right) + \lambda ||\boldsymbol{\theta}_\lambda(\boldsymbol{w})||^r \\ \geq &b\ \log\left(1 + \exp\left(-||\boldsymbol{\theta}_\lambda(\boldsymbol{w})||^a \gamma_\lambda(\boldsymbol{w})\right)\right) + \lambda ||\boldsymbol{\theta}_\lambda(\boldsymbol{w})||^r. \end{aligned} \tag{A.58}$$

When $\lambda \to 0$, $||\boldsymbol{\theta}_\lambda(\boldsymbol{w})|| \to \infty$. Therefore, $\exp\left(-\left\|\boldsymbol{\theta}_\lambda(\boldsymbol{w})\right\|^a \gamma^*\right) \to 0$ and $\exp\left(-\left\|\boldsymbol{\theta}_\lambda(\boldsymbol{w})\right\|^a \gamma_\lambda(\boldsymbol{w})\right) \to 0$. The Taylor formula can be applied:

$$\begin{aligned} &nB(C-1)\exp\left(-||\boldsymbol{\theta}_\lambda(\boldsymbol{w})||^a \gamma^*\right) \geq b\exp\left(-||\boldsymbol{\theta}_\lambda(\boldsymbol{w})||^a \gamma_\lambda(\boldsymbol{w})\right) \\ &- O\left(\max\{\exp\left(-||\boldsymbol{\theta}_\lambda(\boldsymbol{w})||^a \gamma^*\right)^2, \exp\left(-||\boldsymbol{\theta}_\lambda(\boldsymbol{w})||^a \gamma_\lambda(\boldsymbol{w})\right)^2\}\right), \end{aligned} \tag{A.59}$$

which reveals $\gamma^* \leq \liminf_{\lambda \to 0} \gamma_\lambda(\boldsymbol{w})$. By contradiction, i.e., $\liminf_{\lambda \to 0} \gamma_\lambda(\boldsymbol{w}) \leq \gamma^*$. However, it is inconsistent with the occasion when $||\boldsymbol{\theta}_\lambda(\boldsymbol{w})|| >> \left(\log \frac{2(C-1)Bn}{b}\right)^{1/a}$. According to the definition of $\gamma_\lambda(\boldsymbol{w})$, we have $\gamma_\lambda(\boldsymbol{w}) \leq \gamma^*$. Thus, $\liminf_{\lambda \to 0} \gamma_\lambda(\boldsymbol{w})$ exists and equals to γ^*.

Proof of Optimization Accuracy Because $\mathcal{L}_\lambda$ is generally hard to precisely optimize for deep learning, we provide that even if the loss has not yet converged but is close enough to its optimal, the corresponding normalized margin has a reasonable lower bound.

Proof. Now we prove (2) in Theorem 1. Based on the cross entropy loss, considering $D \triangleq \left(\frac{1}{\gamma^\star}\log\frac{(C-1)(\gamma^\star)^{r/a}}{\lambda}\right)^{1/a}$, an upper bound of loss is

$$\begin{aligned}\mathcal{L}_\lambda\left(\boldsymbol{\theta}'(\boldsymbol{w})\right) &\leq \alpha\mathcal{L}_\lambda\left(\boldsymbol{\theta}_\lambda(\boldsymbol{w})\right)\\ &\leq \alpha\mathcal{L}_\lambda\left(D\boldsymbol{\theta}^*\right)\\ &\leq \alpha B\log\left(1+(C-1)\exp\left(-D^a\gamma^*\right)\right)+\alpha\lambda D^r\\ &\leq \alpha B(C-1)\exp\left(-D^a\gamma^*\right)+\alpha\lambda D^r\\ &\leq \alpha B\frac{\lambda}{(\gamma^*)^{r/a}}\left(1+\left(\log\frac{(C-1)(\gamma^*)^{r/a}}{\lambda}\right)^{r/a}\right)\\ &\triangleq \mathcal{L}^{(UB)}.\end{aligned} \tag{A.60}$$

The third step is based on Formula (A.53), and the fourth step is based on $\log(1+x)\leq x$. A lower bound can also be obtained:

$$\begin{aligned}\mathcal{L}_\lambda\left(\boldsymbol{\theta}'(\boldsymbol{w})\right) &\geq \frac{b}{n}\log\left(1+\exp\left(-\gamma'(\boldsymbol{w})||\boldsymbol{\theta}'(\boldsymbol{w})||^a\right)\right)\\ &\geq \frac{b}{n}\frac{\exp\left(-\gamma'(\boldsymbol{w})||\boldsymbol{\theta}'(\boldsymbol{w})||^a\right)}{1+\exp\left(-\gamma'(\boldsymbol{w})||\boldsymbol{\theta}'(\boldsymbol{w})||^a\right)}.\end{aligned} \tag{A.61}$$

The second step is based on the fact that $\log(x)\geq\frac{x}{1+x}$. Thus, combining Formulas (A.60) and (A.61), we obtain

$$\gamma'(\boldsymbol{w})\geq\frac{-\log\frac{\frac{n}{b}\mathcal{L}^{(UB)}}{1-\frac{n}{b}\mathcal{L}^{(UB)}}}{\left\|\boldsymbol{\theta}'(\boldsymbol{w})\right\|^a}. \tag{A.62}$$

Furthermore, it holds that $\lambda||\boldsymbol{\theta}'(\boldsymbol{w})||^r\leq\mathcal{L}^{UB}$. Now we note that

$$\begin{aligned}\mathcal{L}_\lambda\left(\boldsymbol{\theta}'(\boldsymbol{w})\right) &\leq \mathcal{L}^{(UB)}\\ &\leq 2\alpha B\frac{\lambda}{(\gamma^\star)^{r/a}}\left(\log\frac{(C-1)(\gamma^\star)^{r/a}}{\lambda}\right)^{r/a}\\ &\leq \frac{1}{2n},\end{aligned} \tag{A.63}$$

for sufficiently small λ. Thus, we obtain

$$\gamma'(\boldsymbol{w})\geq\frac{-\log\frac{\frac{n}{b}\mathcal{L}^{(UB)}}{1-\frac{n}{b}\mathcal{L}^{(UB)}}}{\left\|\boldsymbol{\theta}'(\boldsymbol{w})\right\|^a}\geq\frac{-\log\left(\frac{2n}{2b-1}\mathcal{L}^{(UB)}\right)}{\left\|\boldsymbol{\theta}'(\boldsymbol{w})\right\|^a}. \tag{A.64}$$

Then, we have

$$\begin{aligned}\gamma'(\boldsymbol{w}) &\geq \frac{-\lambda^{a/r}\log\frac{2n}{2b-1}\mathcal{L}^{(UB)}}{\left(\mathcal{L}^{(UB)}\right)^{a/r}} \\ &= \frac{-\log\left(2B\alpha\frac{n}{2b-1}\frac{\lambda}{(\gamma^*)^{r/a}}\left(1+\left(\log\frac{(C-1)\left(\gamma^*\right)^{r/a}}{\lambda}\right)^{r/a}\right)\right)}{\frac{B\alpha^{a/r}}{\gamma^\star}\left(1+\left(\log\frac{(C-1)(\gamma^*)^{r/a}}{\lambda}\right)^{r/a}\right)^{a/r}} \\ &= \frac{\gamma^*}{\alpha^{\frac{a}{r}}}\underbrace{\frac{-\log\left(2B\alpha\frac{n}{2b-1}\frac{\lambda}{(\gamma^*)^{r/a}}\left(1+\left(\log\frac{(C-1)\left(\gamma^*\right)^{r/a}}{\lambda}\right)^{r/a}\right)\right)}{B\left(1+\left(\log\frac{(C-1)(\gamma^*)^{r/a}}{\lambda}\right)^{r/a}\right)^{a/r}}}_{(I)}.\end{aligned} \tag{A.65}$$

This proof is adapted from Proposition 3 of [4]. Above lower bound indicates that the numerator is at the scale of $\log\left(\lambda\log\left(\frac{1}{\lambda}\right)\right)$ and the denominator is at the scale of $\log\left(\frac{1}{\lambda}\right)$. Thus, the difficulty-based weights only influence the value of the bound. (I) can be bounded by $\frac{1}{10}$, for sufficiently small λ. Thus, it can be drew that: $\gamma'(\boldsymbol{w}) \geq \frac{\gamma^*}{10\alpha^{a/r}}$.

This theorem indicates that the difficulty-based weights affect the convergence speed, which is reflected in $\mathcal{L}_\lambda\left(\boldsymbol{\theta}'(\boldsymbol{w})\right) \leq \alpha L_\lambda\left(\boldsymbol{\theta}_\lambda(\boldsymbol{w})\right)$. With a good set of weights, this criteria (by approaching global optimum) can be achieved faster.

D Supplementary Material for Section 4.2

We prove the generalization error can be bounded which is stated in Theorem 2 via Rademacher complexity and the margin theory. The Rademacher complexity stated by Golowich et al. [10] is adopted which is shown below:

For a real-valued function class $\mathcal{F}$, the empirical Rademacher complexity $\mathcal{R}_{n,\beta}(\mathcal{F})$ is as follows:

$$\mathcal{R}_{n,\beta}(\mathcal{F}) \triangleq \mathbb{E}_{\boldsymbol{\varepsilon}}\left[\sup_{f\in\mathcal{F}}\frac{1}{n}\sum_{i=1}^{n}\varepsilon_i\beta(\boldsymbol{x}_i)f(\boldsymbol{x}_i)\right], \tag{A.66}$$

where ε_i are independent Rademacher random variables. The classical theorem about the generalization error in terms of the Rademacher complexity and margin loss is shown in Theorem A.2.

Theorem A.2. *Given $\mathcal{F}$ a set of functions such that $\forall f\in\mathcal{F}$, $\sum_{\boldsymbol{x}\in\mathcal{X}}|f(\boldsymbol{x})| \leq L$. Then with a probability at least of $1-\delta$, for all margins $\gamma>0$, the following holds:*

$$\begin{aligned}P_{(\boldsymbol{x},y)\sim\mathcal{D}^{te}}\left(yf(\boldsymbol{x})\leq 0\right) \leq & \frac{1}{n}\sum_{i=1}^{n}\beta(\boldsymbol{x}_i)I\left(y_i f(\boldsymbol{x}_i)<\gamma\right) \\ & + 4\frac{R_{n,\beta}(\mathcal{F})}{\gamma} + \epsilon(\gamma,n,\delta),\end{aligned} \tag{A.67}$$

where $\epsilon(\gamma, n, \delta) = \sqrt{\frac{\log\log_2 \frac{4L}{\gamma}}{n}} + \sqrt{\frac{\log(1/\delta)}{n}}$. $\mathcal{R}_{n,\beta}(\mathcal{F})$ *is the weighted Rademacher complexity.*

Proof of Theorem 2

Proof. Let P_s and P_t be the source (training) and target (testing) distributions, respectively, with the corresponding densities of $p_s(\cdot)$ and $p_t(\cdot)$. Assume that the two distributions have the same support. The training and test samples are drawn *i.i.d* according to distributions P_s and P_t, respectively.

Learning with sample weights $\boldsymbol{w}(\boldsymbol{x})$ is equivalent to learning with a new training distribution $\widetilde{P}_s$. The density of the distribution of the weighted training set $\widetilde{P}_s$ is denoted as $\widetilde{p}_s(\boldsymbol{x}) \sim \boldsymbol{w}(\boldsymbol{x}) p_s(\boldsymbol{x})$. Pearson χ^2-divergence is used to measure the difference between $\widetilde{P}_s$ and P_t, i.e., $D_{\chi^2}(P_t \| \widetilde{P}_s) = \int [(d\widetilde{P}_s/dP_t)^2 - 1] d\widetilde{P}_s$. We consider depth-$q$ ($q \geq 2$) networks with the activation function ϕ. The binary setting is considered, in that the network computes a real value

$$f(\boldsymbol{x}) := \boldsymbol{W}_q \phi\left(\boldsymbol{W}_{q-1} \phi\left(\cdots \phi\left(\boldsymbol{W}_1 \boldsymbol{x}\right) \cdots\right)\right), \tag{A.68}$$

where $\phi(\cdot)$ is the element-wise activation function (e.g., ReLU). The training set contains n samples. Denote the generalization error for a network f as $\hat{\mathcal{L}}(f)$. Let $\beta(\boldsymbol{x}_i) = \frac{p_t(\boldsymbol{x}_i)}{\widetilde{p}_s(\boldsymbol{x}_i)}$. Each parameter matrix $W(j)$ has Frobenius norm at most $M_F(j)$. From Theorem 1 of Golowich et al. [10], we yield

$$n\mathcal{R}_{n,\beta}(\mathcal{F}) \leq \frac{1}{\lambda} \log\left(2^q \cdot \mathbb{E}_{\boldsymbol{\varepsilon}} \exp\left(M\lambda \| \sum_{i=1}^{n} \varepsilon_i \beta(\boldsymbol{x}_i) \boldsymbol{x}_i \|\right)\right), \tag{A.69}$$

with $M = \prod_{j=1}^{q} M_F(j)$. i.e., the sum of weight matrix of each layer. Define a stochastic variable:

$$Z = M \cdot \| \sum_{i=1}^{n} \varepsilon_i \beta(\boldsymbol{x}_i) \boldsymbol{x}_i \|. \tag{A.70}$$

Then the following equation holds:

$$\begin{aligned} n\mathcal{R}_{n,\beta}(\mathcal{F}) &\leq \frac{1}{\lambda} \log[2^q \cdot E \exp(\lambda Z)] \\ &= \frac{q \log 2}{\lambda} + \frac{1}{\lambda} \log[E \exp \lambda (Z - EZ) + EZ]. \end{aligned} \tag{A.71}$$

With the help of Jessen inequality, $E[Z]$ can be upper bounded by:

$$\begin{aligned} M\sqrt{\mathbb{E}_{\boldsymbol{\varepsilon}}\left[\| \sum_{i=1}^{n} \varepsilon_i \beta(\boldsymbol{x}_i) \boldsymbol{x}_i \|\right]} &= M\sqrt{\mathbb{E}_{\boldsymbol{\varepsilon}}\left[\sum_{i,i'=1}^{n} \varepsilon_i \varepsilon_{i'} \beta(\boldsymbol{x}_i)^2 \|\boldsymbol{x}_i\|^2\right]} \\ &= M\sqrt{\sum_{i=1}^{n} \beta(\boldsymbol{x}_i)^2 \|\boldsymbol{x}_i\|^2}. \end{aligned} \tag{A.72}$$

For the purpose of solving $\log[E\exp\lambda\left(Z-EZ\right)]$, we define a function Z over a set of *i.i.d.* random variables $\{\varepsilon_1,\varepsilon_2,...,\varepsilon_n\}$ which satisfies:

$$Z\left(\varepsilon_1,\cdots,\varepsilon_i,\cdots\varepsilon_m\right)-Z\left(\varepsilon_1,\cdots,-\varepsilon_i,\cdots\varepsilon_m\right)\leq 2M\beta\left(\boldsymbol{x}_i\right)||\boldsymbol{x}_i||. \tag{A.73}$$

It reveals that Z is bounded and is partially Gaussian, owning the following variance factor:

$$v=\frac{1}{4}\sum_{i=1}^{n}\left(2M\beta\left(\boldsymbol{x}_i\right)||\boldsymbol{x}_i||\right)^2=M^2\sum_{i=1}^{n}\beta(\boldsymbol{x}_i)^2||\boldsymbol{x}_i||^2. \tag{A.74}$$

The following formula satisfies:

$$\begin{aligned}\frac{1}{\lambda}\log[E\exp\lambda\left(Z-EZ\right)]\leq&\frac{1}{\lambda}\frac{\lambda^2M^2\sum\limits_{i=1}^{n}\beta(\boldsymbol{x}_i)^2||\boldsymbol{x}_i||^2}{2}\\=&\frac{\lambda M^2\sum\limits_{i=1}^{n}\beta(\boldsymbol{x}_i)^2||\boldsymbol{x}_i||^2}{2}.\end{aligned} \tag{A.75}$$

Let

$$\lambda=\frac{\sqrt{2\log(2)q}}{M\sqrt{\sum\limits_{i=1}^{n}\beta(\boldsymbol{x}_i)^2||\boldsymbol{x}_i||^2}}, \tag{A.76}$$

the upper bound is then found in the following form:

$$\begin{aligned}\frac{1}{\lambda}\log[2^q\cdot E\exp\left(\lambda Z\right)]\leq&EZ+\sqrt{2\log(2)q}\sqrt{\sum_{i=1}^{n}\beta(\boldsymbol{x}_i)^2||\boldsymbol{x}_i||^2}\\\leq&M\left(\sqrt{2\log(2)q}+1\right)\sqrt{\sum_{i=1}^{n}\beta(\boldsymbol{x}_i)^2||\boldsymbol{x}_i||^2}\\\leq&\sqrt{n}LM\left(\sqrt{2\log(2)q}+1\right)\sqrt{\frac{1}{n}\sum_{i=1}^{n}\beta(\boldsymbol{x}_i)^2},\end{aligned} \tag{A.77}$$

with $L:=\sup_{\boldsymbol{x}}||\boldsymbol{x}||$. Based on the law of large numbers, we have

$$\frac{1}{n}\sum_{i=1}^{n}\beta(\boldsymbol{x}_i)^2=D\left(P_t||\widetilde{P}_s\right)+1+o\left(\frac{1}{\sqrt{n}}\right). \tag{A.78}$$

Then, the desired result follows, and the generalization bound is

$$\begin{aligned}\mathbb{P}_{(\boldsymbol{x},y)\sim\mathcal{D}^{te}}\left(yf\left(\boldsymbol{x}\right)\leq 0\right)\leq&\frac{1}{n}\sum_{i=1}^{n}\beta\left(\boldsymbol{x}_i\right)\mathbb{1}\left(y_if\left(\boldsymbol{x}_i\right)<\gamma\right)\\&+\frac{L\sqrt{D_{\mathcal{X}^2}\left(P_t||\widetilde{P}_s\right)+1}}{\gamma\cdot q^{(q-1)/2}\sqrt{n}}+\epsilon\left(\gamma,n,\delta\right),\end{aligned} \tag{A.79}$$

where $\epsilon(\gamma, n, \delta) = \sqrt{\frac{\log\log_2 \frac{4L}{\gamma}}{n}} + \sqrt{\frac{\log(1/\delta)}{n}}$ is a small quantity compared to (I) and (II). Here, $L := \sup_{\boldsymbol{x}} ||\boldsymbol{x}||$. As $\beta(\boldsymbol{x}_i) = \frac{p_t(\boldsymbol{x}_i)}{\widetilde{p}_s(\boldsymbol{x}_i)}$, we can rewrite Formula (A.79) as

$$\begin{aligned}\hat{\mathcal{L}}(f) \leq & \underbrace{\frac{1}{n}\sum_{i=1}^{n}\frac{p_t(\boldsymbol{x}_i)}{\widetilde{p}_s(\boldsymbol{x}_i)}\mathbb{1}(y_i f(\boldsymbol{x}_i) < \gamma)}_{I} \\ & + \underbrace{\frac{L\cdot\sqrt{D_{\chi^2}\left(P_t \| \widetilde{P}_s\right)+1}}{\gamma\cdot q^{(q-1)/2}\sqrt{n}}}_{(II)} + \underbrace{\epsilon(\gamma, n, \delta)}_{(III)}.\end{aligned} \tag{A.80}$$

Thus, Theorem 2 is proved. This proof is adapted from Theorem 1 of Golowich et al. [10].

Proof of Proposition 6 In this section, we give the proof of Proposition 6.

Proof. Given a fixed training set, f depends on random variables (denoted as $\mathcal{V}$) such as hyper-parameters and initialization. The model trained under a given set of $\mathcal{V}$ is denoted as $f_{\mathcal{V}}$. Taking the expectation of the random variables $\mathcal{V}$ at both ends of Formula (A.80), we have

$$\mathbb{E}_{\mathcal{V}}[\hat{\mathcal{L}}(f_{\mathcal{V}})] \leq \frac{1}{n}\sum_{i=1}^{n}\frac{p_t(\boldsymbol{x}_i)}{\widetilde{p}_s(\boldsymbol{x}_i)}\mathbb{E}_{\mathcal{V}}[\mathbb{1}(y_i f_{\mathcal{V}}(\boldsymbol{x}_i) < \gamma)] + (II) + (III), \tag{A.81}$$

where the expectation of terms (II) and (III) are their own since the two terms are independent of $\mathcal{V}$.

Then, we proof that the a large error err_i indicates a large $\mathbb{E}_{\mathcal{V}}[\mathbb{1}(y_i f(\boldsymbol{x}_i) < \gamma)]$. Assume that the margins under random datasets and random variables obbey the Gaussian distribution, there are $\gamma_{i,\mathcal{V}} \sim \mathcal{N}(\mu_{i,\mathcal{V}}, {\sigma_{i,\mathcal{V}}}^2)$ and $\gamma_{i,T} \sim \mathcal{N}(\mu_{i,T}, {\sigma_{i,T}}^2)$. As the generalization error is often computed using methods such as cross-validation, the random training sets can be assumed to obbey the Gaussian distribution $\mathcal{N}(T, \delta I)$. Therefore, the two distributions (i.e., $\mathcal{N}(\mu_{i,T}, {\sigma^2}_{i,T})$ and $\mathcal{N}(\mu_{i,\mathcal{V}}, \sigma^2_{i,\mathcal{V}})$) are similar because training models on different datasets are essentially different in parameters and the random training sets conform the Gaussian distribution . According to the moment-generating function, we have

$$\mathbb{E}_{\mathcal{V}}\left[e^{-\gamma_{i,\mathcal{V}}}\right] \approx \mathbb{E}_T\left[e^{-\gamma_{i,T}}\right] = e^{-\mu_{i,T}+\frac{1}{2}\sigma^2_{i,T}}. \tag{A.82}$$

Therefore, when all samples' distributions of margins have the same margin variances, a large generalization error $\mathrm{err}_i = \mathbb{E}_T[e^{-\gamma_i(T)}]$ indicates a small $\mu_{i,T}$. For a fixed γ in Formula (A.81), for two samples $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, if $\mu_{i,T} \leq \mu_{j,T}$, then it is obvious that $p(\gamma_{i,T} < \gamma) \geq p(\gamma_{j,T} < \gamma)$. Thus, we have

$$\begin{aligned}\mathbb{E}_{\mathcal{V}}[\mathbb{1}(\gamma_{i,\mathcal{V}} < \gamma)] &\approx \mathbb{E}_T[\mathbb{1}(\gamma_{i,T} < \gamma)] \\ &= p(\gamma_{i,T} < \gamma)\times 1 + (1 - p(\gamma_{i,T} < \gamma))\times 0 \\ &= p(\gamma_{i,T} < \gamma),\end{aligned} \tag{A.83}$$

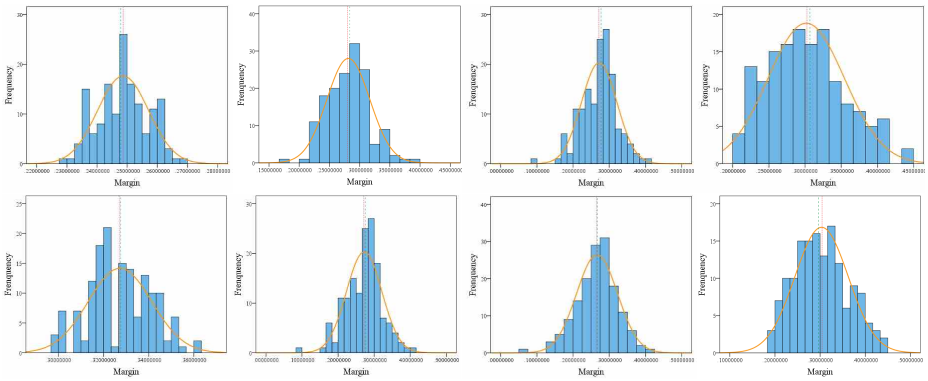

Fig. A-2: The distribution curves of the margins of eight samples on different datasets. Red and green lines refer to the mean of the distribution and the true margin, respectively.

and

$$\begin{aligned}\mathbb{E}_{\mathcal{V}}[\mathbb{1}\left(\gamma_{j,\mathcal{V}} < \gamma\right)] &\approx \mathbb{E}_{T}[\mathbb{1}\left(\gamma_{j,T} < \gamma\right)] \\ &= p(\gamma_{j,T} < \gamma) \times 1 + (1 - p(\gamma_{j,T} < \gamma)) \times 0 \\ &= p(\gamma_{j,T} < \gamma).\end{aligned} \tag{A.84}$$

Therefore, we have

$$\mathbb{E}_{\mathcal{V}}[\mathbb{1}\left(\gamma_{i,\mathcal{V}} < \gamma\right)] \gtrsim \mathbb{E}_{\mathcal{V}}[\mathbb{1}\left(\gamma_{j,\mathcal{V}} < \gamma\right)]. \tag{A.85}$$

According to above analyses, it can be drew that if $\mathrm{err}_i > \mathrm{err}_j$, then $\mathbb{E}_{\mathcal{V}}[\mathbb{1}(y_i f_{\mathcal{V}}(\boldsymbol{x}_i) < \gamma) \gtrsim \mathbb{E}_{\mathcal{V}}[\mathbb{1}(y_j f_{\mathcal{V}}(\boldsymbol{x}_j) < \gamma)$.

E Experiments for the Effectiveness of Generalization Error

F More Experimental Results

In this section, we present more experimental results and discussions.

F.1 Distributions of margin

The Z-scores of the distributions' Kurtosis and Skewness are used to examine if the distributions of margin obey Gaussian distribution. Fig. A-2 shows the distribution curves of the margins of eight samples on different datasets. In addition, Table A-1 demonstrates the Z-scores of skewness and kurtosis of these eight distributions. As all Z-scores are in $[-1.96, 1.96]$, under the test level of $\alpha = 0.05$, the margin distribution obeys the Gaussian distribution.

Table A-1: Z-scores of skewness and kurtosis of these eight distributions. The order is top to bottom and left to right.

Id	1	2	3	4	5	6	7	8
Z-score(skewness)	-0.035	0.227	0.969	-0.267	-1.131	-1.641	1.712	1.257
Z-score(kurtosis)	-1.644	-1.238	1.352	1.555	1.710	1.847	1.005	1.675

F.2 Experiments for the Increasing Weights of Hard Samples on the Simulated Data

In this section, we increase the weights of the samples with small margins, samples in small categories, and noisy samples, as they are hard ones that have been analyzed in Section 3. The experimental results are shown in Fig. A-3. The cosine distances are all increasing to 1 indicating that the angle between the decision boundary and the max-margin solution is decreasing to 0. Thus, the finding reveals that the directions of the parameters (for the linear predictor) and the normalized margin (for the nonlinear predictor) converge to the max-margin solution. As shown in Fig. A-3, increasing the weights of the hard samples (samples with large errors) increases the convergence speed of both the linear and nonlinear predictors. The results are consistent with our theoretical analysis for linear predictors in Section 4.1. Deeper analyses should be conducted for the nonlinear case to explore the conditions in which the difficulty-based weights can accelerate the speed.

F.3 Accuracy, Loss and Margin of the CIFAR10 Data in the Training Process

Fig. A-4 shows the epoch-wise training performances measured by accuracy, loss, and margin, using the ResNet32 on CIFAR10 data. The margin shows an increasing trend during the training.

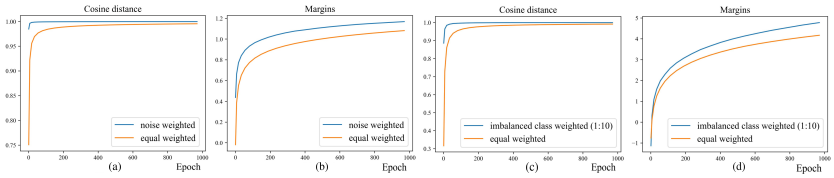

Fig. A-3: “Cosine distance" represents the cosine of the angle between the decision boundary (at that epoch) and the max-margin solution. (a), (b) Cosine distance and average margin of equal weights and increasing weights of noisy samples using the linear predictor on noisy data. (c), (d) Cosine distance and average margin of equal weights and increasing weights of samples in the small category using the nonlinear predictor on imbalanced data. Uniform label noise is adopted. The noise ratio and imbalance ratio are 20% and 10:1. Other noise and imbalance settings are also used and the same conclusions are obtained.

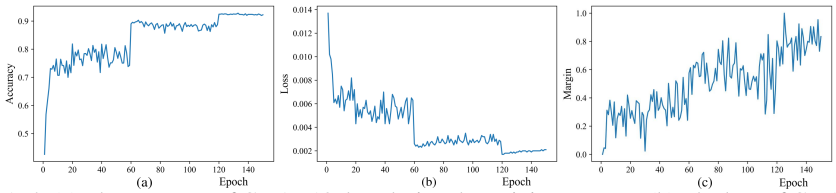

Fig. A-4: (a) The accuracy of CIFAR10 data during the training process. (b) The loss of CIFAR10 data during the training process. (c) The margin of CIFAR10 data during the training process.

F.4 Model Performance Pertaining to Noisy Data with Different Difficulty-Based Weights

We add 10% noise to the simulated data and compare the accuracies under equal weights, large weights on noisy samples, and small weights on noisy samples. The results are shown in Fig. A-5, indicating that assigning small weights on noisy samples, alternatively, the easy-first mode can achieve the best performance. The worst performance is observed when the weights of noisy samples are by increased, which belongs to the hard-first scheme. Therefore, it reveals that although increasing the weights of the hard samples may increase the convergence speed, it is not always the optimal strategy.

References

1. Lyu, K., Li, J.: Gradient Descent Maximizes the Margin of Homogeneous Neural Networks. arXiv preprint arXiv:1906.05890 (2019)
2. Fazlyab, M., Robey, A., Hassani, H., Morari, M., Pappas, G.: Efficient and accurate estimation of lipschitz constants for deep neural networks. In: NeurIPS, pp. 11427—11438. NeurIPS foundation, America (2019)
3. Virmaux, A., Scaman, K.: Lipschitz regularity of deep neural networks: analysis and efficient estimation. In: NeurIPS, pp.3835–3844. NeurIPS foundation, America (2018)
4. Xu, D., Ye, Y., Ruan, C.: Understanding the role of importance weighting for deep learning. In: ICLR, pp. 1–20. ICLR foundation, America (2020)

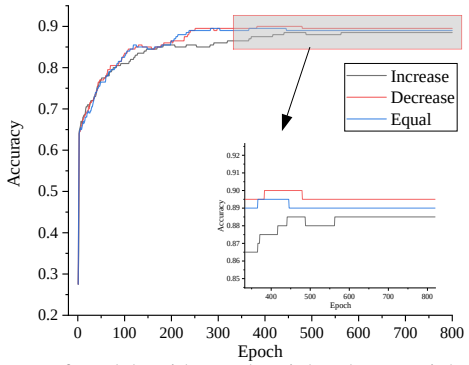

Fig. A-5: The performances of models with equal weights, large weights for noisy samples, and small weights for noisy samples. Simulated data is used here and the noise ratio is set to 10%.

5. Gal, Y., Ghahramani, Z.: Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In: ICML, pp. 1050–1059. International Machine Learning Society, America (2016)
6. Abdar, M., Pourpanah, F., Hussain, S., Rezazadegan, D., Liu, L., Ghavamzadeh, M., Fieguth, P., Cao, X., Khosravi, A., Acharya, U.-R., Makarenkov, V., Nahavandi, S.: A review of uncertainty quantification in deep learning: Techniques, applications and challenges. Information Fusion **76**(1), 243–297 (2021)
7. Yang, Z., Yu, Y., You, C., Jacob, S., Yi, M.: Rethinking bias-variance trade-off for generalization of neural networks. In: ICML, pp. 10767–10777. International Machine Learning Society, America (2020)
8. Soudry, D., Hoffer, E., Nacson, M.-S., Gunasekar, S., Srebro, N.: The implicit bias of gradient descent on separable data. Journal of Machine Learning Research **19**(1), 1–14 (2018)
9. Wei, C., Lee, J.-D., Liu, Q., Ma, T.: Regularization Matters: Generalization and Optimization of Neural Nets v.s. their Induced Kernel. In: NeurIPS, pp. 1–14. NeurIPS foundation, America (2019)
10. Golowich, N., Rakhlin A., Shamir, .: Size-independent sample complexity of neural networks. Information and Inference: A Journal of the IMA **9**(2), 473–504 (2020)