Machine Learning (2022) 111:1073–1101
https://doi.org/10.1007/s10994-021-06087-3

Switching: understanding the class-reversed sampling in tail sample memorization

Chi Zhang[1,2] · Benyi Hu[1,2] · Yuhang Liuzhang[2] · Le Wang[1,2] · Li Liu[3] · Yuehu Liu[1,2]

Received: 19 May 2021 / Revised: 13 August 2021 / Accepted: 22 September 2021 /
Published online: 7 January 2022

Abstract

Long-tailed visual recognition poses significant challenges to traditional machine learning and emerging deep networks due to its inherent class imbalance. Existing reweighting and re-sampling methods, although effective, lack a fundamental theory while leaving the paradoxical effects of long tail unsolved, where network failing with head classes under-represented and tail classes overfitted. In this paper, we investigate long-tailed recognition from a memorization-generalization point of view, which not only unravels the whys of previous methods, but also derives a new principled solution. Specifically, we first empirically identify the regularity of classes under long-tailed distributions, finding that *long-tailed challenge is essentially a trade-off between the representation of high-regularity head classes and generalization to low-regularity tail classes*. To memorize tail samples without seriously damaging the representation of head samples, we propose a simple yet effective sampling strategy for ordinary mini-batch SGD optimization process, *Switching*, which switches from instance-balanced sampling to class-reversed sampling for only once at small learning rate. By theoretical analysis, we show that the upper bound on the generalization error of the proposed sampling strategy is lower than instance-balanced sampling conditionally. In our experiments, the proposed method can reach feasible performance more efficiently than current methods. Further experiments validate the superiority of the proposed *Switching* strategy, implying that the long-tailed learning trade-off could be parsimoniously tackled only in the memorization stage with a small learning rate and over-exposure of tail samples.

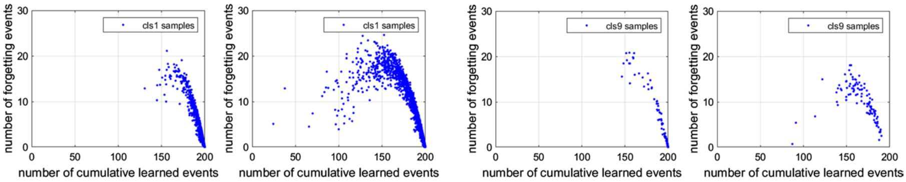
Keywords Long-tailed classification · Network memorization and generalization · Class-reversed sampling · Rademacher complexity

Chi Zhang and Benyi Hu have contributed equally.

Editors: Yu-Feng Li, Mehmet Gönen, Kee-Eung Kim.

✉ Chi Zhang
chizhang@xjtu.edu.cn

Extended author information available on the last page of the article



(a) Cls 1 (3237 samples, Cardinality: 5000→3237) **(b)** Cls 9 (100 samples, Cardinality: 5000→100)

Fig. 1 The visualization of the regularity degradation of selected training samples when training set changing from standard CIFAR-10 to Long-tailed CIFAR-10 with class cardinality reduced. The regularity of one class will be higher with more samples gathering in the lower right corner of the picture. In each subfigure, samples are in one-to-one correspondence within two plots. It could be observed that *regularity of the same training samples will be sharply decreased with the reduction of class cardinality.* The more cardinality reduced, the more regularity decreased

1 Introduction

With the prosperity of deep learning research field, visual recognition has witnessed the prominence of powerful representation learning approaches and high-quality, large-scale datasets, e.g., ImageNet ILSVRC (Russakovsky et al., 2015) and Places (Zhou et al., 2018). These datasets are usually carefully balanced, exhibiting roughly uniform distributions of class labels. However, visual phenomena in real world tends to have skewed data distributions with long-tailed characteristics (Dong et al., 2017; Liu & Tsoumakas, 2018; Xiang & Ding, 2020; Bej et al., 2021), consisting of a few majority classes (*head classes*) and a large number of minority classes (*tail classes*). When dealing with such long-tailed data, many standard approaches fail to work well due to the extreme class imbalance trouble, leading to a significant drop in accuracy for tail classes (Mollaysa et al., 2019).

A common way to solve long-tailed problem is re-sampling or re-weighting, which artificially generates class-balanced batch or loss to avoid extreme long tail (Huang et al., 2016; Buda et al., 2018; Ma et al., 2018; Cao et al., 2019). Inspired by the phenomenon that naively re-weighting or re-sampling inevitably causes under-fitting to the head or overfitting to the tail, latest studies (Kang et al., 2020; Zhou et al., 2020) separate the imbalanced feature learning and balanced classifier learning, leading a two-stage training paradigm. Each of these strands is intuitive, and has proven empirically successful. However, they are not without limitation: no explanations about wherefores that the data sampler of feature extractor learning and classifier learning should be different. On the other hand, imbalanced feature representation will project on the head feature direction due to head classes always dominate training procedure, resulting in the re-trained classifier biased (Zhou et al., 2020).

In this paper, we propose to investigate long-tailed recognition from a memorization-generalization point of view. A recent study (Jiang et al., 2020) suggests rare and low-regularity samples could be learned based on the internal representations built from strongest-domain regularities first. In this case, the relation between sample regularity and cardinality of each class during training is verified. We visualize the cumulative learned events and forgetting events (Toneva et al., 2019) of each sample (see Appendix A) and find that *regularity of the same training samples will be sharply decreased with the reduction of class cardinality.* As shown in Fig. 1, the more class cardinality reduced, the more regularity decreased. Further, it is shown by the comparison between Figs. 9 and 10 that such skewed regularity of the training samples may cause the network generalization degraded, i.e., the less training samples one class have, the lower regularity the validation samples of the same class will own (see Appendix A).

Based on the notion that *long-tail challenge is essentially a trade-off between the representation of high-regularity head classes and generalization to low-regularity tail classes*, we explore a simple yet effective joint training strategy, named *Switching*, which properly shifts learning focus from high-regularity head classes to low-regularity tail classes and give the theoretical generalization bound of changing data samplers during training for the first time.

Specifically, we employ the standard training procedure with cross-entropy loss and instance-balanced sampler w.r.t. the original data distribution to ensure the learning of universal visual patterns. We only switch from instance-balanced sampler to class-reversed sampler for the last several epochs of training, tending tail classes to be over-exposed. In earlier training, with head classes dominate the training data, the patterns and structures discovered in regular examples are utilized to build a generalizable representation. In later training phase, the memorization of tail classes will not seriously disrupt the learned representation as the learning rate is much smaller than earlier stages. Such strategy can simultaneously boost the representation and classification towards long-tailed distributions, avoiding the risk of re-trained classifier excessive dependent on feature extractor.

We conduct extensive experiments across four benchmark long-tailed datasets: CIFAR10-LT, CIFAR100-LT, iNaturalist 2018 and ImageNet-LT, to evaluate the effectiveness of our proposed method. With such a simple training strategy, we obtain comparable or better results more efficiently compared with previous state-of-the-art methods.

To summarize, the main contributions are as follows:

- We empirically identify that the *low-regularity of tail classes* is the primary hurdle for learning an accurate model for long-tailed distributions and appropriately memorizing them is essential for better generalization across all classes.
- We propose a simple yet effective strategy, named *Switching*, to handle the trade-off between high-regularity head classes and low-regularity tail classes and give the theoretical generalization error bound proving that class-reversed sampling is better than instance-balanced sampling during the last training stage.
- We investigate the effectiveness and efficiency of the proposed method through extensive experimentation and demonstrate that tackling long-tail trade-off could only cost a few training epochs with a small learning rate and over-exposure of tail samples.

2 Related work

2.1 Long-tailed visual recognition

Re-sampling strategies Re-sampling strategies can be divided into two classical types: over-sampling the minority classes by repeatedly adding augmented images (Drummond et al., 2003; Han et al., 2005; Buda et al., 2018); or under-sampling the majority classes by removing several images (Japkowicz & Stephen, 2002; He & Garcia, 2009; Bellinger et al., 2018). All these re-sampling methods tend to provide a more balanced data distribution during training to solve the long-tailed problem. However, over-sampling may sometimes cause over-fitting towards minority classes, while under-sampling may weaken the representation ability of networks.

Re-weighting losses Re-weighting methods usually allocate different weights for training samples of each class to re-balance data distribution (Huang et al., 2016; Cao et al., 2019; Wu et al., 2020). Cui et al. (2019) assigns weights to each class based on the effective numbers of samples instead of the proportional frequency. Further, Jamal et al. (2020) utilizes both effective numbers (Cui et al., 2019) and conditional weights to augment the classic class-balanced learning by explicitly estimating the differences between the class-conditioned distributions with a meta-learning approach.

Two-stage fine-tuning Various methods (Ouyang et al., 2016; Cao et al., 2019; Liu et al., 2019; Peng et al., 2020) are proposed to modify re-balancing for further improvements in long-tailed recognition. These methods usually separate training process into two single stages. In general, they train the networks with instance-balanced sampling in the first stage and exploit re-sampling or re-weighting methods at the second stage to fine-tune the network. More radically, Kang et al. (2020) re-train the classifier from scratch in a class-aware manner in the second stage with backbone fixed.

Different from them, we provide the theoretical analysis on the upper bound of generalization error for switching data samplers. Based on this, we do not artificially generate class-balanced batches or losses; instead, we simply emphasize the memorization of low-regularity tail class samples by only switching from the instance-balanced sampler to class-reversed sampler during the standard training procedure.

2.2 Memorization-generalization mechanism in deep learning

Memorization was once considered a failure of deep networks since it implies a lack of generalization. However, the view that memorization is harmful may be a misunderstanding towards deep learning. Zhang et al. (2017) was the first to demonstrate that standard deep learning algorithms can achieve high training accuracy even on large and randomly labeled datasets, leading a large wave of research interest in the topic of generalization for deep learning. Toneva et al. (2019) introduced the “forgetting event” to describe the learning dynamics of neural networks, where some instances flip flop between “learned” and “forgotten” states during training. In order to analyze how individual instances are treated by a model on the memorization-generalization continuum, Jiang et al. (2020) proposed the C-score to measure the consistency of a sample with respect to the rest of the training set. They found that samples having lower C-scores are learned more slowly, indicating the need for a stage-wise learning rate schedule during training.

A recent work of Feldman’s (2020) proposed a new theoretical explanation for the benefits of memorization. In their abstract model, algorithm can only get the frequency of a subpopulation through the empirical frequency of its representatives, thus it can only avoid the risk of missing subpopulations with significant frequency by memorizing examples. Further, Feldman and Zhang (2020) introduced the influence estimation to validate the necessity of memorizing useful examples for achieving close-to-optimal generalization error.

3 Method

Long-tailed visual recognition follows a long-tailed distribution over classes, leading model to exhibit under-fitting on tail classes and over-fitting to head classes (Tao et al., 2018; Baloch et al., 2019). Since increasing the exposure of tail classes may lead to over-fitting while under-sampling head classes may weaken the representation ability of networks, the trade-off between the representation of head and generalization towards tail becomes the main dilemma in long-tailed problem. To solve this dilemma, we first introduce the cumulative learned and forgetting events (Toneva et al., 2019) to verify the relation between cardinality and regularity. Based on the fact that *regularity of the same training samples will be sharply decreased with the reduction of class cardinality* (see Appendix A), we propose the *Switching* training strategy by only switching the standard instance-balanced sampler to a class-reversed sampler during the last training procedure, in order to learn low-regularity samples (tail classes) without seriously disrupting the representation of the strongest domain regularities (head classes) first.

3.1 Theoretical motivations

Problem setup and notations Let $f_\theta(\cdot)$ denote a feature extractor implemented by a CNN model with parameter θ, we get the class prediction through $\hat{y} = \arg\max g(f_\theta(\mathbf{x}))$, where $\mathbf{x}$ is the input image and $g(\cdot)$ is a classifier function. Given a training set $\mathcal{D} = \{x_i, y_i\}, i \in \{1, ..., n\}$ with C classes, let n_j denote the number of samples for class j and $n = \sum_{i=1}^{C} n_i$ be the total number of samples. Without loss of generality, we assume classes are sorted by cardinality in decreasing order, *i.e.*, if $i < j$, then $n_i \geq n_j$. For most sampling strategies, the probability p_j of sampling a data point from class j is given by:

$$p_j = \frac{n_j^q}{\sum_{i=1}^{C} n_i^q}, \tag{1}$$

with different values of q arise for different sampling strategies. The sampling of each data can be capsuled into the following two steps: 1) Randomly sample a class according to p_j; 2) Uniformly pick up a sample from class j. Sampling strategies that corresponding to q = 1, q = 0, and q = -1 are introduced as below:

Instance-balanced sampling (IB) This is the most common and standard way of sampling data, where each sample of the training dataset is sampled only once with equal probability in a training epoch. For instance-balanced sampling, the probability p_j^{IB} is given by Eq. 1 with q = 1, *i.e.*, a sample from class j will be sampled proportionally to the cardinality n_j of the class.

Class-balanced sampling CB To alleviate the extreme data imbalance during training, class-balanced sampling is proposed to artificially generate class-balanced data. The probability p_j^{CB} is given by Eq. 1 with q = 0, *e.g.*, $p_j^{CB} = 1/C$. In this scenario, the probability of each class j being selected is equal, independent to its cardinality n_j.

Class-reversed sampling (CR) Zhou et al. (2020) utilizes the reversed sampler to re-balance feature representation and particularly improve the classification accuracy on tail classes. Here we integrate p_j^{CR} into Eq. 1 with q = -1. For class-reversed sampling, a data point from class j will be sampled proportionally to the reciprocal of its cardinality n_j, i.e., the more samples in a class, the smaller sampling possibility that class has.

Objective function Let $L_{ji}(\theta)$ denote standard training error on i-th sample of class j:

$$L_{ji}(\theta) = \ell\left(f_\theta\left(x_{ji}\right), y_{ji}\right), \tag{2}$$

where ℓ is the loss function, *e.g.*, cross-entropy loss.

For standard training process with IB sampling, where each sample is sampled with equal probability, the objective function over the total training set $\mathcal{D}$ is given as follows:

$$L^s(\theta) = \frac{1}{n} \sum_{i=1}^{n} L_i(\theta) + R(\theta), \tag{3}$$

where $R(\theta)$ is the regular terms.

Now considering a more general scene, where sampling a data containing two steps: (1) Randomly chooses one class according to p_j; 2) Uniformly pick up one sample from its n_j samples, we have the following objective function:

$$L(\theta) = \sum_{j=1}^{C} \sum_{i=1}^{n_j} \frac{p_j}{n_j} L_{ji}(\theta) + R(\theta). \tag{4}$$

Generalization error upper bound Now we give the generalization analysis for such an objective function by deriving its generalization error upper bound. Let Θ be the family function of our learned neural network, we define $\mathfrak{R}_{\mathrm{n}}(\Theta)$ as the standard Rademacher complexity (Bartlett & Mendelson, 2002) of the set $\{(x, y) \mapsto \ell(f(x;\theta), y) : \theta \in \Theta\}$:

$$\mathfrak{R}_n(\Theta) = \mathbb{E}_{\mathcal{D},\xi}\left[\sup_{\theta \in \Theta} \frac{1}{n} \sum_{i=1}^{n} \xi_i \ell\left(f_\theta\left(x_{ji}\right), y_{ji}\right)\right], \tag{5}$$

where $\xi_1, \dots, \xi_n$ are in-dependent uniform random variables taking values in $\{-1,1\}$ (*i.e.* Rademacher variables).

Let $\mathcal{M}$ denote the least upper bound on the difference of individual loss values: $\left|\ell(f_\theta(x), y) - \ell\left(f_\theta\left(x'\right), y'\right)\right| \le \mathcal{M}$ for all $\theta \in \Theta$. For the standard training process with $L^s(\theta)$, for any $\delta > 0$, with probability at least $1 - \delta$ over the training set $\mathcal{D}$, the following error bound holds for all $\theta \in \Theta$ (Kawaguchi & Lu, 2020):

$$\mathbb{E}^s_{(x,y)}[\ell(f_\theta(x), y)] \le L^s(\theta) + 2\mathfrak{R}_n(\Theta) + \mathcal{M}\sqrt{\frac{\ln(1/\delta)}{2n}}. \tag{6}$$

Analogously, for the general objective function L_θ, we have the following error bound for all $\theta \in \Theta$ (the proof is given in Appendix B.1):

$$\begin{aligned}\mathbb{E}_{(x,y)}[\ell(f_\theta(x), y)] \le{}& L(\theta) + 2\mathfrak{R}_n(\Theta) - \mathcal{Q}_n(\Theta;p, n)\\ &+\mathcal{M}\sqrt{\sum_{j\in C} \frac{p_j^2}{n_j}}\sqrt{\frac{\ln(1/\delta)}{2}},\end{aligned} \tag{7}$$

where $\mathcal{Q}_n(\Theta;p, n) = \mathbb{E}_{\mathcal{D}}\left[\inf_{\theta\in\Theta} \sum_{j=1}^{C} \sum_{i=1}^{n_j} \left(\frac{p_j}{n_j} - \frac{1}{n}\right) L_{ji}(\theta)\right]$, a residual term which measures the expectation of the minimum difference between the empirical value of the training error of the proposed Switching method and that of Instance-balanced resampling method (IB) under the global distribution $\mathcal{D}$.

With the above derivation, we have the following Theorem 1 to serve as a theoretical evidence supporting the superiority of the generalization of the proposed method.

Theorem 1 *With a small size of Θ and a bounded $\mathcal{M}$, the upper bound on the expected error for CR is strictly* **lower** *than IB if $\mathcal{Q}_n(\Theta;p, n) + L^s - L > 0$ or if $L^s - L > 0$ (the proof is given in Appendix B.2).*

In our experimental settings, a small learning rate is adopted in the last training stage, which is equivalent to fine-tuning on a pre-processed initial value to produce a narrow parameter space Θ (See the first assumption in Appendix B.2). Therefore, Theorem 1 can theoretically guarantee that the upper bound of the CR method used in the small learning rate stage is strictly lower than that of the IB training method.

3.2 Switching data samplers during training

Algorithm 1 The *Switching* algorithm

```
Require: $D$: Training Set;   $M$: Mini-batch size;
Require: $T$: Total epochs;   $S$: Switch epoch;
Require: $[m_1, ..., m_n]$: Learning rate decay milestones;
initial $p_j^{CR} = \frac{n_j^{-1}}{\sum_{i=1}^{C} n_i^{-1}}$;
for $t \in [1, T]$ do
    $B = \{\}$;
    if $t \le m_n + S$ then
        Randomly pick M samples from $D$ as $B$.
    else
        for $i \in [1, M]$ do
            Sample class $j$ from the $D$ according to $p_j^{CR}$.
            Uniformly pick up a sample $\{x_i, y_i\}$ from class $j$ without replacement.
            $B = \{B; \{x_i, y_i\}\}$.
        end for
    end if
    Optimize network by SGD based on $B$.
end for
```

Switching is proposed to shift the learning focus from head classes to tail classes by simply switching the IB sampler to CR sampler at some epoch during training. Before the switching happens, the uniform IB sampler retains the characteristics of original distributions and almost the high-regularity samples from head classes are learned, the patterns and structures discovered in those head class samples can be used to build a generalizable representation. In later stages, the memorization of tail class samples will not seriously disrupt the learned representation as the learning rate is much smaller than the earlier stages.

Concretely, the number of total training epochs is denoted as T and the learning rate milestones are denoted as $[m_1, \dots, m_n]$, where $m_1 < \cdots < m_n \le T$. Let $\gamma \in (0, 1)$ becomes the multiplicative factor, learning rate will be decayed by γ once the epoch reaches one of the learning rate milestones during training. When training procedure reaches the $m_n + S$ epoch, we switch IB sampling to CR sampling and continuing training, where S is the hyper-parameter in our method indicating when to switch. The details of our switching strategy are shown in Algorithm 1 and illustrated by Fig. 2.

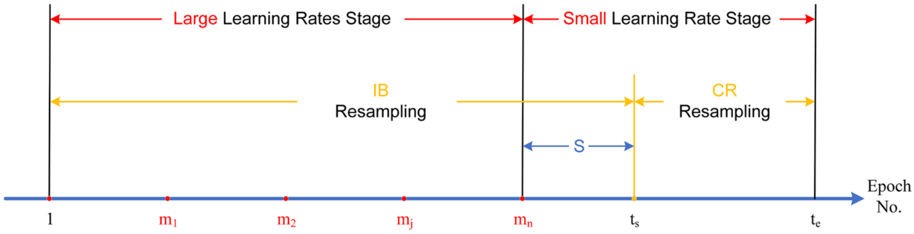


Fig. 2 The illustration of Algorithm 1. m_j denotes the j-th exact moment to decay learning rate (j-th learning rate decay milestone), m_n denotes the last learning rate decay milestone, t_s denotes the exact moment to switch Resampling strategy and t_e denotes the ending epoch

Our method is simple and clean, which only switches the data sampler from IB to CR once during training, without changing any structure of the original network or artificially generating class-balanced batches or losses.

4 Experiments

4.1 Datasets and empirical settings

Long-tailed CIFAR-10 and CIFAR-100. Both CIFAR-10 and CIFAR-100 contain 60,000 images, with 50,000 for training and 10,000 for validation with category number of 10 and 100, respectively. For fair comparisons, we use the long-tailed versions of CIFAR datasets as the same as those used in Zhou et al. (2020) with controllable degrees of data imbalance. Imbalance factor β is utilized to describe the severity of the long tail problem with the number of training samples for the most frequent class and the least frequent class, *e.g.*, $\beta = \frac{n_{max}}{n_{min}}$. We use β as 10, 50, and 100 in our experiments.

iNaturalist 2018 The iNaturalist species classification dataset is a large-scale real-world, naturally long-tailed dataset, suffering from extremely imbalanced label distributions. We choose the 2018 version in our experiments, which consists of 437,513 images from 8142 categories. Note that, besides the extreme imbalance, the iNaturalist datasets also face the fine-grained problem. For fair comparisons, we utilize the official splits of training and validation images.

ImageNet-LT ImageNet-LT is artificially truncated from their balanced versions so that the labels of the training set follow a long-tailed distribution. ImageNet-LT has 1000 classes and the number of images per class ranges from 1280 to 5 images. Note that the validation set is balanced of 1000 classes.

4.2 Implementation details

Implementations details on CIFAR We adopt the plaining ResNet-32 (He et al., 2016) as our model in all experiments. Standard mini-batch stochastic gradient descent (SGD) with momentum of 0.9, weight decay of 2×10^{-4} is utilized to optimize the whole network. We train all the models on one single NVIDIA 2080Ti GPU for 200 epochs with batch size of 64. The initial learning rate is set to 0.1 and the first five epochs is trained with the linear warm-up learning rate schedule (Goyal et al., 2017). The learning rate is decayed at the 100th by 0.1. S is set to 1, which means we switch the instance-balanced sampling to class-reversed sampling at the 101st epoch.

Implementations details on iNaturalist For fair comparisons, we utilize the plaining ResNet-50 (He et al., 2016) as our network in all experiments. We train all the models on eight NVIDIA 2080Ti GPUs with batch size of 512 for 90 epochs and 200 epochs, respectively. The initial learning rate is set to 0.05 and decayed by 0.1 at the 60th and 80th epoch for 90, 120th and 160th for 200. The batch size is 512 and S is set to 1, which is similar to experiments on CIFAR. For fair comparison with Decouple (Kang et al., 2020), we also set the S as 10 and 40 respectively, which means to train it for additional 10 epochs after the same standard training procedure have done, with total number of training epochs as 100 epochs and 210 epochs.

Implementations details on ImageNet-LT We adopt ResNet-50 and ResNext-50 as our backbone to analyze the effectiveness of our method. The initial learning rate is set to 0.2 and decayed by 0.1 at the 60th and 80th epoch for total 90 epochs. The batch size is 256 and S is set to 1, which is similar to experiments on CIFAR. For fair comparison with Decouple (Kang et al., 2020), we also set the S as 10, which means to switch sampler and train it for additional 10 epochs after the same standard training procedure have done, with total number of training epochs as 100 epochs.

4.3 Comparison methods

In experiments, we compare our method with four groups of methods:

Baseline methods We employ plaining training with cross-entropy loss and focal loss (Lin et al., 2017) as our baselines.

Re-weighting methods For re-weighting methods, we compare with the CB-Focal (Cui et al., 2019) and LDAM (Cao et al., 2019), where effective numbers or margin-based generalization are utilized to alleviate the extreme data imbalance during training.

Two-stage fine-tuning strategies To prove the effectiveness of our switching strategy, we compare it with two-stage fine-tuning strategies proposed in Cao's work (2019). Networks are trained with cross-entropy (CE) on imbalanced data first, and then are trained with class re-balancing strategy in the second stage. CE-DRW and CE-DRS refer to the two-stage baselines using re-weighting and re-sampling at the second stage. We also compare with Decouple (Kang et al., 2020), which trains network with instance-balanced sampling and uses class-balanced sampling to re-train classifier in the second stage with backbone fixed.

State-of-the-art methods For state-of-the-art methods, we compare with the recently proposed BBN (Zhou et al., 2020), which utilizes class-reversed sampling to re-balance the feature extractor but has a more complicated model structure, neglecting the proper combination of different data samplers itself.

4.4 Overall performance

In this section, we compare the performance of the proposed scheme to other recent works that report state-of-the-art results on four common long-tailed benchmarks: Long-tailed CIFAR-10, Long-tailed CIFAR-100, iNaturalist2018 and ImageNet-LT.

Long-tailed CIFAR We conduct extensive experiments on long-tailed CIFAR datasets with three different imbalanced ratios: 10, 50 and 100. Table 1 reports the accuracy of various methods. For CIFAR-10 series, our method achieves comparable or better results comparing other complicated methods. When working on CIFAR-100 series, our method achieves best results across all imbalance ratios, compared with two-stage fine-tuning strategies (*i.e.*, CE-DRW/CE-DRS) and previous state-of-the-arts (*i.e.*, Decouple and BBN). Especially for long-tailed CIFAR-100 with imbalanced ratio 100 (the most extreme imbalance case), we get **44.7%** accuracy which is **2.1%** higher than previous BBN.

iNaturalist 2018 We further evaluate our methods on the iNaturalist 2018 dataset. Similar to Decouple (Kang et al., 2020) and BBN (Zhou et al., 2020), we present results training after 90 and 200 epochs for fair comparison. As illustrated in the Table 2, with an end-to-end trained plain ResNet-50 model, we surpass other complicated methods including two-stage fine-tune (Decouple) and well-designed architecture (BBN). When S =1, where total training epochs are 10 epochs less than Decouple, we get **1.5%** gains compared with the totally decouple training strategy cRT. We can achieve further improvements with the same training epochs as Decouple (see $S = 10$ for 90 epochs and $S = 40$ for 200 epochs).

ImageNet-LT Table 3 presents results for the most challenging ImageNet-LT. The results of BBN are conducted using the author's open-sourced codebase. From the table we see that our simple method with plain ResNet50, with less training epochs (see $S = 1$), outperform the current state-of-the-art about **0.6%** higher than Decouple and **2.0%** higher than BBN. With the same training epochs as them (see $S = 10$), our method gets further improvements, about **0.9%** higher than Decouple and **2.3%** higher than BBN.

Fine-grained analysis To better validate our assumption that memorizing low-regularity samples with small learning rate can avoid seriously damage the representation of high-regularity samples, we further report accuracy on three splits of the set of classes: *Many-shot* (more than 100 images), *Medium-shot* (20–100 images) and *Few-shot* (less than 20 images). As shown in Table 4, standard training process (see Cross Entropy with IB only) always perform best on *Many-shot* since head class samples dominate the training batch all the time. Meanwhile, our method can improve the performance of tail classes by a large margin due to the CR sampling in the last training stage. It is worth to note that while greatly boosting the recognition of tail classes, our switching method only slightly damage the performance of head classes (compared with Cross Entropy with CR only), indicating that memorizing tail class samples with small learning rate can better handle the trade-off between high-regularity head classes and low-regularity tail classes.

4.5 Ablation studies

4.5.1 Analysis on hyper-parameter S

To find the optimal setting of S, which is the hyper-parameter controlling when to switch, we investigate S and corresponding results are shown in Table 5. Interestingly, our method achieves comparable results despite different values of S, indicating S is not dataset/distribution dependent or sensitive. This is consistent with our motivation: memorization of tail classes will not seriously disrupt the learned representation with smaller learning rate. Thus, once there is CR during the small learning rate stage, model could jointly fine-tune both feature extractor and classifier to achieve better generalization, regardless of the specific value of S.

When S turns to infinity, the only difference between our method and regular SGD training is that we still need a switching action. To simulate this situation, we enlarge S to 200 and 500. When S=200 the classification performances at an imbalance ratio of 50 on Long-tailed CIFAR-10 and CIFAR-100 are 82.2 and 48.1, respectively. Even when S=500, our method still achieves 82.0 and 47.5. Considering that the regular SGD only achieves 77.9 and 44.9, as also illustrated in Table 6, the superiority of the switching from IB to CR at small learning rate is shown inevitably.

4.5.2 Combinations of sampling strategies

In order to find the optimal sampler combination before and after switching, we conduct comprehensive experiments on long-tailed CIFAR-10 (imbalance ratio: 50) with combinations of different data samplers used in different stages. As shown in Table 6, our strategy, which switches instance-balanced sampling to class-reversed sampling in the small learning rate stage, achieves the best performance across all experimental settings. We draw the same conclusion with Decouple that *instance-balanced sampling gives the most generalizable representations*, for using instance-balanced in the first stage always performs better than other results. In addition, switching to class-reversed sampling can always bring a significant improvement no matter what samplers used in the first stage, except class-reversed sampling on the long-tailed CIFAR-100 with imbalance ratio 100 and 50 (see the last row in Table 6). We conjecture this is because class-reversed sampling cannot learn the general representations on such extreme imbalanced data, since it mainly samples from the tail classes with low cardinality. Without generalizable representation and seeing samples from other classes, network cannot generalize well across all classes.

We also investigate the progressively *Switching* in Table 7. For the first CB then CR setting, results are almost the same as only CR, showing *Switching* is robust to the samplers used in earlier stages. However, first CR then CB will lead a great drop in accuracy, indicating memorizing low-regularity tail classes should happen in the last training stage with high-regularity domain knowledge first-disrespect of it will hurt the performance.

4.5.3 Comparing with decoupling paradigm

To further compare our method with Decouple, we investigate the factors of fixing feature extractor and re-training classifier towards learning long-tailed distributions, which are adopted in Decouple. From the results shown in Table 8, the following observations can be made:

- *Joint training is better.* Training the backbone and the classifier jointly always performs better than fixing the backbone. This phenomenon indicates that although instance-balanced sampling gives the most generalizable representations, it is not good enough. Fine-tuning the backbone with low-regularity tail class samples in the small learning rate stage can significantly improve its representation ability across tail classes.

Table 1 Top-1 accuracy of ResNet-32 on long-tailed CIFAR-10 and CIFAR-100

Dataset	Long-tailed CIFAR-10			Long-tailed CIFAR-100		
Imbalance ratio	100	50	10	100	50	10
Cross Entropy†	70.4	74.8	86.7	38.3	43.9	55.7
Cross Entropy*	73.1	77.9	86.4	40.7	44.9	57.2
Focal† (Lin et al., 2017)	70.4	76.7	86.7	38.4	44.3	55.8
CE-DRW† (Cao et al., 2019)	76.3	80.0	87.6	41.5	45.3	58.1
CE-DRS† (Cao et al., 2019)	75.6	79.8	87.4	41.6	45.5	58.1
CB-Focal† (Cui et al., 2019)	74.6	79.3	87.1	39.6	45.2	58.0
LDAM-DRW† (Cao et al., 2019)	77.0	81.0	88.2	42.0	46.6	58.7
Decouple-cRT* (Kang et al., 2020)	73.8	80.7	86.7	40.1	46.4	57.7
Decouple-LWS* (Kang et al., 2020)	73.5	77.5	86.1	40.2	45.7	58.1
BBN† (Zhou et al., 2020)	**79.8**	82.2	88.3	42.6	47.0	59.1
Ours ($S = 1$)	79.7	**82.9**	**88.4**	**44.7**	**49.5**	**59.5**

Bold values indicate the best performance, i.e., (Top-1 accuracy, %)

Rows with † denote results directly copied from BBN (Zhou et al., 2020)

*denotes results reproduced with the authors' code

Table 2 Top-1 accuracy of ResNet-50 on iNaturalist 2018

Dataset	iNaturalist 2018
Cross Entropy†	57.2
CE-DRW†	63.7
CE-DRS†	63.6
CB-Focal†	61.1
LDAM-DRW†	68.0
Decouple-NCM†	58.2 / 63.1
Decouple-cRT†	65.2 / 67.6
Decouple-τ-normed †	65.6 / 69.3
Decouple-LWS†	65.9 / 69.5
BBN†	66.3 / 69.6
Ours ($S = 1$ / $S = 1$)	**66.7/70.4**
Ours ($S = 10$ / $S = 40$)	**66.8/70.0**

Bold values indicate the best performance, i.e., (Top-1 accuracy, %)

Rows with † denote results directly copied from their original paper. We present results when training for 90 / 200 epochs for fair comparison

- *Re-training is matter when no switching.* When training with the switching strategy, results with re-training or without re-training the classifier are much similar (see rows with CB, CR as the switching sampler). However, interestingly, re-training the classifier can bring improvements in the standard training procedure (see rows with IB as the switching sampler). We speculate that model trained by uniform instance-balanced sampling would have a strong bias towards tail classes in both backbone and classifier. Re-training classifier based on the learned general representations can alleviate it.
- *Switching and joint training are complementary.* We compare the results of only switching to only joint training, finding that while switching samplers and joint training can bring improvements respectively, their combination can improve the performance

Table 3 Top-1 accuracy of ResNet-50 on large-scale long-tailed datasets ImageNet-LT

Dataset	ImageNet-LT
Cross Entropy†	41.6
Decouple-NCM†	44.3
Decouple-cRT†	47.3
Decouple-τ-normed †	46.7
Decouple-LWS†	47.7
BBN*	45.9
Ours ($S = 1$)	**47.9**
Ours ($S = 10$)	**48.2**

Rows with † denote results directly copied from Decouple (Kang et al., 2020)

*denotes results reproduced with the authors' code

Table 4 Fine-grained results on the most skewed long-tailed CIFAR-100 (imbalance ratio: 100) and the most challenging ImageNet-LT, compared with the previous state-of-art

Dataset	Long-tailed CIFAR-100				ImageNet-LT			
	Many	Medium	Few	All	Many	Medium	Few	All
Cross Entropy (IB only)	**68.2**	39.7	9.9	40.7	**64.0**	33.8	5.8	41.6
Cross Entropy (CR only)	39.8	32.8	12.9	32.1	31.6	32.0	10.3	28.9
Decouple-cRT	58.1	40.3	18.0	40.1	58.8	44.0	26.1	47.3
Decouple-LWS	59.5	40.7	17.4	40.2	57.1	45.2	29.3	47.7
BBN†	54.5	**51.0**	16.7	42.6	56.2	**46.6**	14.1	45.9
Ours ($S = 1$)	57.1	48.1	**26.5**	**44.7**	53.5	45.2	**41.8**	**47.9**

Our methods can boost the performance of tail classes while sightly damaging the performance of head classes

further. Fine-tuning with class-balanced or class-reversed distributions can boost the generalization ability further.

Further, we valid the quality of features learned by standard training procedure and our switching training procedure in Table 10, just like Decouple. Although a slightly lower with IB, re-training based on our feature can bring significant improvements compare with standard features. These results also indicate a disadvantage of Decouple: performance of re-training classifier depends on the performance of feature extractor. Once the feature representation is sub-optimal, the re-trained classifier is sub-optimal.

To validate our method could reach a better balance under bias-variance trade-off, we calculate the total error of each method in Table 9. Our method (when S=1) yields a lower upper bound on the generalization error, and therefore higher test accuracy, lower Bias, and lower Variance, which indicate our switching algorithm performs better in the challenging trade-off compared with other methods.

Table 5 Determining of the optimal S on long-tailed CIFAR-10 (imbalance ratio: 50) and CIFAR-100 (imbalance ratio: 50)

S	CIFAR-10	CIFAR-100
0	82.6	**49.7**
1	**82.9**	**49.5**
5	**82.9**	49.3
10	82.8	49.1
50	82.6	49.1
200	82.2	48.1
500	82.0	47.5
CE (IB only)	77.9	44.9

Results indicate that S is not dataset/distribution dependent or sensitive

Table 6 Comprehensive results on long-tailed CIFAR-10 (imbalance ratio: 50) with combinations of different data samplers used in different stages

Sampling strategy combination	Long-tailed CIFAR-10			Long-tailed CIFAR-100		
Imbalance ratio	100	50	10	100	50	10
IB $\Longrightarrow$ IB	73.1	77.9	86.4	40.7	44.9	57.2
IB $\Longrightarrow$ CB	77.1	81.9	87.9	44.2	48.7	59.2
IB $\Longrightarrow$ CR	**79.7**	**82.9**	**88.4**	**44.7**	**49.5**	**59.5**
CB $\Longrightarrow$ IB	66.5	73.8	86.7	33.3	37.1	55.0
CB $\Longrightarrow$ CB	73.0	78.5	87.3	36.2	40.3	56.9
CB $\Longrightarrow$ CR	74.8	80.7	87.9	38.6	42.4	57.8
CR $\Longrightarrow$ IB	63.8	74.8	85.2	24.7	28.7	51.3
CR $\Longrightarrow$ CB	64.2	72.7	86.3	24.7	29.2	52.5
CR $\Longrightarrow$ CR	68.4	76.3	86.8	22.9	28.0	53.2

Table 7 Determining of the way of switching strategies on long-tailed CIFAR-10 (imbalance ratio: 50)

First sampler	Switching sampler	Accuracy
IB	CR	**82.9**
	CB for 5 epochs, then CR	82.7
	CR for 5 epochs, then CB	79.2

4.6 Validation and visualization of our proposals

4.6.1 Learning speed

In order to further validate our method could learn long-tailed distributions more efficiently, we plot the test accuracy per epoch of three methods with different sampling strategies in Fig. 3. Compared with using IB only, switching to CR can immediately improve the performance by a large margin. Meanwhile, although BBN could achieve comparable performance with ours, it converges more slowly since it optimizes two branches of feature extractor in turn during training.

4.6.2 Learning rate scheduling

Intuitively, a training example from head classes should be learned quickly since it is consistent with many others and the gradient steps for all consistent examples should be well aligned. As Jiang et al. (2020) indicates that strong regularities in a data set are not only better learned at asymptote leading to better generalization performance but are also learned sooner in the time course of training, we conjecture that head class samples will be learned sooner than tail class samples and plot average proportion correct as a function of training epoch for each class to validate it.

Figure 4 shows the learning speed of 4 selected classes with SGD using stage-wise constant learning rate scheduling. In Fig. 5 we show the learning speeds of 4 selected classes trained with SGD using constant learning rate scheduling with the standard training procedure. The 4 panels show the results of different values of constant learning rate used in training. It is observed that faster convergence is achieved with smaller learning rate (see

0.1, 0.02 and 0.01). While the learning rate is so small, *e.g.*, 0.001, the learning speed of each class is significantly slowed down.

In Fig. 6 we show the learning speeds of our switching training procedure trained with SGD using constant learning rate scheduling. Similar to Fig. 5, proper small learning rate could accelerate the convergence, with higher and more stable accuracy. It is worth to note that switching to class-reversed sampler always improves the accuracy of tail classes, but will damage the representative ability of head classes to some extent. Stage-wise constant learning brings the smallest damage to the head class representations, showing the necessity of building generalization representations first. Quantitative results of both standard training and switching training are shown in Table 11.

Here we manage to explain why class-reversed sampler is effective. The reason that switching to class-reversed sampler performs well is that it delayed the learning of low-regularity samples (tail classes samples) to later small learning rate stages. In the first stage, when almost head class samples are learned, the patterns and structures discovered in those high-regularity samples can be used to build a generalizable representation. In later stage, network is able to learn or memorize low-regularity samples of tail classes based on the representations from a clean subset of high-regularity samples. In addition, learning or memorizing tail class samples will not seriously disrupt the learned representation as the learning rate is much smaller than the earlier stages. In contrast, standard learning procedure without switching could not focus on the tail class samples since the extreme data imbalance, leading under-representation for tail classes, while SGD with (small) constant learning rate learns the examples across all classes quickly, which cannot learn the generalizable representation from high-regularity samples of head classes before.

Table 8 Comparisons between Decouple learning paradigm and our learning paradigm on long-tailed CIFAR-10 (imbalance ratio: 50), where Decouple indicates fixing the backbone and re-train the classifier from scratch while we continue to joint train both

First sampler	*S*	Switching sampler	Joint training	Re-training classifier	Test accuracy
IB	1	IB			76.3
			✓		77.9
				✓	77.7
			✓	✓	78.9
		CB			81.9
			✓		81.9
				✓	81.8
			✓	✓	81.9
		CR			81.4
			✓		**82.9**
				✓	81.6
			✓	✓	82.4

Table 9 Total error (bias^2+variance) of different methods on the test set of long-tailed CIFAR-10 (imbalance ratio: 50)

Method	Test accuracy ↑	Bias^2 ↓	Variance ↓	Total Error ↓
Cross Entropy (IB only)	0.779	0.049	0.168	0.217
Cross Entropy (CR only)	0.763	0.056	0.183	0.239
Decouple-cRT	0.807	0.037	0.148	0.185
BBN	0.822	0.032	0.146	0.178
Ours ($S = 1$)	**0.829**	**0.029**	**0.142**	**0.171**

Table 10 Feature quality of Decouple learning paradigm and our switching learning paradigm on long-tailed CIFAR-10 (imbalance ratio: 50)

Feature	Re-training	Test accuracy
Standard	IB	77.7
	CB	80.7
	CR	82.2
Our	IB	77.0
	CB	81.4
	CR	**82.6**

We firstly train the model with standard and switching procedure respectively, then re-train the classifier with different data samplers with backbone fixed

5 Conclusion

In this paper, we investigate long-tailed visual recognition from a memorization-generalization point of view, which not only theoretically explains the previous methods, but also provides a simple yet effective *Switching* strategy to memorize tail classes without huge damage to the head classes. The detailed implementation only contains switching instance-balanced sampling to class-reversed sampling during the last few training epochs, which is clean and elegant. Closely afterwards, we give the generalization error upper bound of different sampling strategies. Further empirical findings show the inevitability to deal the trade-off between head class representing and tail class memorizing in the memorization stage with small learning rate.

Appendix A: Regularity under long-tailed distributions

A.1 Unified regularity measures

To investigate the memorization-generalization continuum in deep learning towards long-tailed distributions, we introduce a pair of sample regularity measures for both training and testing samples with a formulation-consistent representation according to Zhang et al. (2021):

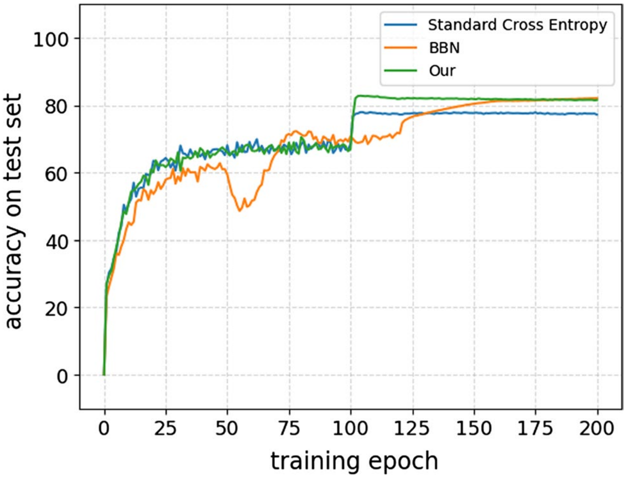


Fig. 3 Test performance of three methods with different sampling strategies on long-tailed CIFAR-10 (imbalance ratio: 50) with SGD using stage-wise constant learning rate

- For training samples, the sample regularity is measured with the combination of forgetting events (Toneva et al., 2019) and cumulative learned events (also named CBTL (Jiang et al., 2020)) as follows:

  Forgetting events Let $for_i^t = \mathbf{1}_{acc_i^{t-1}=1, acc_i^t=0}$, the forgetting events of one sample $\{x_i, y_i\}$ at epoch t are defined as follows:

$$\mathbb{F}_i^t = \sum_{n=1}^{t} for_i^n. \tag{8}$$

  Cumulative learned events For sample $\{x_i, y_i\}$, $\hat{y}_i^t = \arg\max g(y_i|x_i;\theta^t)$ is the predicted label for sample x_i obtained after t epochs of SGD optimization. Let $acc_i^t = \mathbf{1}_{\hat{y}_i^t = y_i}$ be a binary variable indicating whether the sample is correctly classified at time epoch t, the cumulative learned events events at epoch t are defined as follows:

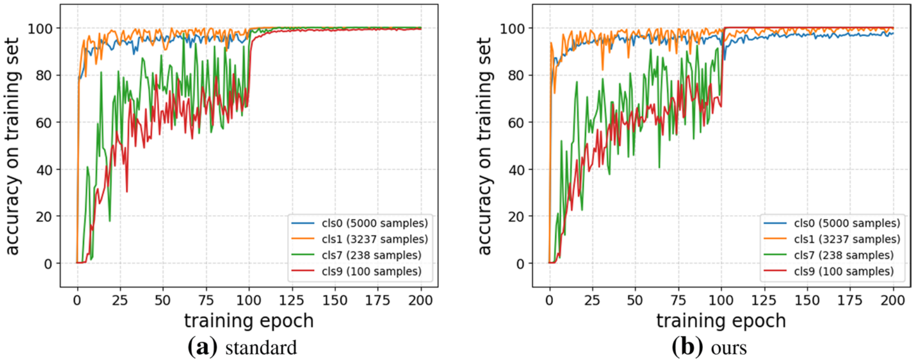
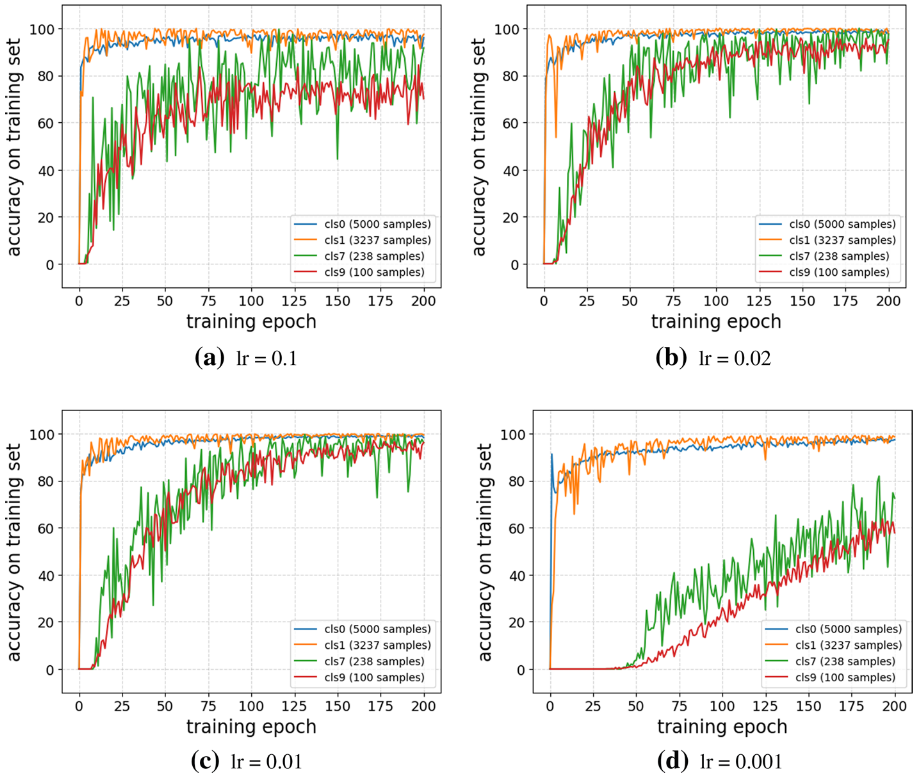


Fig. 4 Learning speed of examples of 4 selected class with SGD using stage-wise constant learning rate. Left: standard training procedure. Right: our switching training procedure



(a) lr = 0.1

(b) lr = 0.02

(c) lr = 0.01

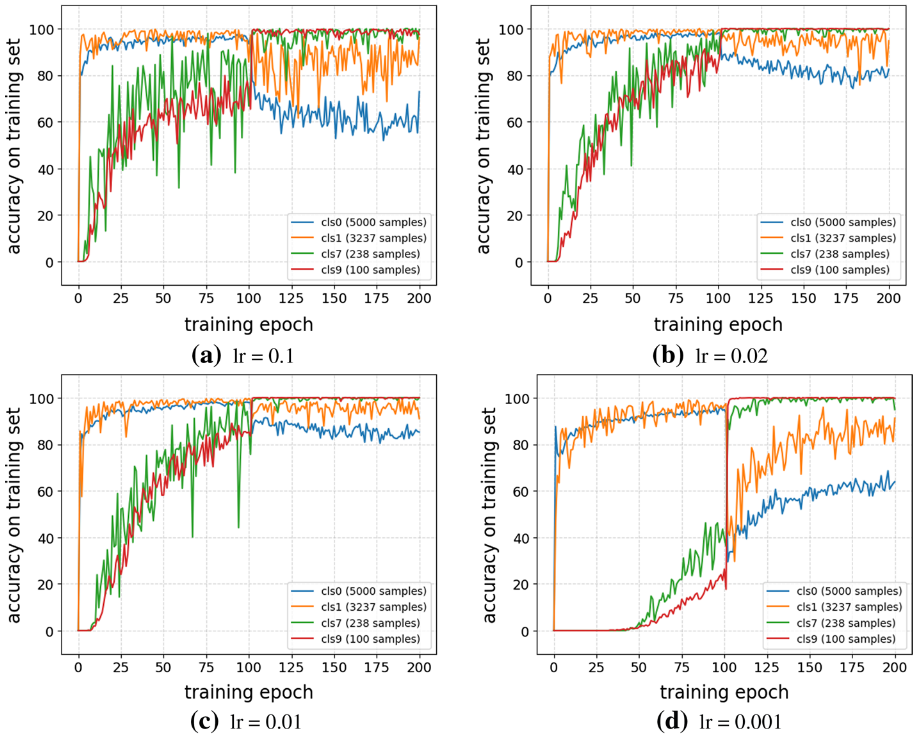
(d) lr = 0.001

Fig. 5 Learning speed of examples of 4 selected classes with SGD using constant learning rate with standard training strategy. The 4 different learning rates correspond to the constants used in the stage-wise scheduler

$$\mathbb{L}_i^t = \sum_{n=1}^{t} acc_i^n. \tag{9}$$

- For testing samples, the sample regularity is measured with the combination of mal-generalizing events and cumulative generalized events (Zhang et al., 2021). The definition of the mal-generalizing is nothing but substitute training samples with testing samples in Eq. 8. Similarly, cumulative generalized events could be calculated through Eq. 9 for certain testing sample.

It is worth to note that mal-generalizing events and cumulative generalized events are recorded after each epoch ends, since testing samples do not participate the training process. Therefore, for the consistency between both regularity measures (e.g., the visual comparability between Figs. 7, 8 and 9, 10), the forgetting events are recorded after each epoch ends. This may not harm the correctness of the empirical studies since experimental evidence shows a strong statistical correlation between Toneva's and ours, i.e. the Pearson correlation coefficients of their results are as high as 0.9835 (Zhang et al., 2021).



(a) lr = 0.1

(b) lr = 0.02

(c) lr = 0.01

(d) lr = 0.001

Fig. 6 Learning speed of examples of 4 selected classes with SGD using constant learning rate with our training strategy. The 4 different learning rates correspond to the constants used in the stage-wise scheduler

Table 11 Test performance of models trained with various learning rate schedulers on long-tailed CIFAR-10 (imbalance ratio: 50)

Standard			Our		
Optimizer	Learning rate	Test accuracy	Optimizer	Learning rate	Test accuracy
SGD	Stage-wise	**77.9**	SGD	Stage-wise	**82.9**
SGD	0.1	75.5	SGD	0.1	77.7
SGD	0.02	76.1	SGD	0.02	78.3
SGD	0.01	75.0	SGD	0.01	77.7
SGD	0.001	65.9	SGD	0.001	66.1

Based on these notations, we give the metric to describe the regularity of one sample, where with higher cumulative learned events as well as lower forgetting events, the higher regularity it will be and vice versa.

A.2 Regularity analysis

For Long-tailed CIFAR-10 with imbalance ratio 50, we run the ResNet-32 for 10 times and plot the averaged cumulative learned events and forgetting events of each sample grouped by its class, as shown in Fig. 7. We surprisingly find that the clustering degree of samples is almost proportional to its cardinality. To explore this phenomenon further, we plot the same events of same samples when learning the standard CIFAR-10 with class-balanced distributions in Fig. 8. Compared with the same samples under class-balanced distributions, long-tailed distribution samples show different properties of each class: higher degree of clustering of head classes (cls0, cls1) and lower degree of clustering of tail classes (cls6, cls7, cls8, cls9). Differences between them indicate that the cardinality of one class can significantly affect the regularity itself during training: *regularity of the same training samples will be sharply decreased with the reduction of class cardinality*, which is easy to understand: the more samples one class have, the higher regular it will be. The comparison between Figs. 9 and 10 further shows that such skewed regularity of the training samples may cause the network generalization degraded, i.e., the less training samples one class have, the lower regularity the validation samples of the same class will own.

In order to analysis this phenomenon, we propose a novel metric to quantize the regularity of each class. For class j containing n_j samples, regularity event of each sample $\{x_i, y_i\}$ can be denoted by its cumulative learned events and forgetting events as $r_{i,j} = \{\mathbb{L}_i^T, \mathbb{F}_i^T\}$, which is a point on the two-dimensional plane. There are three sub-procedures to calculate the regularity of each class j:

1. Let $\{(\mathbb{L}_i^T, \mathbb{F}_i^T) | 1 \leq i \leq n_j\} = [\boldsymbol{LF}]$ denote the regularity set of class j, we calculate the covariance matrix as follows:

$$\boldsymbol{C}_j = \begin{bmatrix} \mathbb{E}[(\boldsymbol{L} - \mathbb{E}[\boldsymbol{L}])(\boldsymbol{L} - \mathbb{E}[\boldsymbol{L}])] & \mathbb{E}[(\boldsymbol{L} - \mathbb{E}[\boldsymbol{L}])(\boldsymbol{F} - \mathbb{E}[\boldsymbol{F}])] \\ \mathbb{E}[(\boldsymbol{F}^T - \mathbb{E}[\boldsymbol{F}])(\boldsymbol{L} - \mathbb{E}[\boldsymbol{L}])] & \mathbb{E}[(\boldsymbol{F} - \mathbb{E}[\boldsymbol{F}])(\boldsymbol{F} - \mathbb{E}[\boldsymbol{F}])] \end{bmatrix} \tag{10}$$

   where $\mathbb{E}$ is the expectation.
2. calculate the F-norm of $\boldsymbol{C}_j$:

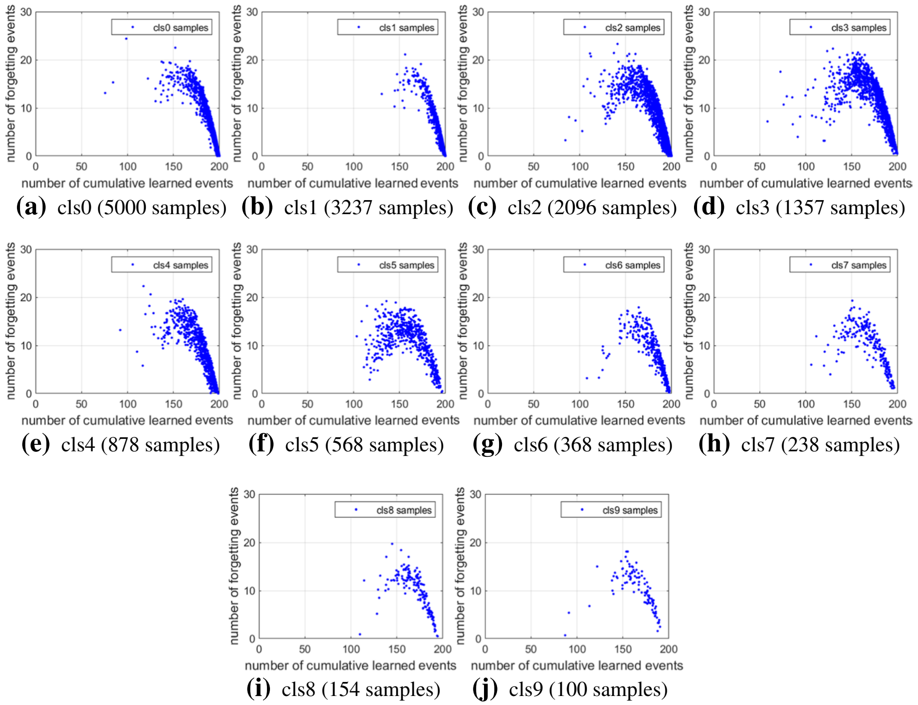
$$||\boldsymbol{C}_j||_F = \sqrt{\sum_{m=1}^{2} \sum_{n=1}^{2} |c_{mn}|^2}. \tag{11}$$

3. normalize the F-norm by its cardinality:

$$I_j = ||\boldsymbol{C}_j||_F / n_j. \tag{12}$$

   This metric essentially indicates the deviation of each class, so we name it *Irregularity*.

As shown in Table 12, the *Irregularity* is almost proportionally to the reciprocal of its cardinality, which is consistent with our visual perception. To further validate the correlation between regularity and its cardinality, we exploit the Pearson correlation coefficient. Let $\boldsymbol{I} = \{I_i | 1 \leq i \leq C\}$ be the regularity set of all classes and $\boldsymbol{N} = \{n_i | 1 \leq i \leq C\}$ be the cardinality of all classes, we calculate the Pearson coefficient as follows:



(a) cls0 (5000 samples) **(b)** cls1 (3237 samples) **(c)** cls2 (2096 samples) **(d)** cls3 (1357 samples)

(e) cls4 (878 samples) **(f)** cls5 (568 samples) **(g)** cls6 (368 samples) **(h)** cls7 (238 samples)

(i) cls8 (154 samples) **(j)** cls9 (100 samples)

Fig. 7 Cumulative learned events and forgetting events of each sample of Long-tailed CIFAR-10 with imbalance ratio 50. The regularity of one class will be higher with more samples gathering in the lower right corner of the picture

$$P = \frac{\sum \boldsymbol{NI} - \frac{\sum \boldsymbol{N} \sum \boldsymbol{I}}{C}}{\sqrt{\left(\sum \boldsymbol{N}^2 - \frac{(\sum \boldsymbol{N})^2}{C}\right)\left(\sum \boldsymbol{I}^2 - \frac{(\sum \boldsymbol{I})^2}{C}\right)}}. \tag{13}$$

the Pearson coefficient is **−0.6112**, indicating cardinality and regularity are significantly negatively correlated.

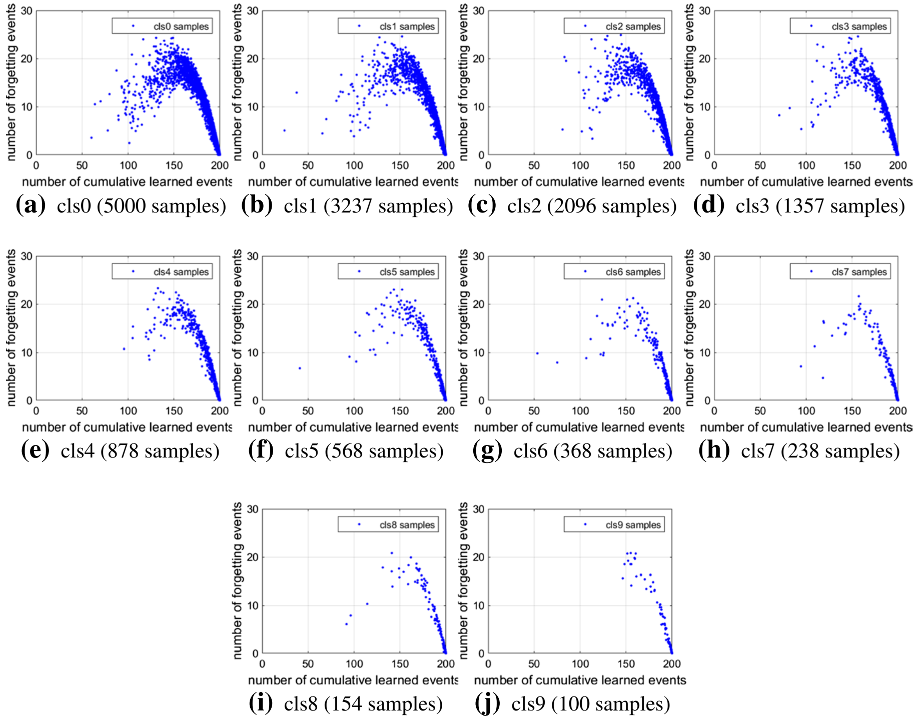
Appendix B: Proofs and derivations in Sect. 3.1

B.1 Proof of the upper bound

Proof Given a long-tailed dataset $\mathcal{D}$ sampled from the main dataset $\mathcal{S}$, we define:

$$\Phi(\mathcal{D}) = \sup_{\theta \in \Theta} \mathbb{E}_{(x,y)}[\ell(f_\theta(x), y)] - L(\theta;\mathcal{D}). \tag{14}$$

To apply McDiarmid's inequality (Rastogi, 2011) to provide the upper bound on $\Phi(\mathcal{D})$, we first show that $\Phi(\mathcal{D})$ satisfies the remaining condition of McDiarmid's inequality. Let $\mathcal{D}$



(a) cls0 (5000 samples) **(b)** cls1 (3237 samples) **(c)** cls2 (2096 samples) **(d)** cls3 (1357 samples)

(e) cls4 (878 samples) **(f)** cls5 (568 samples) **(g)** cls6 (368 samples) **(h)** cls7 (238 samples)

(i) cls8 (154 samples) **(j)** cls9 (100 samples)

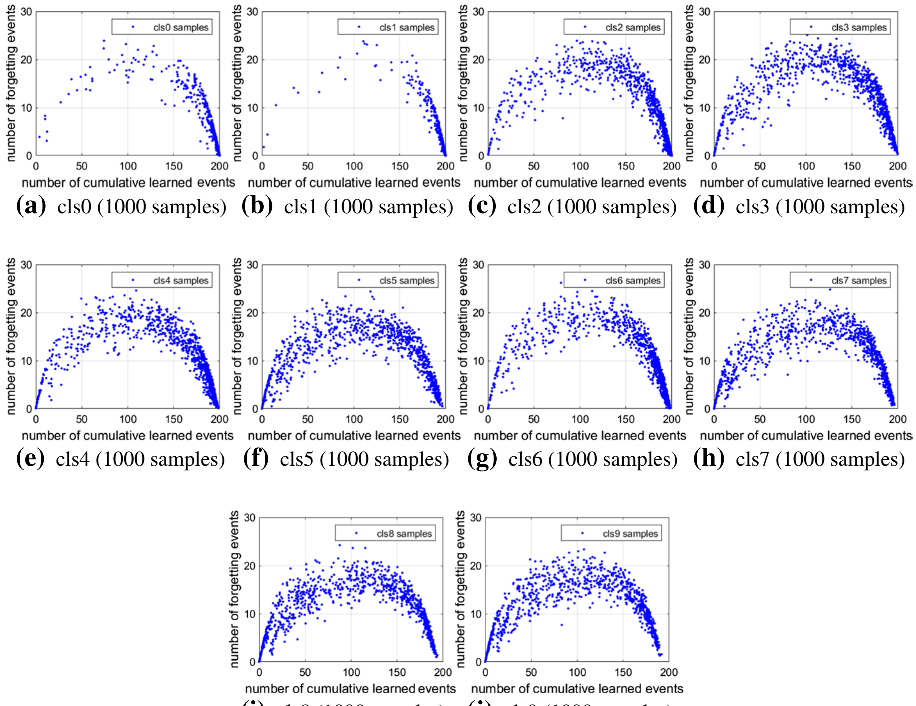
Fig. 8 Cumulative learned events and forgetting events of each sample of standard CIFAR-10. Samples here are in one-to-one correspondence with samples in Fig. 7

and $\mathcal{D}'$ be two datasets differing by exactly one point of an arbitrary index i_0, *i.e.*, $\mathcal{D}_i = \mathcal{D}'_i$ for all $i \neq i_0$ and $\mathcal{D}_{i_0} \neq \mathcal{D}'_{i_0}$. Then, the upper bound on $\Phi(\mathcal{D}') - \Phi(\mathcal{D})$ is given as follows:

$$\begin{aligned}\Phi(\mathcal{D}') - \Phi(\mathcal{D}) &\le \sup_{\theta\in\Theta} L(\theta;\mathcal{D}) - L(\theta;\mathcal{D}') \\ &= \sup_{\theta\in\Theta}\left(\sum_{j\in C}\sum_{i\in n_j}\frac{p_j}{n_j}L(\theta;\mathcal{D}) - \sum_{j\in C}\sum_{i\in n_j}\frac{p_j}{n_j}L(\theta;\mathcal{D}')\right) \\ &\le \sup_{\theta\in\Theta}\frac{p_j}{n_j}\left|L_{ji_0}(\theta;\mathcal{D}) - L_{ji_0}(\theta;\mathcal{D}')\right| \\ &\le \frac{p_j}{n_j}M\end{aligned} \tag{15}$$

Therefore, $\left|\Phi(\mathcal{D}) - \Phi(\mathcal{D}')\right| \le \frac{p_j}{n_j}M$ since we also have $\Phi(\mathcal{D}) - \Phi(\mathcal{D}') \le \frac{p_j}{n_j}M$. Thus, according to McDiarmid's inequality, for any $\delta > 0$, with probability at least $1 - \delta$ we have:

$$\Phi(\mathcal{D}) \le \mathbb{E}_{\mathcal{D}}[\Phi(\mathcal{D})] + \sqrt{\sum_{j\in C}\frac{p_j^2}{n_j}}\sqrt{\frac{\ln(1/\delta)}{2}}M. \tag{16}$$



(a) cls0 (1000 samples) **(b)** cls1 (1000 samples) **(c)** cls2 (1000 samples) **(d)** cls3 (1000 samples)

(e) cls4 (1000 samples) **(f)** cls5 (1000 samples) **(g)** cls6 (1000 samples) **(h)** cls7 (1000 samples)

(i) cls8 (1000 samples) **(j)** cls9 (1000 samples)

Fig. 9 Cumulative generalized events and mal-generalizing events of each validation sample of Long-tailed CIFAR-10 with imbalance ratio 50. Note that the validation set of the Long-tailed CIFAR-10 is class-balanced. The regularity of one class will be higher with more samples gathering in the lower right corner of the picture

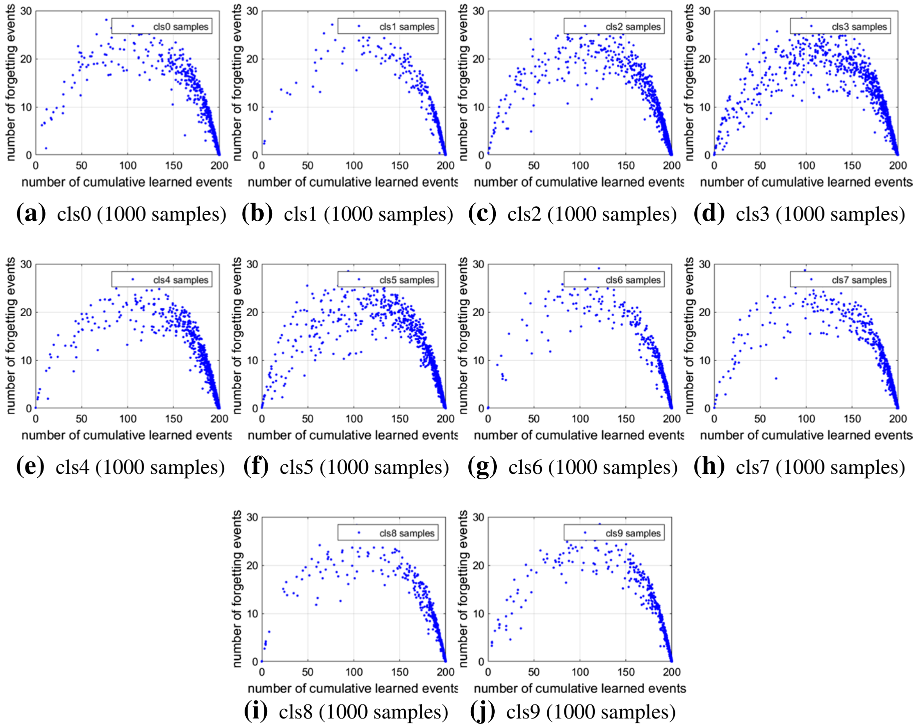
Therefore,

$$\begin{aligned}\mathbb{E}_{\mathcal{D}}[\Phi(\mathcal{D})] &= \mathbb{E}_{\mathcal{D}}\left[\sup_{\theta\in\Theta}\mathbb{E}_{(x,y)}\left[\ell\left(f_\theta(x),y\right)\right] - L^s(\theta;\mathcal{D}) + L^s(\theta;\mathcal{D}) - L(\theta;\mathcal{D})\right]\\ &\le \mathbb{E}_{\mathcal{D}}\left[\sup_{\theta\in\Theta}\mathbb{E}_{(x,y)}\left[\ell\left(f_\theta(x),y\right)\right] - L^s(\theta;\mathcal{D})\right] - \mathcal{Q}_n\\ &\le \mathbb{E}_{\xi,D,\mathcal{D}'}\left[\sup_{\theta\in\Theta}\frac{1}{n}\sum_{i=1}^{n}\xi_i\left(\ell\left(f_\theta\left(\bar{x}'_i\right),\bar{y}'_i\right) - \ell\left(f_\theta\left(\bar{x}_i\right),\bar{y}_i\right)\right)\right] - \mathcal{Q}_n\\ &\le 2\mathfrak{R}_n(\Theta) - \mathcal{Q}_n,\end{aligned} \tag{17}$$

where

$$\mathcal{Q}_n \quad = \quad \mathbb{E}_{\mathcal{D}}\left[\inf_{\theta\in\Theta}\sum_{j=1}^{C}\sum_{i=1}^{n_j}\left(\frac{p_j}{n_j} - \frac{1}{n}\right)\ell\left(f_\theta\left(x_i\right),y_i\right)\right]. \tag{18}$$

Therefore, for any $\delta > 0$, with probability at least $1-\delta$ we have:



(a) cls0 (1000 samples) **(b)** cls1 (1000 samples) **(c)** cls2 (1000 samples) **(d)** cls3 (1000 samples)

(e) cls4 (1000 samples) **(f)** cls5 (1000 samples) **(g)** cls6 (1000 samples) **(h)** cls7 (1000 samples)

(i) cls8 (1000 samples) **(j)** cls9 (1000 samples)

Fig. 10 Cumulative generalized events and mal-generalizing events of each validation sample of standard CIFAR-10. Samples here are in one-to-one correspondence with samples in Fig. 9

Table 12 Quantitative results of the regularity of each class on long-tailed CIFAR-10 (imbalance ratio: 50)

Class	Cardinality	F-norm	Irregularity (F-norm / Cardinality)	Pearson
0	5000	85.7341	0.0171	**−0.6112**
1	3237	44.3574	0.0137	
2	2096	296.3582	0.1414	
3	1357	397.6501	0.2930	
4	878	267.5524	0.3047	
5	568	447.2937	0.7875	
6	368	253.5138	0.6889	
7	238	355.5270	1.4938	
8	154	279.5059	1.8150	
9	100	363.4100	3.6341	

All indexes are calculated based on the cumulative learned events and forgetting events

$$\Phi(\mathcal{D}) \leq 2\mathfrak{R}_n(\Theta) - \mathcal{Q}_n + M\sqrt{\sum_{j\in C}\frac{p_j^2}{n_j}}\sqrt{\frac{\ln(1/\delta)}{2}}. \tag{19}$$

Substituting Eqs. 19 into 14 we have:

$$\begin{aligned}\mathbb{E}_{(x,y)}[\ell(f_\theta(x),y)] \leq L(\theta;\mathcal{D}) + 2\mathfrak{R}_n(\Theta) - \mathcal{Q}_n \\ +M\sqrt{\sum_{j\in C}\frac{p_j^2}{n_j}}\sqrt{\frac{\ln(1/\delta)}{2}}\end{aligned} \tag{20}$$

□

B.2 Proof of the Theorem 1

Proof We assume that:

1. The range of Θ is narrow due to the learning rate is small and the network has converged in the previous training process, so $\mathfrak{R}_n(\Theta) \to 0$ as $n \to \infty$, which has been shown to be satisfied for various models and sets Θ (Bartlett and Mendelson, 2002; Mohri et al., 2012; Kawaguchi et al., 2017; Bartlett et al., 2017).
2. Without loss of generality, the classes are sorted by cardinality in decreasing order, thus sampling weight p_j of each class j is ordered. p_1 is weight of the class with most samples and p_c is the weight of the class with least samples.
3. $L_i \leq L_j$ if $i < j$. This is an empirical conclusion that average loss of tail class samples are always higher than its of head class samples with a model trained by instance-balanced sampling.

Now let's compare $\mathbb{E}^s_{(x,y)}[\ell(f_\theta(x),y)]$ and $\mathbb{E}_{(x,y)}[\ell(f_\theta(x),y)]$. Since both $M\sqrt{\frac{\ln(1/\delta)}{2n}}$ and $M\sqrt{\sum_{j\in C}\frac{p_j^2}{n_j}}\sqrt{\frac{\ln(1/\delta)}{2}}$ will disappear as $n \to \infty$, the core is to discuss the $\mathcal{Q}_n$.

We first consider the situation which only exchange the sampling rate for class 1 and class c under the instance-balanced sampling ($p_1 > p_c$). Here we have:

$$\begin{aligned}\sum_{j=1}^{C}\sum_{i=1}^{n_j}\left(\frac{p_j}{n_j}-\frac{1}{n}\right) &= \sum_{i=1}^{n_1}\left(\frac{p_c}{n_1}-\frac{1}{n}\right)L_{1i} + \sum_{i=1}^{n_c}\left(\frac{p_1}{n_c}-\frac{1}{n}\right)L_{ci} \\ &= n_1\left(\frac{p_c}{n_1}-\frac{1}{n}\right)\overline{L_1} + n_c\left(\frac{p_1}{n_c}-\frac{1}{n}\right)\overline{L_c} \\ &= (p_c-p_1)(\overline{L_1}-\overline{L_c}) > 0\end{aligned} \tag{21}$$

Therefore, for high probability we can hold that $\mathcal{Q}_n > 0$ if we only exchange the sampling weight of the class 1 and class c. Naturally, $\mathcal{Q}_n > 0$ will always hold if we exchange the sampling weight of class i and class j ($i < j$), which is exactly how class-reversed sampling works.

Now let's promote our conclusion to more general situations, what will happen if we just change the sampling weight of one class instead of exchanging? Let's increase the p_c

from p_c to p'_c, then every p_j will change to $p'_j = p_j \frac{1-p'_c}{1-p_c}$ due to the constraint $\sum_{j=1}^{C} p_j = 1$, now we have:

$$\begin{aligned}
\sum_{j=1}^{C}\sum_{i=1}^{n_j}\left(\frac{p_j}{n_j}-\frac{1}{n}\right) &= \left(\frac{p'_c}{n_c}-\frac{1}{n}\right)n_c\overline{L_c} + \sum_{j=1}^{C-1}\left(\frac{p_j}{n_j}\frac{1-p'_c}{1-p_c}-\frac{1}{n}\right)n_j\overline{L_j} \\
&= \left(\frac{p'_c}{n_c}-\frac{1}{n}\right)n_c\overline{L_c} - \sum_{j=1}^{C-1}\frac{p'_c-p_c}{1-p_c}p_j\overline{L_j} \\
&\geq (p'_c-p_c)\overline{L_c} - \sum_{j=1}^{C-1}\frac{p'_c-p_c}{1-p_c}p_j\overline{L_{c-1}} \\
&\geq (p'_c-p_c)\overline{L_c} - \frac{p'_c-p_c}{1-p_c}\overline{L_{c-1}}\sum_{j=1}^{C-1}p_j \\
&\geq (p'_c-p_c)\overline{L_c} - \frac{p'_c-p_c}{1-p_c}\overline{L_{c-1}}(1-p_c) \\
&\geq (p'_c-p_c)(\overline{L_c}-\overline{L_{c-1}}) \geq 0
\end{aligned} \tag{22}$$

□

Therefore, for high probability we can hold that $Q_n > 0$ if we increase the sampling weight of the last class, and we can extend it to any tail class similarly.

To sum up, we can draw the conclusion that $Q_n > 0$ holds if the sampling weight of tail classes is increased. Thus, with $n \to \infty$ and M is bounded, the upper bound on the expected error of class-reversed sampling is strictly lower than that for instance-balanced sampling if $Q_n + L^s - L > 0$ or if $L^s - L > 0$. Based on the empirical experience that L^s and L will always be very close after training (no matter using only IB or only CR, the final training loss will always be small), $L^s - L \to 0$ holds after the complete training, thus $Q_n + L^s - L > 0 \iff Q_n > 0$ holds.

Author contributions Chi Zhang conceived the conception of the study; Chi Zhang and Benyi Hu jointly designed the experiments and analyzed experimental results (each contributing 35% of the whole research); Benyi Hu performed the experiments; Yuhang Liuzhang contributes significantly to the theoretical analyses and proofs(contributing 15% of the whole research); Li Liu helped improve the writing quality of the original manuscript; Le Wang, Li Liu and Yuehu Liu helped perform the analysis with constructive discussions (each contributing 5% of the whole research); Chi Zhang and Benyi Hu prepared the manuscript, and all authors provided feedback during the manuscript revisions.

Funding This work was supported by the National Key Research and Development Project of New Generation Artificial Intelligence of China under Grant 2018AAA0102504.

Availability of data and materials Not available.

Declarations

Conflict of interest The authors declare that they have no conflict of interest/competing interests.

Consent to participate Not applicable.

Consent to publication Not applicable.

Ethics approval Not applicable.

References

Baloch, B. K., Kumar, S., Haresh, S., Rehman, A., & Syed, T. (2019). Focused anchors loss: cost-sensitive learning of discriminative features for imbalanced classification. In *Proceedings of The 11th Asian conference on machine learning, ACML*, PMLR, vol. 101, pp. 822–835.

Bartlett, P. L., Foster, D. J., & Telgarsky, M. (2017). Spectrally-normalized margin bounds for neural networks. In *Advances in neural information processing systems, NIPS*, pp. 6240–6249.

Bartlett, P. L., & Mendelson, S. (2002). Rademacher and gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research, 3,* 463–482.

Bej, S., Davtyan, N., Wolfien, M., Nassar, M., & Wolkenhauer, O. (2021). Loras: An oversampling approach for imbalanced datasets. *Machine Learning, 110*(2), 279–301.

Bellinger, C., Drummond, C., & Japkowicz, N. (2018). Manifold-based synthetic oversampling with manifold conformance estimation. *Machine Learning, 107*(3), 605–637.

Buda, M., Maki, A., & Mazurowski, M. A. (2018). A systematic study of the class imbalance problem in convolutional neural networks. *Neural Networks, 106,* 249–259.

Cao, K., Wei, C., Gaidon, A., Aréchiga, N., & Ma, T. (2019). Learning imbalanced datasets with label-distribution-aware margin loss. In *Advances in neural information processing systems 32: Annual conference on neural information processing systems (NeurIPS)*, pp. 1565–1576.

Cui, Y., Jia, M., Lin, T., Song, Y., & Belongie, S. J. (2019). Class-balanced loss based on effective number of samples. In *IEEE conference on computer vision and pattern recognition, CVPR*, pp. 9268–9277.

Dong, Q., Gong, S., & Zhu, X. (2017). Class rectification hard mining for imbalanced deep learning. In *IEEE international conference on computer vision, ICCV*, pp. 1869–1878.

Drummond, C., Holte, R. C., et al. (2003). C4. 5, class imbalance, and cost sensitivity: Why under-sampling beats over-sampling. In *Workshop on learning from imbalanced datasets II, Citeseer*, *11*, 1–8.

Feldman, V. (2020). Does learning require memorization? A short tale about a long tail. In Makarychev, K., Makarychev, Y., Tulsiani, M., Kamath, G., Chuzhoy, J. (Eds.) *Proccedings of the 52nd annual ACM SIGACT symposium on theory of computing, STOC*, pp. 954–959.

Feldman, V., & Zhang, C. (2020). What neural networks memorize and why: Discovering the long tail via influence estimation. CoRR abs/2008.03703.

Goyal, P., Dollár, P., Girshick, R. B., Noordhuis, P., Wesolowski, L., Kyrola, A., Tulloch, A., Jia, Y., & He, K. (2017). Accurate, large minibatch SGD: training imagenet in 1 hour. CoRR abs/1706.02677

Han, H., Wang, W., & Mao, B. (2005). Borderline-smote: A new over-sampling method in imbalanced data sets learning. In *Advances in intelligent computing, international conference on intelligent computing, ICIC proceedings, part I, lecture notes in computer science*, *3644*, 878–887.

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

He, H., & Garcia, E. A. (2009). Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering, 21*(9), 1263–1284.

Huang, C., Li, Y., Loy, C.C., & Tang, X. (2016). Learning deep representation for imbalanced classification. In *2016 IEEE conference on computer vision and pattern recognition, CVPR*, pp. 5375–5384.

Jamal, M. A., Brown, M., Yang, M., Wang, L., & Gong, B. (2020). Rethinking class-balanced methods for long-tailed visual recognition from a domain adaptation perspective. In *2020 IEEE/CVF conference on computer vision and pattern recognition, CVPR*, pp. 7607–7616.

Japkowicz, N., & Stephen, S. (2002). The class imbalance problem: A systematic study. *Intelligent Data Analysis, 6*(5), 429–449.

Jiang, Z., Zhang, C., Talwar, K., & Mozer, M. C. (2020). Exploring the memorization-generalization continuum in deep learning. CoRR abs/2002.03206

Kang, B., Xie, S., Rohrbach, M., Yan, Z., Gordo, A., Feng, J., & Kalantidis, Y. (2020). Decoupling representation and classifier for long-tailed recognition. In *8th International conference on learning representations, ICLR*.

Kawaguchi, K., & Lu, H. (2020). Ordered SGD: A new stochastic optimization framework for empirical risk minimization. In *The 23rd international conference on artificial intelligence and statistics, AISTATS*, vol. 108, pp. 669–679.

Kawaguchi, K., Kaelbling, L. P., & Bengio, Y. (2017). Generalization in deep learning. CoRR abs/1710.05468.

Lin, T., Goyal, P., Girshick, R. B., He, K., & Dollár, P. (2017). Focal loss for dense object detection. In *IEEE international conference on computer vision, ICCV*, pp. 2999–3007.

Liu, B., & Tsoumakas, G. (2018). Making classifier chains resilient to class imbalance. In *Proceedings of The 10th Asian conference on machine learning, ACML*, PMLR, vol. 95, pp. 280–295.

Liu, Z., Miao, Z., Zhan, X., Wang, J., Gong, B., & Yu, S. X. (2019). Large-scale long-tailed recognition in an open world. In *IEEE conference on computer vision and pattern recognition, CVPR*, pp. 2537–2546.

Ma, Y., Sun, J., Zhou, Q., Cheng, K., Chen, X., & Zhao, Y. (2018). CHS-NET: A cascaded neural network with semi-focal loss for mitosis detection. In *Proceedings of the 10th Asian conference on machine learning, ACML*, PMLR, vol. 95, pp. 161–175.

Mohri, M., Rostamizadeh, A., & Talwalkar, A. (2012). *Foundations of Machine Learning. Adaptive computation and machine learning*. MIT Press.

Mollaysa, A., Kalousis, A., Bruno, E., & Diephuis, M. (2019). Learning to augment with feature side-information. In *Proceedings of the 11th Asian conference on machine learning*, ACML 2019, 17–19 November 2019, Nagoya, Japan, PMLR, vol 101, pp. 173–187.

Ouyang, W., Wang, X., Zhang, C., & Yang, X. (2016). Factors in finetuning deep model for object detection with long-tail distribution. In *2016 IEEE conference on computer vision and pattern recognition, CVPR*, pp. 864–873.

Peng, J., Bu, X., Sun, M., Zhang, Z., Tan, T., & Yan, J. (2020). Large-scale object detection in the wild from imbalanced multi-labels. In *2020 IEEE/CVF conference on computer vision and pattern recognition, CVPR*, pp. 9706–9715.

Rastogi, A. (2011). *McDiarmid's inequality*. US: Springer.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., et al. (2015). Imagenet large scale visual recognition challenge. *International Journal of Computer Vision, 115*(3), 211–252.

Tao, Y., Takagi, K., & Nakata, K. (2018). RDEC: integrating regularization into deep embedded clustering for imbalanced datasets. In *Proceedings of the 10th Asian conference on machine learning, ACML*, PMLR, vol. 95, pp. 49–64.

Toneva, M., Sordoni, A., des Combes, R. T., Trischler, A., Bengio, Y., & Gordon, G. J. (2019). An empirical study of example forgetting during deep neural network learning. In *7th International conference on learning representations, ICLR*.

Wu, T., Huang, Q., Liu, Z., Wang, Y., & Lin, D. (2020). Distribution-balanced loss for multi-label classification in long-tailed datasets. CoRR abs/2007.09654

Xiang, L., & Ding, G. (2020). Learning from multiple experts: Self-paced knowledge distillation for long-tailed classification. CoRR abs/2001.01536

Zhang, C., Bengio, S., Hardt, M., Recht, B., & Vinyals, O. (2017). Understanding deep learning requires rethinking generalization. In *5th International conference on learning representations, ICLR*.

Zhang, C., Ma, X., Liu, Y., Wang, L., Su, Y., & Liu, Y. (2021). Unified regularity measures for sample-wise learning and generalization. CoRR abs/2108.03913.

Zhou, B., Cui, Q., Wei, X., & Chen, Z. (2020). BBN: Bilateral-branch network with cumulative learning for long-tailed visual recognition. In *IEEE Conference on computer vision and pattern recognition, CVPR*, pp. 9716–9725.

Zhou, B., Lapedriza, À., Khosla, A., Oliva, A., & Torralba, A. (2018). Places: A 10 million image database for scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 40*(6), 1452–1464.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Authors and Affiliations

Chi Zhang[1,2] · Benyi Hu[1,2] · Yuhang Liuzhang[2] · Le Wang[1,2] · Li Liu[3] · Yuehu Liu[1,2]

Li Liu
li.liu@inceptioniai.org

Yuehu Liu
liuyh@xjtu.edu.cn

[1] Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University, Xi'an, China

[2] College of Artificial Intelligence, Xi'an Jiaotong University, Xi'an, China

[3] Computer Vision Research Group, Inception Institute of Artificial Intelligence, Abu Dhabi,
Abu Dhabi, United Arab Emirates