Class and Attribute-Aware Logit Adjustment for Generalized Long-Tail Learning

Xiaoling Zhou[1,2], **Ou Wu**[1,3*], **and Nan Yang**[1]

[1]Center for Applied Mathematics, Tianjin University, China
[2] National Engineering Research Center for Software Engineering, Peking University, China
[3] HIAS, University of Chinese Academy of Sciences, Hangzhou, China
{xiaolingzhou, yny, wuou}@tju.edu.cn

Abstract

Compared to conventional long-tail learning, which focuses on addressing class-wise imbalances, generalized long-tail (GLT) learning considers that samples within each class still conform to long-tailed distributions due to varying attributes, known as attribute imbalance. In the presence of such imbalance, the assumption of equivalence between the class-conditional probability densities of the training and testing sets is no longer tenable. Existing GLT approaches typically employ regularization techniques to avoid directly modeling the class-conditional probability density (CCPD) ratio between training and test data, leading to suboptimal performance. This study aims to directly estimate this ratio, for which a novel class-attribute aware logit-adjusted (CALA) loss incorporating both the CCPD ratio and the class priors is presented. Two new GLT learning methods, named Heuristic-CALA and Meta-CALA, are then proposed, which estimate the CCPD ratio in the CALA loss by leveraging the neighborhood information of samples. Extensive experiments across diverse scenarios susceptible to class and attribute imbalances showcase the state-of-the-art performance of Meta-CALA. Furthermore, while Heuristic-CALA exhibits inferior performance compared to Meta-CALA, it incurs only negligible additional training time compared to the Cross-Entropy loss, yet surpasses existing methods by a significant margin.

Introduction

Long-tail (LT) learning is a common challenge in many real-world applications, where only a few categories are represented by a large number of instances while many others are represented by only a few (Cui et al. 2019; Zhou, Yang, and Wu 2023). A popular technique to address this challenge is logit adjustment (Menon et al. 2021; Wang et al. 2024; Zhao et al. 2022). However, existing methods (Tao et al. 2023; Menon et al. 2021; Li et al. 2021) typically assume that the primary difference between training and test data lies in the prior probabilities over categories, where $p_{\rm tr}(y) \neq p_{\rm te}(y)$, while the class-conditional probabilities for the training and test data remain the same, i.e., $p_{\rm tr}(\boldsymbol{x}|y) = p_{\rm te}(\boldsymbol{x}|y)$. Additionally, these methods presume uniform prior probabilities over classes when evaluating model performance. Therefore, the adjustment terms in existing methods (Menon et al. 2021; Cao et al. 2019) are primarily based on $p_{\rm tr}(y)$. Additionally, several imbalanced benchmarks, such as CIFAR-LT (Cui et al. 2019), are manually constructed based on these assumptions, making existing algorithms well-suited for these datasets but limiting their generalizability to others.

Recently, Tang et al. (2022) emphasized that the assumption of identical class-conditional probability densities between training and test data cannot be guaranteed for real-world datasets. They have thus identified a new type of imbalance known as attribute imbalance. For instance, concerning the color attribute, the training set may consist mostly of white doves, whereas the number of white and dark doves may be equal in the testing set. Attribute imbalance can lead to poor performance of samples with rare attributes and compromise the generalization ability of deep learning models. Consequently, they formulated a new research problem, named generalized long-tail (GLT) learning, which encompasses both class and attribute imbalances. An example of GLT learning is shown in Fig. 1. Given the poor performance of existing LT baselines for GLT learning, they proposed a regularization technique to learn invariant features. Despite demonstrating improved performance, the challenge of attribute imbalance remains unsolved because the fundamental issue, $p_{\rm tr}(\boldsymbol{x}|y) \neq p_{\rm te}(\boldsymbol{x}|y)$, has not been adequately addressed.

This study pioneers the estimation of the class-conditional probability density (CCPD) ratio between training and test data. We first introduce a modified Cross-Entropy (CE) loss, termed class-attribute aware logit-adjusted (CALA) loss, which incorporates both class priors and the ratio of class-conditional probability densities as adjustment terms to address class and attribute imbalances. Next, we develop a novel GLT method called Heuristic-CALA, which utilizes neighborhood information of samples to estimate the CCPD ratio within the CALA loss. Notably, Heuristic-CALA serves as a generalization of several conventional LT approaches. Finally, leveraging the strong performance of meta-learning, we propose another GLT method named Meta-CALA. This method employs an adjustment network optimized through meta-learning to estimate the CCPD ratio based on the neighborhood-related training characteristics of samples. We conduct extensive experiments across various learning scenarios prone to class and attribute imbalances: LT learning, GLT learning, and subpopulation shift learn-

*Corresponding author.

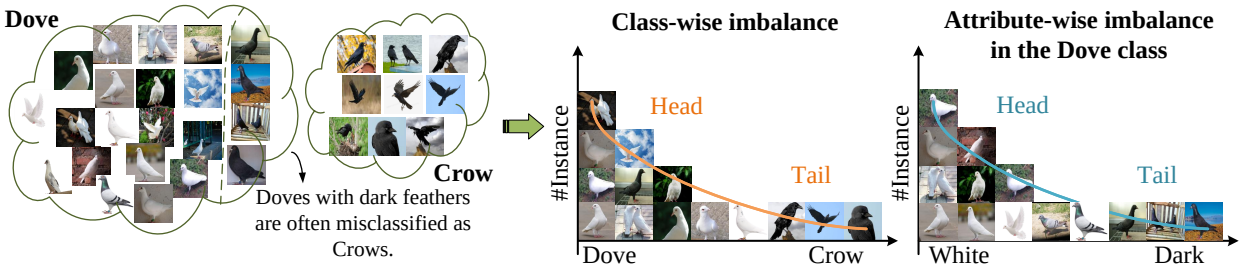

Figure 1: Illustration of imbalances at both the class and attribute levels. Dove and Crow represent the head and tail classes, respectively. Due to the prevalence of white feathers among doves, there exists an attribute imbalance within the Dove class.

ing. The results demonstrate that our methods consistently achieve state-of-the-art (SOTA) performance by effectively addressing both class and attribute imbalances.

Our main contributions can be summarized as follows:

- We conduct a pioneering exploration by directly utilizing the CCPD ratio for logit calibration. A novel logit adjustment loss (termed CALA) that accounts for both class priors and the ratio of class-conditional probability densities between training and test data is then presented.
- We propose two new GLT learning methods, Heuristic-CALA and Meta-CALA, which employ K-neighborhood-based and meta-learning-based estimation approaches, respectively, to estimate the CCPD ratio in the CALA loss.
- We conduct extensive experiments across three learning scenarios susceptible to class and attribute imbalances. The results conclusively demonstrate the effectiveness of our approaches in enhancing the generalization and robustness of deep learning models.

Related Work

Long-Tail Classification Despite success in various applications, deep neural networks still struggle with long-tailed datasets (De Alvis and Seneviratne 2024; Mao, Fan, and Li 2023). Different approaches have been proposed to address this issue, including algorithms based on resampling (Lin, Tsai, and Lin 2023; Tripathi, Chakraborty, and Kopparapu 2021; Yan et al. 2019), reweighting (Wan et al. 2023; Cui et al. 2019), knowledge distillation (Zhang et al. 2023), data augmentation (Li et al. 2021; Zhou et al. 2024; Zhou and Wu 2023), multiple experts (Wang et al. 2020; Xiang, Ding, and Han 2020), and contrastive learning (Cui et al. 2021). Among these methods, logit adjustment-based approaches (Wang et al. 2024; Li, Cheung, and Lu 2022; Menon et al. 2021) have gained popularity and demonstrated their effectiveness. For instance, LA (Menon et al. 2021) perturbs the logits of samples to encourage a large relative margin between logits of rare versus dominant labels. More recently, ALA (Zhao et al. 2022) introduces an adaptive adjustment term that consists of two complementary factors: a quantity factor and a difficulty factor. However, existing methods (Wang et al. 2024; Tao et al. 2023) primarily focus on addressing class-wise imbalances, while overlooking the imbalanced attribute distribution within each class.

Generalized Long-Tail Classification Tang et al. (2022) argued that imbalanced classifications suffer from both class- and attribute-wise imbalances and, therefore, proposed the GLT learning task. They subsequently presented an invariant feature learning (IFL) method to tackle GLT learning by maintaining the feature center of each class across different environments. Apart from this approach, there are currently few dedicated solutions available to address the emerging GLT problem. Nevertheless, some methods tailored for addressing subpopulation shift (Deng et al. 2024; Liang and Zou 2022; Koh et al. 2021) and spurious correlation (Chen et al. 2023; Srivastava, Hashimoto, and Liang 2020; Agarwal, Shetty, and Fritz 2020) are deemed effective for tackling GLT learning by implicitly mitigating the issue of attribute imbalance. However, nearly all existing studies overlook the direct modeling of attribute distributions within each class, leading to subpar performance when dealing with attribute imbalance.

Class-Attribute Aware Logit-Adjusted Loss

Following prior studies, the classification model is formulated as $p(y|\boldsymbol{x})$, which predicts the label y from the input $\boldsymbol{x}$. The training and test data are drawn from different joint distributions, namely $p_{\text{tr}}(\boldsymbol{x}, y)$ and $p_{\text{te}}(\boldsymbol{x}, y)$, respectively. Utilizing Bayes' Rule, we have $p_{\text{tr}}(y|\boldsymbol{x}) \propto p_{\text{tr}}(\boldsymbol{x}|y)p_{\text{tr}}(y)$ and $p_{\text{te}}(y|\boldsymbol{x}) \propto p_{\text{te}}(\boldsymbol{x}|y)p_{\text{te}}(y)$. Hence, we arrive at

$$p_{\text{tr}}(y|\boldsymbol{x}) \propto p_{\text{te}}(y|\boldsymbol{x}) \cdot \frac{p_{\text{tr}}(\boldsymbol{x}|y)}{p_{\text{te}}(\boldsymbol{x}|y)} \cdot \frac{p_{\text{tr}}(y)}{p_{\text{te}}(y)}. \tag{1}$$

To simplify Eq. (1), existing methods commonly rely on the following two assumptions:

Assumption 1 *The class-conditional probability densities of the training and testing sets are equal:* $\forall \boldsymbol{x}, y, p_{\text{tr}}(\boldsymbol{x}|y)/p_{\text{te}}(\boldsymbol{x}|y) \equiv 1$.

Assumption 2 *The class priors* $p_{\text{te}}(y)$ *are assumed to be identical when evaluating the model's performance.*

Given the two assumptions mentioned above, the objective of Eq. (1) can be expressed as

$$\arg\max p_{\text{te}}(y|\boldsymbol{x}) = \arg\max p_{\text{tr}}(y|\boldsymbol{x})/p_{\text{tr}}(y). \tag{2}$$

From Eq. (2), the adjustment terms should be determined by the class prior $p_{\text{tr}}(y)$, which has been adopted by existing LT learning proposals, such as LDAM (Cao et al. 2019) and LA (Menon et al. 2021).

Assumption 2 evidently promotes fairness across different categories. However, we challenge the validity of Assumption 1 in practical learning scenarios and argue that the objective function in Eq. (2) is overly simplified, rendering it

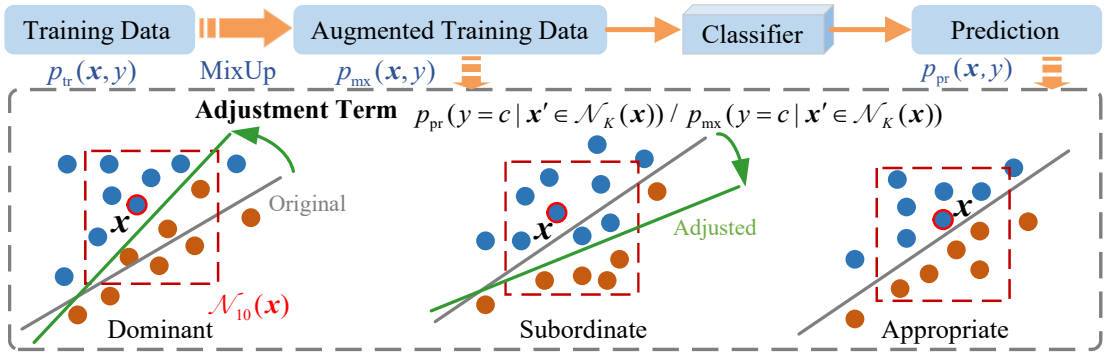

Figure 2: Illustration of Heuristic-CALA. Blue and orange dots represent samples from two classes. The gray and green lines denote the original and adjusted classifiers. The red boxes indicate the ten-nearest neighborhood $\mathcal{N}_{10}(\boldsymbol{x})$. If $p_{\text{pr}}(y=y_{\boldsymbol{x}}|\boldsymbol{x}'\in\mathcal{N}_K(\boldsymbol{x}))>(<,=)p_{\text{mx}}(y=y_{\boldsymbol{x}}|\boldsymbol{x}'\in\mathcal{N}_K(\boldsymbol{x}))$, then $\boldsymbol{x}$ is in a dominant (subordinate, appropriate) position in the training set. Our adjustment term will make $\boldsymbol{x}$ easier (or harder, or leave it unchanged) than before, facilitating better adjusting the classifier.

incapable of resolving a number of issues, such as the misclassification of samples with rare attributes. As illustrated in Fig. 1, although the number of samples in the Dove class exceeds that of the Crow class, doves with dark feathers are often misclassified as crows due to the fact that $p_{\text{tr}}(\boldsymbol{x}_{\text{feather}}=\text{white}|y=\text{Dove}) \gg p_{\text{tr}}(\boldsymbol{x}_{\text{feather}}=\text{dark}|y=\text{Dove})$ and $p_{\text{tr}}(\boldsymbol{x}_{\text{feather}}=\text{white}|y=\text{Crow}) \ll p_{\text{tr}}(\boldsymbol{x}_{\text{feather}}=\text{dark}|y=\text{Crow})$[1]. Even if the issue of class imbalance is addressed, the presence of attribute imbalance remains and substantially impairs the generalization performance of models.

Actually, the training objective without guarantying Assumption 1 should be

$$\arg\max p_{\text{te}}(y|\boldsymbol{x}) = \arg\max p_{\text{tr}}(y|\boldsymbol{x}) \cdot \frac{p_{\text{te}}(\boldsymbol{x}|y)}{p_{\text{tr}}(\boldsymbol{x}|y)} \cdot \frac{1}{p_{\text{tr}}(y)}. \tag{3}$$

Eq. (3) suggests that the adjustment terms should be determined by both the CCPD ratio and the class priors. Accordingly, building on the inference method used in LA (Menon et al. 2021), we derive a novel logit-adjusted loss that incorporates these two terms to mitigate attribute imbalance and class bias. With $\tau_1, \tau_2 > 0$, the resulting loss, termed CALA, is as follows:

$$\begin{aligned}\ell_{CALA}(\boldsymbol{x}) &= -\log \frac{\exp[f_y(\boldsymbol{x}) + \tau_1 \log p_{\text{tr}}(y) + \tau_2 \log \frac{p_{\text{tr}}(\boldsymbol{x}|y)}{p_{\text{te}}(\boldsymbol{x}|y)}]}{\sum_{y'\in[\mathcal{C}]} \exp[f_{y'}(\boldsymbol{x}) + \tau_1 \log p_{\text{tr}}(y') + \tau_2 \log \frac{p_{\text{tr}}(\boldsymbol{x}|y')}{p_{\text{te}}(\boldsymbol{x}|y')}]} \\ &= \log[1 + \sum_{y'\neq y} \left(\frac{p_{\text{tr}}(y')}{p_{\text{tr}}(y)}\right)^{\tau_1} \cdot \left(\frac{p_{\text{tr}}(\boldsymbol{x}|y')p_{\text{te}}(\boldsymbol{x}|y)}{p_{\text{te}}(\boldsymbol{x}|y')p_{\text{tr}}(\boldsymbol{x}|y)}\right)^{\tau_2} \cdot \frac{e^{f_{y'}(\boldsymbol{x})}}{e^{f_y(\boldsymbol{x})}}],\end{aligned} \tag{4}$$

where $f(\cdot)$ and $\mathcal{C}$ represent the classifier and the number of classes. We then explain the CALA loss from a regularization perspective using Taylor expansion. Our analysis indicates that the CALA loss imposes significant penalties on samples from tail classes (e.g., Crow) and those with rare attributes (e.g., dark doves). This increased penalization amplifies the impact of these samples during model training, thereby enhancing their prediction performance. Detailed regularization analyses are provided in Section A of the Appendix. However, directly obtaining the CCPD ratio $p_{\text{tr}}(\boldsymbol{x}|y)/p_{\text{te}}(\boldsymbol{x}|y)$ is impossible due to the unknown $p_{\text{te}}(\boldsymbol{x}|y)$. To this end, we propose two estimation approaches, as stated in the subsequent sections.

[1]It is worth noting that $p_{\text{te}}(\boldsymbol{x}_{\text{feather}}=\text{white}|y=\text{Dove}) = p_{\text{te}}(\boldsymbol{x}_{\text{feather}}=\text{dark}|y=\text{Dove})$ is assumed to be established for fairness. Consequently, $p_{\text{tr}}(\boldsymbol{x}|y=\text{Dove}) \neq p_{\text{te}}(\boldsymbol{x}|y=\text{Dove})$.

Learning with CALA Loss

To estimate the CCPD ratio in the CALA loss, we propose two methods: one based on K-neighborhood and the other based on meta-learning. Consequently, two logit adjustment approaches, named Heuristic-CALA and Meta-CALA, are devised for GLT learning.

Heuristic-CALA Framework

To simplify the notation, the CALA loss is expressed as

$$\ell_{CALA}(\boldsymbol{x}) = -\log \frac{\exp[f_y(\boldsymbol{x}) + \tau_1 u(y) + \tau_2 v(\boldsymbol{x},y)]}{\sum_{y'\in[\mathcal{C}]} \exp[f_{y'}(\boldsymbol{x}) + \tau_1 u(y') + \tau_2 v(\boldsymbol{x},y')]}, \tag{5}$$

where $u(y) = \log p_{\text{tr}}(y)$ and $v(\boldsymbol{x},y) = \log[p_{\text{tr}}(\boldsymbol{x}|y)/p_{\text{te}}(\boldsymbol{x}|y)]$. The neighborhood information of each sample reflects its local distribution in the feature space, thus providing an estimate of the CCPD values. Accordingly, we establish the following relationship, with detailed inference provided in Section B.I of the Appendix:

$$p_{\text{te}}(y|\boldsymbol{x}) \approx p_{\text{tr}}(y|\boldsymbol{x}) \cdot \frac{p_{\text{te}}\left(y|\boldsymbol{x}'\in\mathcal{N}_K(\boldsymbol{x})\right)}{p_{\text{tr}}\left(y|\boldsymbol{x}'\in\mathcal{N}_K(\boldsymbol{x})\right)}, \tag{6}$$

where $\mathcal{N}_K(\boldsymbol{x})$ represents the K-nearest neighbors of $\boldsymbol{x}$. Eq. (6) manifests that the adjustment terms should be determined by $p_{\text{te}}\left(y|\boldsymbol{x}'\in\mathcal{N}_K(\boldsymbol{x})\right)/p_{\text{tr}}\left(y|\boldsymbol{x}'\in\mathcal{N}_K(\boldsymbol{x})\right)$. However, the distribution of the test data is not readily available. Zhang et al. (2018) proposed a generic vicinal distribution called MixUp, which effectively estimates the unknown data distribution and significantly improves models' generalization performance. Consequently, we employ the training data augmented using MixUp, denoted as $p_{\text{mx}}(\boldsymbol{x},y)$, to approximate the test data. However, MixUp applied to imbalanced datasets fails to balance the label distribution. To address this, we utilize the ratio $p_{\text{mx}}(y|\boldsymbol{x}'\in\mathcal{N}_K(\boldsymbol{x}))/p_{\text{tr}}(y)$

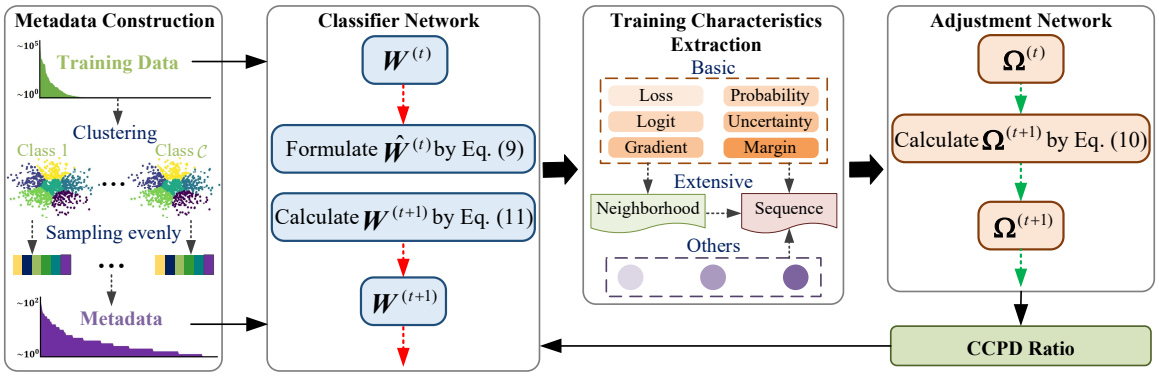

Figure 3: Illustration of Meta-CALA, which contains four main components, including the metadata construction module, the classifier network, the training characteristics extraction module, and the adjustment network.

to substitute $p_{\text{te}}(y|\boldsymbol{x}' \in \mathcal{N}_K(\boldsymbol{x}))$. Furthermore, $p_{\text{tr}}(y|\boldsymbol{x}' \in \mathcal{N}_K(\boldsymbol{x}))$ is replaced by the predicted probability $p_{\text{pr}}(y|\boldsymbol{x}' \in \mathcal{N}_K(\boldsymbol{x}))$ to more precisely and dynamically adjust the classifier during training. Thus, Eq. (6) transforms into

$$p_{\text{te}}(y|\boldsymbol{x}) := p_{\text{tr}}(y|\boldsymbol{x}) \cdot \frac{p_{\text{mx}}(y|\boldsymbol{x}' \in \mathcal{N}_K(\boldsymbol{x}))}{p_{\text{pr}}(y|\boldsymbol{x}' \in \mathcal{N}_K(\boldsymbol{x}))} \cdot \frac{1}{p_{\text{tr}}(y)}. \quad (7)$$

We observe that Eq. (7) employs the ratio $p_{\text{pr}}(y|\boldsymbol{x}' \in \mathcal{N}_K(\boldsymbol{x}))/p_{\text{mx}}(y|\boldsymbol{x}' \in \mathcal{N}_K(\boldsymbol{x}))$ to replace the CCPD ratio $p_{\text{tr}}(\boldsymbol{x}|y)/p_{\text{te}}(\boldsymbol{x}|y)$ in Eq. (3). Consequently, the adjustment term $v(\boldsymbol{x}, y)$ can be calculated as

$$v(\boldsymbol{x}, y) = \log[\frac{p_{\text{pr}}(y|\boldsymbol{x}' \in \mathcal{N}_K(\boldsymbol{x}))}{p_{\text{mx}}(y|\boldsymbol{x}' \in \mathcal{N}_K(\boldsymbol{x}))}], \quad (8)$$

where $p_{\text{mx}}(y = c|\boldsymbol{x}' \in \mathcal{N}_K(\boldsymbol{x}))$ represents the label distribution of samples within the neighborhood. Additionally, $p_{\text{pr}}(y = c|\boldsymbol{x}' \in \mathcal{N}_K(\boldsymbol{x}))$ denotes the prediction distribution within the neighborhood, which can be calculated as $\sum_{\boldsymbol{x}' \in \mathcal{N}_K(\boldsymbol{x})} q_{\boldsymbol{x}',c}/K$, where $\boldsymbol{q}_{\boldsymbol{x}'} = \text{Softmax}(f(\boldsymbol{x}'))$ represents the probability vector. Furthermore, since neighborhood information can be sensitive to border effects and outliers, we employ class averages of the corresponding values within the neighborhood to smooth the values of $p_{\text{pr}}(y|\boldsymbol{x}' \in \mathcal{N}_K(\boldsymbol{x}))$ and $p_{\text{mx}}(y|\boldsymbol{x}' \in \mathcal{N}_K(\boldsymbol{x}))$. Detailed algorithmic procedures are provided in Section B.II of the Appendix.

We then validate the rationality of our approach by elaborating on its meaning. The term $p_{\text{pr}}(y|\boldsymbol{x}' \in \mathcal{N}_K(\boldsymbol{x}))/p_{\text{mx}}(y|\boldsymbol{x}' \in \mathcal{N}_K(\boldsymbol{x}))$ reflects the dominance of $\boldsymbol{x}$ in the training data. As shown in Fig. 2, if $p_{\text{pr}}(y = y_{\boldsymbol{x}}|\boldsymbol{x}' \in \mathcal{N}_K(\boldsymbol{x})) > (<, =) p_{\text{mx}}(y = y_{\boldsymbol{x}}|\boldsymbol{x}' \in \mathcal{N}_K(\boldsymbol{x}))$, then $\boldsymbol{x}$ is in a dominant (subordinate, appropriate) position in the training set. Our adjustment term will make $\boldsymbol{x}$ easier (or harder, or leave it unchanged) than before, resulting in a smaller (or larger, or unchanged) impact on the model training. Generally, samples from tail classes and those with rare attributes occupy a subordinate position, and their influence will be amplified by our approach.

Furthermore, we demonstrate that several typical LT baselines can be viewed as special cases of Heuristic-CALA. Indeed, the adjustment term for class c in Heuristic-CALA is determined by $p_{\text{tr}}(y = c) \cdot \frac{p_{\text{pr}}(y=c|\boldsymbol{x}' \in \mathcal{N}_K(\boldsymbol{x}))}{p_{\text{mx}}(y=c|\boldsymbol{x}' \in \mathcal{N}_K(\boldsymbol{x}))}$. As when $K = 0$, $\mathcal{N}_0(\boldsymbol{x}) = \boldsymbol{x}$, we have the following findings:

- If $K = 0$ and $p_{\text{pr}}(y = y_{\boldsymbol{x}}|\boldsymbol{x}) \equiv 1$, as $p_{\text{mx}}(y = y_{\boldsymbol{x}}|\boldsymbol{x}) \equiv 1$, then only the adjustment term for class $y_{\boldsymbol{x}}$ is non-zero and equals to $p_{\text{tr}}(y = y_{\boldsymbol{x}})$, relying on the class proportion to adjust the logit of $y_{\boldsymbol{x}}$. Therefore, Heuristic-CALA is equivalent to LDAM (Cao et al. 2019) in this scenario.
- If $K = 0$ and $p_{\text{pr}}(y = y_{\boldsymbol{x}}|\boldsymbol{x}) \neq 1$, as $p_{\text{mx}}(y = y_{\boldsymbol{x}}|\boldsymbol{x}) \equiv 1$, then Heuristic-CALA's adjustment term for class $y_{\boldsymbol{x}}$ is $p_{\text{tr}}(y = y_{\boldsymbol{x}})p_{\text{pr}}(y = y_{\boldsymbol{x}}|\boldsymbol{x})$, considering both the class proportion and model prediction. This adjustment term is equivalent to that of ALA (Zhao et al. 2022).
- When $K = +\infty$, $p_{\text{mx}}(y = c|\boldsymbol{x}' \in \mathcal{N}_{+\infty}(\boldsymbol{x}))$ approximates $p_{\text{tr}}(y = c)$. Thus, the adjustment term for the cth class in Heuristic-CALA is $p_{\text{pr}}(y = c)$ which is similar to LA (Menon et al. 2021). The distinction is that Heuristic-CALA in this case employs the class proportions of the predicted labels, which vary with model performance. We have verified that this approach is superior and more rational compared to LA. The comparisons are detailed in Section D.VII of the Appendix.

Meta-CALA Framework

Leveraging the universal approximation capability of deep neural networks, we introduce an adjustment network to estimate the CCPD ratio of samples. The classifier and the adjustment network are alternately updated using a meta-learning-based optimization strategy. Consequently, another GLT method, termed Meta-CALA, is presented. Fig. 3 illustrates the pipeline of the Meta-CALA framework, which consists of four primary components: metadata construction, classifier network, training characteristics extraction, and adjustment network.

We first construct a metadata set with balanced classes and attributes to represent the meta-knowledge of the ground-truth distribution. This metadata set is then utilized to train the adjustment network. To ensure class and attribute balance as much as possible, samples from each class in the training data are clustered into six groups using KMeans with a pre-trained ResNet-50 model (He et al. 2016). We then evenly sample instances from each group and class, as illustrated in the first box of Fig. 3.

Considering that the CCPD ratio can be reflected by the neighborhood information of samples, we extract a series of

Dataset	CIFAR10-LT		CIFAR100-LT	
Imbalance ratio	100:1	10:1	100:1	10:1
Class-Balanced CE (Cui et al. 2019)	72.68%	86.90%	38.77%	57.57%
Class-Balanced Focal (Cui et al. 2019)	74.57%	87.48%	39.60%	57.99%
LDAM-DRW (Cao et al. 2019)	78.12%	88.37%	42.89%	58.78%
De-confound-TDE (Tang et al. 2020)	80.60%	88.50%	44.10%	59.60%
LA (Menon et al. 2021)	77.67%	88.93%	43.89%	58.34%
MiSLAS* (Zhong et al. 2021)	82.10%	90.00%	47.00%	<u>63.20%</u>
LADE (Hong et al. 2021)	81.17%	89.15%	45.42%	61.69%
GLC (Li, Cheung, and Lu 2022)	<u>82.68%</u>	89.81%	<u>48.71%</u>	62.97%
ALA (Zhao et al. 2022)	77.65%	88.32%	43.67%	58.92%
LDAM-DRW-SAFA (Hong et al. 2022)	80.48%	88.94%	46.04%	59.11%
BKD (Zhang et al. 2023)	82.50%	89.50%	46.50%	62.00%
CSA (Shi et al. 2023)	82.53%	<u>90.80%</u>	46.61%	62.60%
Heuristic-CALA (Ours)	**83.91%**	**91.78%**	**50.53%**	**64.34%**
Meta-Weight-Net (Shu et al. 2019)	73.57%	87.55%	41.61%	58.91%
MetaSAug (Li et al. 2021)	<u>80.54%</u>	<u>89.44%</u>	<u>46.87%</u>	<u>61.73%</u>
Meta-CALA (Ours)	**84.79%**	**92.47%**	**52.34%**	**65.51%**

Table 1: Accuracy comparison on the CIFAR-LT benchmark. Bold and underlined numbers are the best and second-best results, respectively.

neighborhood-related training characteristics from the classifier and feed them into the adjustment network to estimate the CCPF ratio $v(\boldsymbol{x}, y)$, thereby obtaining the adjustment vector $\boldsymbol{\delta}_{\boldsymbol{x}} = [v(\boldsymbol{x}, y_1), \cdots, v(\boldsymbol{x}, y_C)]$. The characteristics extraction module is depicted in the third box of Fig. 3. We first extract six basic characteristics that reflect the learning difficulty of samples, including sample loss, logit vector, loss gradient, probability vector, uncertainty which is quantified by the information entropy of the Softmax output, and sample margin. Regarding the logit and probability characteristics, we utilize their values specific to the ground-truth category. Subsequently, we consider the neighborhood extensions of the six basic characteristics. First, we compute the mean values of these characteristics for the samples in the neighborhood. Second, we establish the disparities between the sample's characteristic values and the neighborhood's average values. Additionally, we incorporate three other characteristics: 1) the ratio of samples sharing the same label in the neighborhood, 2) the ratio of heterogeneous samples with the highest proportion in the neighborhood, and 3) the cosine distance between the deep feature of the sample and the average feature of the samples in the neighborhood. Furthermore, all characteristics can be extended through the sequence by considering the differences in these training characteristics between the current and previous epochs. In summary, a total number of 42 characteristics are finally extracted. The calculations of all characteristics are detailed in Section C.I of the Appendix.

As the training characteristics are tabular data, we employ a two-layer Multilayer Perceptron as the adjustment network. Moreover, a meta-learning-based learning strategy is proposed to alternatively update the parameters in the classifier $\boldsymbol{W}$ and the adjustment network $\boldsymbol{\Omega}$, as shown in the second and fourth boxes of Fig. 3. Denote the training data as $\mathcal{D}^{\text{tr}} = \{\boldsymbol{x}_i, y_i\}_{i=1}^N$ and the metadata as $\mathcal{D}^{\text{meta}} = \{\boldsymbol{x}_i^{\text{meta}}, y_i^{\text{meta}}\}_{i=1}^M$. First, a batch of training samples $\{\boldsymbol{x}_i, y_i\}_{i=1}^n$ is selected, where n is the batch size and the updating of $\boldsymbol{W}$ is formulated as

$$\hat{\boldsymbol{W}}^{(t)} \leftarrow \boldsymbol{W}^{(t)} - \eta_1 \frac{1}{n} \sum\nolimits_{i=1}^{n} \nabla_{\boldsymbol{W}} \ell_{CALA}\left(f(\boldsymbol{x}_i), y_i; \boldsymbol{\delta}_i^{(t)}\right), \tag{9}$$

Dataset	iNat 2018	Places-LT
CE loss	65.76%	30.20%
Decoupling (Kang et al. 2020)	69.49%	37.62%
LA (Menon et al. 2021)	66.36%	34.23%
DisAlign (Zhang et al. 2021)	70.06%	39.30%
MisLAS (Zhong et al. 2021)	71.51%	40.15%
LADE (Hong et al. 2021)	70.00%	38.87%
GCL (Li, Cheung, and Lu 2022)	<u>72.01%</u>	<u>42.64%</u>
LDAM-DRS-SAFA (Hong et al. 2022)	69.78%	41.53%
BKD (Zhang et al. 2023)	71.20%	38.92%
Heuristic-CALA (Ours)	**73.23%**	**43.42%**
Meta-Weight-Net (Shu et al. 2019)	67.95%	37.14%
MetaSAug (Li et al. 2021)	<u>68.75%</u>	<u>39.83%</u>
Meta-CALA (Ours)	**74.05%**	**43.97%**

Table 2: Accuracy comparison on the iNat 2018 and Places-LT benchmarks.

where η_1 is the step size and $\boldsymbol{\delta}_i$ represents the adjustment vector for sample $\boldsymbol{x}_i$. Then, the parameters of the adjustment network $\boldsymbol{\Omega}$ can be updated on a minibatch of metadata $\{\boldsymbol{x}_i^{\text{meta}}, y_i^{\text{meta}}\}_{i=1}^m$, with the following formula:

$$\boldsymbol{\Omega}^{(t+1)} \leftarrow \boldsymbol{\Omega}^{(t)} - \eta_2 \frac{1}{m} \sum\nolimits_{i=1}^{m} \nabla_{\boldsymbol{\Omega}} \ell_{CE}\left(f_{\hat{\boldsymbol{W}}}(\boldsymbol{x}_i^{\text{meta}}), y_i^{\text{meta}}\right), \tag{10}$$

where m and η_2 are the minibatch size of metadata and the step size, respectively. Finally, the parameters of the classifier are updated using the resulting adjustment terms:

$$\boldsymbol{W}^{(t+1)} \leftarrow \boldsymbol{W}^{(t)} - \eta_1 \frac{1}{n} \sum\nolimits_{i=1}^{n} \nabla_{\boldsymbol{W}} \ell_{CALA}\left(f(\boldsymbol{x}_i), y_i; \boldsymbol{\delta}_i^{(t+1)}\right). \tag{11}$$

Utilizing the aforementioned steps, both the classifier and the adjustment network can be effectively optimized.

Experimental Investigation

We evaluate the performance of our methods in addressing class imbalance, attribute imbalance, and their combination across three typical learning scenarios, including LT learning, subpopulation shift learning, and GLT learning. All experiments are repeated three times using different random seeds. Due to space constraints, details regarding the comparison methods and datasets are included in Section D of the Appendix.

Experiments for Class Imbalance

Three LT benchmarks are evaluated: CIFAR-LT (Cui et al. 2019), Places-LT (Liu et al. 2019), and iNaturalist (iNat) 2018. The imbalance ratios for the CIFAR-LT benchmark are set to 100:1 and 10:1. For all experiments, we utilize the SGD optimizer with a momentum of 0.9. For CIFAR-LT, we primarily follow Cao et al. (2019) and train all models with a ResNet-32 (He et al. 2016) backbone on a single GPU, employing a multistep learning rate schedule that reduces the learning rate by a factor of 0.01 at the 160th and 180th epochs. For Places-LT and iNat 2018, we mainly follow Kang et al. (2020) and use the cosine learning rate schedule (Loshchilov and Hutter 2016) to train the ResNet-152 and ResNet-50 backbones, respectively. For the hyper-parameters in CALA, the neighborhood size K is selected

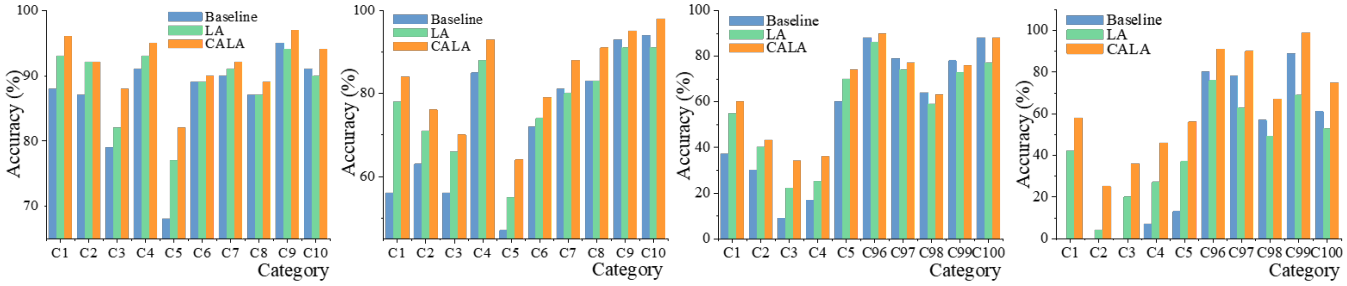

Figure 4: Comparison of class-wise accuracy among baseline (CE loss), LA, and Heuristic-CALA on CIFAR10 with imbalance ratios of 10:1 (a) and 100:1 (b). Accuracy of the top and bottom five categories on CIFAR100 with an imbalance ratio of 10:1 (c) and 100:1 (d). Moving left to right, the categories progress from tail to head.

Dataset	Waterbirds		CMNIST	
Method	Avg.	Worst	Avg.	Worst
CORAL (Sun and Saenko 2016)	90.3%	79.8%	71.8%	69.5%
IRM (Arjovsky et al. 2019)	87.5%	75.6%	72.1%	70.3%
GroupDRO (Sagawa et al. 2020)	91.8%	90.6%	72.3%	68.6%
DomainMix (Xu et al. 2020)	76.4%	53.0%	51.4%	48.0%
IB-IRM (Ahuja et al. 2021)	88.5%	76.5%	72.2%	70.7%
V-REx (Krueger et al. 2021)	88.0%	73.6%	71.7%	70.2%
Fish (Shi et al. 2022)	85.6%	64.0%	46.9%	35.6%
LISA (Yao et al. 2022)	91.8%	89.2%	74.0%	73.3%
COSMOS (Chen et al. 2023)	91.7%	89.3%	73.5%	72.4%
PDE (Deng et al. 2024)	92.4%	90.5%	78.1%	75.9%
Heuristic-CALA (Ours)	**94.3%**	**91.8%**	**79.5%**	**77.0%**

Table 3: Comparison of the average and worst-group accuracy on two subpopulation shift datasets.

from $\{20, 40, 60, 80, 100\}$ for all experiments unless noted. τ_1 and τ_2 are set to 1.5 and 1, respectively. The metadata size is 3,000 for CIFAR-LT. For iNat 2018 and Places-LT, one image is selected per class and group to construct the metadata. In Meta-CALA, the adjustment network is optimized using Adam with an initial learning rate of 1×10^{-3}.

Results. Table 1 presents the comparison results on CIFAR-LT, while Table 2 shows the results on the iNat 2018 and Places-LT datasets, with some results sourced from the original papers. The results are divided into two groups based on the usage of meta-learning. Our proposed methods consistently achieve SOTA performance across various datasets and imbalance ratios. Specifically, Heuristic-CALA surpasses the best compared baselines by 1.11% and 1.48% for CIFAR10-LT and CIFAR100-LT, respectively. Moreover, it outperforms the best compared baselines by 1.22% and 0.78% for the iNat 2018 and Places-LT benchmarks, respectively. Notably, Meta-IADA achieves even superior performance compared to Heuristic-CALA, as it leverages the metadata distribution to adjust the model during training. The accuracy of each class for the three methods, including CE loss, LA, and Heuristic-CALA, is compared in Fig. 4. While LA improves the accuracy of the tail classes, it compromises the performance of some head classes. Conversely, our approach demonstrates optimal performance without adversely affecting the head classes. Additionally, we utilize the Wilcoxon signed-rank test to establish the significance of our performance improvement. The obtained p-value of 0.03 signifies a statistically significant enhancement.

The superior performance of CALA compared to other LT learning methods provides evidence of attribute imbalances in LT datasets, an aspect that has usually been overlooked by previous LT baselines. Furthermore, our approach surpasses De-confound-TDE, which utilizes causal intervention during training and counterfactual reasoning during inference, demonstrating the effectiveness of CALA in mitigating spurious correlations induced by class and attribute imbalances. We then analyze the distinct characteristics of Heuristic-CALA and Meta-CALA. Although Meta-CALA necessitates an additional metadata set and increases time complexity, it achieves SOTA performance by adjusting model training using a high-quality meta dataset. In contrast, Heuristic-CALA does not require a metadata set and incurs only a marginal increase in computational time compared to the CE loss. Although Heuristic-CALA demonstrates lower performance compared to Meta-CALA, it significantly outperforms existing methods. Detailed comparisons of training times are provided in Section D.VIII of the Appendix.

Experiments for Attribute Imbalance

Three subpopulation shift datasets are adopted: CMNIST (Arjovsky et al. 2019), Waterbirds (Sagawa et al. 2020), and CelebA (Liu et al. 2015), each exhibiting significant attribute imbalances within classes. Taking the Waterbirds dataset as an example, it aims to classify birds as either "waterbirds" or "landbirds," with the spurious attribute being the scene context of "water" or "land." Two groups ("land", "waterbird") and ("water", "landbird") are minority groups. The experimental settings follow Yao et al. (2022), utilizing the ResNet-50 model as the backbone network. Since none of the compared methods rely on meta-learning, we include only Heuristic-CALA in the comparison. We set both τ_1 and τ_2 to 1, and K to 10. Performance is evaluated using both average and worst-group accuracy metrics.

Results. Table 3 presents the comparison results on the Waterbirds and CMNIST datasets, while those for CelebA are provided in the Appendix. Our approach surpasses other invariant learning methods in both average and worst-group accuracy. Specifically, Heuristic-CALA surpasses the second-best baselines by 1.65% in average accuracy and 1.15% in worst-group accuracy. This demonstrates its effectiveness in improving model generalization and enhancing performance for samples with rare attributes.

Benchmark	ImageNet-GLT						MSCOCO-GLT					
Protocol	CLT		ALT		GLT		CLT		ALT		GLT	
Method	Acc.	Prec.	Acc.	Prec.	Acc.	Prec.	Acc.	Prec.	Acc.	Prec.	Acc.	Prec.
CE loss	42.52%	47.92%	41.73%	41.74%	34.75%	40.65%	72.34%	76.61%	50.17%	50.94%	63.79%	70.52%
MixUp (Zhang et al. 2018)	38.81%	45.41%	42.11%	42.42%	31.55%	37.44%	74.22%	78.61%	48.90%	49.53%	64.45%	71.13%
LDAM (Cao et al. 2019)	46.74%	46.86%	42.66%	41.80%	38.54%	39.08%	75.57%	77.70%	55.52%	56.21%	67.26%	70.70%
cRT (Kang et al. 2020)	45.92%	45.34%	41.59%	41.43%	37.57%	37.51%	73.64%	75.84%	49.97%	50.37%	64.69%	68.33%
De-confound-TDE (Tang et al. 2020)	45.70%	44.48%	41.40%	42.36%	37.56%	37.00%	73.79%	74.90%	50.76%	51.68%	66.07%	68.20%
BLSoftmax (Ren et al. 2020)	45.79%	46.27%	41.32%	41.37%	37.09%	38.08%	72.64%	75.25%	49.72%	50.65%	64.07%	68.59%
BBN (Zhou et al. 2020)	46.46%	49.86%	43.26%	43.86%	37.91%	41.77%	73.69%	77.35%	51.83%	51.77%	64.48%	70.20%
RandAug (Cubuk et al. 2020)	46.40%	52.13%	46.29%	46.32%	38.24%	44.74%	76.81%	79.88%	53.69%	54.71%	67.71%	72.73%
LA (Menon et al. 2021)	46.53%	45.56%	41.73%	41.74%	37.80%	37.56%	75.50%	76.88%	50.17%	50.94%	66.17%	68.35%
IFL (Tang et al. 2022)	45.97%	52.06%	45.89%	46.42%	37.96%	44.47%	74.31%	78.90%	52.86%	53.49%	65.31%	72.24%
RISDA (Chen et al. 2022)	46.31%	51.24%	43.65%	43.23%	38.45%	42.77%	74.34%	78.27%	51.58%	52.28%	66.85%	71.36%
CSA (Shi et al. 2023)	46.49%	50.77%	43.03%	44.05%	37.22%	42.01%	74.25%	78.56%	52.34%	52.11%	64.78%	69.10%
BKD (Zhang et al. 2023)	46.51%	50.15%	42.17%	41.83%	37.93%	41.50%	75.82%	78.23%	51.88%	51.23%	65.48%	70.59%
Heuristic-CALA (Ours)	**54.13%**	**58.38%**	**51.88%**	**52.75%**	**44.71%**	**50.82%**	**79.14%**	**82.04%**	**56.67%**	**57.78%**	**69.04%**	**75.51%**
MetaSAug (Li et al. 2021)	50.53%	55.21%	49.12%	48.56%	41.27%	47.38%	77.89%	79.45%	54.87%	54.78%	67.83%	73.05%
Meta-CALA (Ours)	**55.14%**	**59.47%**	**52.76%**	**53.66%**	**45.83%**	**51.36%**	**80.05%**	**82.98%**	**58.99%**	**59.56%**	**71.05%**	**76.21%**

Table 4: Comparison of accuracy and precision of the CLT, GLT, and ALT protocols on ImageNet-GLT and MSCOCO-GLT.

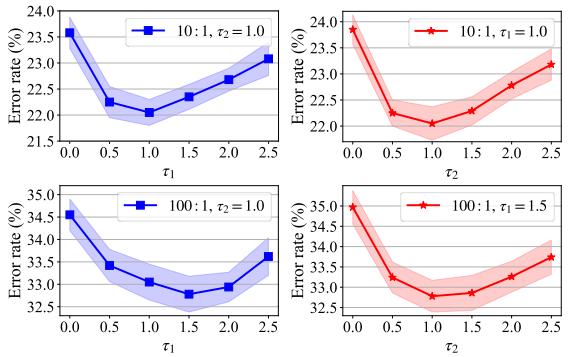

Figure 5: Influence of different τ_1 and τ_2 values on CIFAR-LT with imbalance ratios of 10:1 and 100:1.

Experiments for Class & Attribute Imbalance

We consider two GLT benchmarks, ImageNet-GLT and MSCOCO-GLT (Tang et al. 2022). Each benchmark consists of three protocols: CLT, ALT, and GLT, involving changes in the class, attribute, and both class and attribute distributions from training to testing. The experimental settings follow those in Tang et al. (2022), utilizing ResNeXt-50 (Xie et al. 2017) as the backbone network. For Meta-CALA, we optimize the adjustment network using Adam with an initial learning rate of 1×10^{-3}. To construct the metadata, we randomly select two samples per group and class from the training data. Additionally, we set τ_1 and τ_2 to 1.5 and 1 for the CLT protocol. Both τ_1 and τ_2 are set to 1 for the ALT and GLT protocols. We report both accuracy and precision to provide a comprehensive evaluation.

Results. The comparison results for ImageNet-GLT and MSCOCO-GLT are presented in Table 4, with some results sourced from the IFL (Tang et al. 2022) paper. As observed, there is a significant performance decline from the CLT protocol to the GLT protocol, highlighting the challenge posed by attribute imbalance. Heuristic-CALA, which incorporates two adjustment terms to address both class and attribute imbalances, achieves substantial improvements over other methods across all three protocols. Furthermore, Meta-CALA attains SOTA performance by leveraging metadata information to adjust the model during training.

Setting	ImageNet-GLT		MSCOCO-GLT	
	Acc.	Prec.	Acc.	Prec.
Heuristic-CALA	**44.71%**	**50.82%**	**69.04%**	**75.51%**
w/o $u(y)$	42.51%	46.28%	67.49%	72.34%
w/o $v(\boldsymbol{x}, y)$	37.80%	37.56%	66.17%	68.35%

Table 5: Accuracy and precision on the GLT protocol of the ImageNet-GLT and MSCOCO-GLT benchmarks.

Sensitivity and Ablation Studies

We perform sensitivity analyses on τ_1 and τ_2, which govern the influence of the two adjustment terms. The results for Heuristic-CALA are shown in Fig. 5. $\tau_2 = 1$ yields the best performance across different imbalance ratios. For τ_1, the optimal value is 1 under a 10:1 ratio, while a value of 1.5 is optimal under a 100:1 ratio. These findings suggest that as the class imbalance becomes more pronounced, a larger τ_1 is preferable. Additionally, we perform ablation studies on the CALA loss, considering two settings that remove $u(y)$ and $v(\boldsymbol{x}, y)$ separately. The results, presented in Table 5, indicate that both terms are necessary and crucial for addressing class and attribute imbalances. Furthermore, the role of the CCPD ratio $v(\boldsymbol{x}, y)$ is generally more significant than that of the class priors $u(y)$ under GLT learning.

Conclusion

This study underscores the importance of directly estimating the CCPD ratio between the training and test data in addressing GLT learning. We first introduce a novel logit-adjusted loss function, termed CALA, which incorporates both class priors and the CCPD ratio as adjustment terms. Subsequently, we propose two methods for estimating the CCPD ratio in the CALA loss: a K-neighborhood-based approach and a meta-learning-based approach. These methods give rise to two logit adjustment techniques, Heuristic-CALA and Meta-CALA. Extensive experiments validate the efficacy of our methodologies in addressing both class- and attribute-wise imbalances, achieving SOTA performance across various learning scenarios.

Acknowledgments

This work was mainly conducted by the authors during their tenure at Tianjin University and was partially supported by the NSFC under Grants 6207617 and 62476191.

References

Agarwal, V.; Shetty, R.; and Fritz, M. 2020. Towards causal vqa: Revealing and reducing spurious correlations by invariant and covariant semantic editing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 9687–9695.

Ahuja, K.; Caballero, E.; Zhang, D.; Gagnon-Audet, J.-C.; Bengio, Y.; Mitliagkas, I.; and Rish, I. 2021. Invariance principle meets information bottleneck for out-of-distribution generalization. In *Proceedings of the Advances in Neural Information Processing Systems*, 3438–3450.

Arjovsky, M.; Bottou, L.; Gulrajani, I.; and Lopez-Paz, D. 2019. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*.

Cao, K.; Wei, C.; Gaidon, A.; Arechiga, N.; and Ma, T. 2019. Learning imbalanced datasets with label-distribution-aware margin loss. In *Proceedings of the Advances in Neural Information Processing Systems*, 1567–1578.

Chen, A. S.; Lee, Y.; Setlur, A.; Levine, S.; and Finn, C. 2023. Confidence-based model selection: When to take shortcuts for subpopulation shifts. *arXiv preprint arXiv:2306.11120*.

Chen, X.; Zhou, Y.; Wu, D.; Zhang, W.; Zhou, Y.; Li, B.; and Wang, W. 2022. Imagine by reasoning: A reasoning-based implicit semantic data augmentation for long-tailed classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 356–364.

Cubuk, E. D.; Zoph, B.; Shlens, J.; and Le, Q. V. 2020. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 3008–3017.

Cui, J.; Zhong, Z.; Liu, S.; Yu, B.; and Jia, J. 2021. Parametric contrastive learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 715–724.

Cui, Y.; Jia, M.; Lin, T.-Y.; Song, Y.; and Belongie, S. 2019. Class-balanced loss based on effective number of samples. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 9260–9269.

De Alvis, C.; and Seneviratne, S. 2024. A survey of deep long-tail classification advancements. *arXiv preprint arXiv:2404.15593*.

Deng, Y.; Yang, Y.; Mirzasoleiman, B.; and Gu, Q. 2024. Robust learning with progressive data expansion against spurious correlation. In *Proceedings of the Advances in Neural Information Processing Systems*, 1390–1402.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 770–778.

Hong, Y.; Han, S.; Choi, K.; Seo, S.; Kim, B.; and Chang, B. 2021. Disentangling label distribution for long-tailed visual recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 6626–6636.

Hong, Y.; Zhang, J.; Sun, Z.; and Yan, K. 2022. Safa: Sample-adaptive feature augmentation for long-tailed image classification. In *Proceedings of the European Conference on Computer Vision*, 587–603.

Kang, B.; Xie, S.; Rohrbach, M.; Yan, Z.; Gordo, A.; Feng, J.; and Kalantidis, Y. 2020. Decoupling representation and classifier for long-tailed recognition. In *Proceedings of the International Conference on Learning Representations*.

Koh, P. W.; Sagawa, S.; Marklund, H.; Xie, S. M.; Zhang, M.; Balsubramani, A.; Hu, W.; Yasunaga, M.; Phillips, R. L.; Gao, I.; Lee, T.; David, E.; Stavness, I.; Guo, W.; Earnshaw, B.; Haque, I.; Beery, S. M.; Leskovec, J.; Kundaje, A.; Pierson, E.; Levine, S.; Finn, C.; and Liang, P. 2021. Wilds: A benchmark of in-the-wild distribution shifts. In *Proceedings of the International Conference on Machine Learning*, 5637–5664.

Krueger, D.; Caballero, E.; Jacobsen, J.-H.; Zhang, A.; Binas, J.; Zhang, D.; Le Priol, R.; and Courville, A. 2021. Out-of-distribution generalization via risk extrapolation (rex). In *Proceedings of the International Conference on Machine Learning*, 5815–5826.

Li, M.; Cheung, Y.-m.; and Lu, Y. 2022. Long-tailed visual recognition via gaussian clouded logit adjustment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6929–6938.

Li, S.; Gong, K.; Liu, C. H.; Wang, Y.; Qiao, F.; and Cheng, X. 2021. Metasaug: Meta semantic augmentation for long-tailed visual recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5212–5221.

Liang, W.; and Zou, J. 2022. Metashift: A dataset of datasets for evaluating contextual distribution shifts and training conflicts. In *Proceedings of the International Conference on Learning Representations*.

Lin, C.; Tsai, C.-F.; and Lin, W.-C. 2023. Towards hybrid over- and under-sampling combination methods for class imbalanced datasets: an experimental study. *Artificial Intelligence Review*, 56(2): 845–863.

Liu, Z.; Luo, P.; Wang, X.; and Tang, X. 2015. Deep learning face attributes in the wild. In *Proceedings of the IEEE International Conference on Computer Vision*, 3730–3738.

Liu, Z.; Miao, Z.; Zhan, X.; Wang, J.; Gong, B.; and Yu, S. X. 2019. Large-scale long-tailed recognition in an open world. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2537–2546.

Loshchilov, I.; and Hutter, F. 2016. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*.

Mao, R.; Fan, W.; and Li, Q. 2023. GCARe: Mitigating subgroup unfairness in graph condensation through adversarial regularization. *Applied Sciences*, 13(16): 9166.

Menon, A. K.; Jayasumana, S.; Rawat, A. S.; Jain, H.; Veit, A.; and Kumar, S. 2021. Long-tail learning via logit adjustment. In *Proceedings of the International Conference on Learning Representations*.

Ren, J.; Yu, C.; Sheng, S.; Ma, X.; Zhao, H.; Yi, S.; and Li, H. 2020. Balanced meta-softmax for long-tailed visual recognition. In *Proceedings of the Advances in Neural Information Processing Systems*, 4175–4186.

Sagawa, S.; Koh, P. W.; Hashimoto, T. B.; and Liang, P. 2020. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. In *Proceedings of the International Conference on Learning Representations*.

Shi, J.-X.; Wei, T.; Xiang, Y.; and Li, Y.-F. 2023. How re-sampling helps for long-tail learning? In *Proceedings of the Advances in Neural Information Processing Systems*, 75669–75687.

Shi, Y.; Seely, J.; Torr, P. H.; Siddharth, N.; Hannun, A.; Usunier, N.; and Synnaeve, G. 2022. Gradient matching for domain generalization. In *Proceedings of the International Conference on Learning Representations*.

Shu, J.; Xie, Q.; Yi, L.; Zhao, Q.; Zhou, S.; Xu, Z.; and Meng, D. 2019. Meta-weight-net: Learning an explicit mapping for sample weighting. In *Proceedings of the Advances in Neural Information Processing Systems*, 1919–1930.

Srivastava, M.; Hashimoto, T.; and Liang, P. 2020. Robustness to spurious correlations via human annotations. In *Proceedings of the International Conference on Machine Learning*, 9046–9056.

Sun, B.; and Saenko, K. 2016. Deep coral: Correlation alignment for deep domain adaptation. In *Proceedings of the European Conference on Computer Vision Workshops*, 443–450.

Tang, K.; Tao, M.; Qi, J.; Liu, Z.; and Zhang, H. 2022. Invariant feature learning for generalized long-tailed classification. In *Proceedings of the European Conference on Computer Vision*, 709–726.

Tao, Y.; Sun, J.; Yang, H.; Chen, L.; Wang, X.; Yang, W.; Du, D.; and Zheng, M. 2023. Local and global logit adjustments for long-tailed learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 11783–11792.

Tripathi, A.; Chakraborty, R.; and Kopparapu, S. K. 2021. A novel adaptive minority oversampling technique for improved classification in data imbalanced scenarios. In *Proceedings of the International Conference on Pattern Recognition*, 10650–10657.

Wan, M.; Zha, D.; Liu, N.; and Zou, N. 2023. In-processing modeling techniques for machine learning fairness: A survey. *ACM Transactions on Knowledge Discovery from Data*, 17(3): 1–27.

Wang, X.; Lian, L.; Miao, Z.; Liu, Z.; and Yu, S. X. 2020. Long-tailed recognition by routing diverse distribution-aware experts. *arXiv preprint arXiv:2010.01809*.

Wang, Z.; Xu, Q.; Yang, Z.; He, Y.; Cao, X.; and Huang, Q. 2024. A unified generalization analysis of re-weighting and logit-adjustment for imbalanced learning. In *Proceedings of the Advances in Neural Information Processing Systems*, 48417–48430.

Xiang, L.; Ding, G.; and Han, J. 2020. Learning from multiple experts: Self-paced knowledge distillation for long-tailed classification. In *Proceedings of the European Conference on Computer Vision*, 247–263.

Xie, S.; Girshick, R.; Dollár, P.; Tu, Z.; and He, K. 2017. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1492–1500.

Xu, M.; Zhang, J.; Ni, B.; Li, T.; Wang, C.; Tian, Q.; and Zhang, W. 2020. Adversarial domain adaptation with domain mixup. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 6502–6509.

Yan, Y. T.; Wu, Z. B.; Du, X. Q.; Chen, J.; Zhao, S.; and Zhang, Y. P. 2019. A three-way decision ensemble method for imbalanced data oversampling. *International Journal of Approximate Reasoning*, 107: 1–16.

Yao, H.; Wang, Y.; Li, S.; Zhang, L.; Liang, W.; Zou, J.; and Finn, C. 2022. Improving out-of-distribution robustness via selective augmentation. In *Proceedings of the International Conference on Machine Learning*, 25407–25437.

Zhang, H.; Cisse, M.; Dauphin, Y. N.; and Lopez-Paz, D. 2018. Mixup: Beyond empirical risk minimization. In *Proceedings of the International Conference on Learning Representations*.

Zhang, S.; Chen, C.; Hu, X.; and Peng, S. 2023. Balanced knowledge distillation for long-tailed learning. *Neurocomputing*, 527: 36–46.

Zhang, S.; Li, Z.; Yan, S.; He, X.; and Sun, J. 2021. Distribution alignment: A unified framework for long-tail visual recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2361–2370.

Zhao, Y.; Chen, W.; Tan, X.; Huang, K.; and Zhu, J. 2022. Adaptive logit adjustment loss for long-tailed visual recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 3472–3480.

Zhong, Z.; Cui, J.; Liu, S.; and Jia, J. 2021. Improving calibration for long-tailed recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 16489–16498.

Zhou, B.; Cui, Q.; Wei, X.; and Chen, Z. 2020. Bbn: Bilateral-branch network with cumulative learning for long-tailed visual recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 9719–9728.

Zhou, X.; and Wu, O. 2023. Implicit counterfactual data augmentation for deep neural networks. *arXiv preprint arXiv:2304.13431*.

Zhou, X.; Yang, N.; and Wu, O. 2023. Combining adversaries with anti-adversaries in training. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 11435–11442.

Zhou, X.; Ye, W.; Lee, Z.; Xie, R.; and Zhang, S. 2024. Boosting model resilience via implicit adversarial data augmentation. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 5653–5661.

Appendix for *Class and Attribute-Aware Logit Adjustment for Generalized Long-Tail Learning*

Xiaoling Zhou[1,2], **Ou Wu**[1,3]*, **and Nan Yang**[1]
[1]Center for Applied Mathematics, Tianjin University, China
[2] National Engineering Research Center for Software Engineering, Peking University, China
[3] HIAS, University of Chinese Academy of Sciences, Hangzhou, China
{xiaolingzhou, yny, wuou}@tju.edu.cn

A Regularization Analysis of CALA Loss

We elucidate the significance of the CALA loss from a regularization perspective. The loss function, which adjusts the logits, can be expressed as follows:

$$\ell\left(\boldsymbol{a}_i, y_i\right)=-\log \frac{\exp\left(a_{i,y_i}+\Delta a_{i,y_i}\right)}{\sum_{c\in[C]}\exp\left(a_{i,c}+\Delta a_{i,c}\right)}, \tag{A.1}$$

where $\boldsymbol{a}_i = f(\boldsymbol{x}_i)$ represents the logit vector associated with sample $\boldsymbol{x}_i$. $\mathcal{C}$ denotes the number of classes. The CALA loss for sample $\boldsymbol{x}_i$ is defined as follows:

$$\ell_{CALA}(\boldsymbol{a}_i, y_i)=-\log \frac{\exp[a_{i,y_i}+\tau_1\log p_{\mathrm{tr}}(y=y_i)+\tau_2\log\frac{p_{\mathrm{tr}}(\boldsymbol{x}|y=y_i)}{p_{\mathrm{te}}(\boldsymbol{x}|y=y_i)}]}{\sum_{c\in[\mathcal{C}]}\exp[a_{i,c}+\tau_1\log p_{\mathrm{tr}}(y=c)+\tau_2\log\frac{p_{\mathrm{tr}}(\boldsymbol{x}|y=c)}{p_{\mathrm{te}}(\boldsymbol{x}|y=c)}]}. \tag{A.2}$$

Accordingly, the logit adjustment term for category c is $\Delta a_{i,c}=\tau_1\log p_{\mathrm{tr}}(y=c)+\tau_2\log\frac{p_{\mathrm{tr}}(\boldsymbol{x}|y=c)}{p_{\mathrm{te}}(\boldsymbol{x}|y=c)}$. Utilizing the first-order Taylor expansion, we can obtain the following expression for the logit-adjusted Cross-Entropy (CE) loss:

$$\ell(\boldsymbol{a}+\Delta\boldsymbol{a})\approx\ell(\boldsymbol{a})+(\frac{\partial\ell}{\partial\boldsymbol{a}})^T\Delta\boldsymbol{a}=\ell(\boldsymbol{a})+(\boldsymbol{q}-\boldsymbol{y})^T\Delta\boldsymbol{a}, \tag{A.3}$$

where $\boldsymbol{q}=\mathrm{Softmax}(\boldsymbol{a})$ represents the probability vector. Consequently, the regularization term for the logit-adjusted loss can be defined as $R=(\boldsymbol{q}-\boldsymbol{y})^T\Delta\boldsymbol{a}$. Next, we derive the regularization term of the CALA loss using Eq. (A.3). The derivation process is outlined as follows:

$$\begin{aligned}
&\ell_{CALA}(\boldsymbol{a}_i+\Delta\boldsymbol{a}_i)\\
&\quad\approx\ell_{CE}(\boldsymbol{a}_i)+(\frac{\partial\ell}{\partial\boldsymbol{a}_i})^T\Delta\boldsymbol{a}_i\\
&\quad=\ell_{CE}(\boldsymbol{a}_i)+(\boldsymbol{q}_i-\boldsymbol{y})^T\Delta\boldsymbol{a}_i\\
&\quad=\ell_{CE}(\boldsymbol{a}_i)+[q_{i,1},\cdots,q_{i,y_i}-1,\cdots,q_{i,\mathcal{C}}]\times\begin{bmatrix}\tau_1\log\frac{p_{\mathrm{tr}}(y=1)}{p_{\mathrm{tr}}(y=y_i)}+\tau_2\log\frac{p_{\mathrm{tr}}(\boldsymbol{x}|y=1)/p_{\mathrm{te}}(\boldsymbol{x}|y=1)}{p_{\mathrm{tr}}(\boldsymbol{x}|y=y_i)/p_{\mathrm{te}}(\boldsymbol{x}|y=y_i)}\\ \vdots\\ 0\\ \vdots\\ \tau_1\log\frac{p_{\mathrm{tr}}(y=\mathcal{C})}{p_{\mathrm{tr}}(y=y_i)}+\tau_2\log\frac{p_{\mathrm{tr}}(\boldsymbol{x}|y=\mathcal{C})/p_{\mathrm{te}}(\boldsymbol{x}|y=\mathcal{C})}{p_{\mathrm{tr}}(\boldsymbol{x}|y=y_i)/p_{\mathrm{te}}(\boldsymbol{x}|y=y_i)}\end{bmatrix}\\
&\quad=\ell_{CE}(\boldsymbol{a}_i)+\sum_{c\neq y_i}q_{i,c}[\tau_1\log\frac{p_{\mathrm{tr}}(y=c)}{p_{\mathrm{tr}}(y=y_i)}+\tau_2\log\frac{p_{\mathrm{tr}}(\boldsymbol{x}|y=c)/p_{\mathrm{te}}(\boldsymbol{x}|y=c)}{p_{\mathrm{tr}}(\boldsymbol{x}|y=y_i)/p_{\mathrm{te}}(\boldsymbol{x}|y=y_i)}].
\end{aligned} \tag{A.4}$$

*Corresponding author.

Therefore, we can conclude that the regularization term for all samples in CALA loss is given by

$$R_{CALA} = \sum_{i=1}^{N} \sum_{c \neq y_i} q_{i,c} [\tau_1 \log \frac{p_{\text{tr}}(y=c)}{p_{\text{tr}}(y=y_i)} + \tau_2 \log \frac{p_{\text{tr}}(\boldsymbol{x}|y=c)/p_{\text{te}}(\boldsymbol{x}|y=c)}{p_{\text{tr}}(\boldsymbol{x}|y=y_i)/p_{\text{te}}(\boldsymbol{x}|y=y_i)}]. \tag{A.5}$$

As we can see, this regularizer applies large penalties to samples belonging to tail classes, which are characterized by high values of $p_{\text{tr}}(y=c)/p_{\text{tr}}(y=y_i)$, as well as samples with rare attributes within their respective class in the training data, as indicated by high values of $\frac{p_{\text{tr}}(\boldsymbol{x}|y=c)/p_{\text{te}}(\boldsymbol{x}|y=c)}{p_{\text{tr}}(\boldsymbol{x}|y=y_i)/p_{\text{te}}(\boldsymbol{x}|y=y_i)}$. These penalties enhance the impact of these samples on model training, thereby improving their prediction performance.

B More Details of Heuristic-CALA

B.I Detailed Derivation Process

We first present the derivation of Eq. (6) in the main text. Let V_ϵ denote the volume of the neighborhood $\mathcal{N}_\epsilon(\boldsymbol{x})$. For the class-conditional probability density, we have the following approximation:

$$\begin{aligned} p(\boldsymbol{x}|y) &\approx \frac{p(\boldsymbol{x}' \in \mathcal{N}_\epsilon(\boldsymbol{x})|y)}{V_\epsilon} \\ &= \frac{p(\boldsymbol{x}' \in \mathcal{N}_\epsilon(\boldsymbol{x}), y)}{p(y)V_\epsilon} \\ &= \frac{p(y|\boldsymbol{x}' \in \mathcal{N}_\epsilon(\boldsymbol{x}))p(\boldsymbol{x}' \in \mathcal{N}_\epsilon(\boldsymbol{x}))}{p(y)V_\epsilon} \\ &\approx \frac{p(y|\boldsymbol{x}' \in \mathcal{N}_\epsilon(\boldsymbol{x}))p(\boldsymbol{x})V_\epsilon}{p(y)V_\epsilon} \\ &= \frac{p(y|\boldsymbol{x}' \in \mathcal{N}_\epsilon(\boldsymbol{x}))p(\boldsymbol{x})}{p(y)}. \end{aligned} \tag{A.6}$$

Thus, the following formula holds:

$$p(\boldsymbol{x}|y) \approx \frac{p(\boldsymbol{x})}{p(y)} \cdot p(y|\boldsymbol{x}' \in \mathcal{N}_\epsilon(\boldsymbol{x})). \tag{A.7}$$

Then, we have

$$\frac{p_{\text{tr}}(\boldsymbol{x}|y)}{p_{\text{te}}(\boldsymbol{x}|y)} \approx \frac{p_{\text{tr}}(\boldsymbol{x})}{p_{\text{te}}(\boldsymbol{x})} \cdot \frac{p_{\text{tr}}(y|\boldsymbol{x}' \in \mathcal{N}_\epsilon(\boldsymbol{x}))}{p_{\text{te}}(y|\boldsymbol{x}' \in \mathcal{N}_\epsilon(\boldsymbol{x}))} \cdot \frac{p_{\text{te}}(y)}{p_{\text{tr}}(y)}. \tag{A.8}$$

Incorporating Eq. (A.8) into the following formula

$$\frac{p_{\text{tr}}(y|\boldsymbol{x})}{p_{\text{te}}(y|\boldsymbol{x})} = \frac{p_{\text{tr}}(\boldsymbol{x}|y)}{p_{\text{te}}(\boldsymbol{x}|y)} \cdot \frac{p_{\text{tr}}(y)}{p_{\text{te}}(y)} \cdot \frac{p_{\text{te}}(\boldsymbol{x})}{p_{\text{tr}}(\boldsymbol{x})}, \tag{A.9}$$

we yield

$$p_{\text{te}}(y|\boldsymbol{x}) \approx p_{\text{tr}}(y|\boldsymbol{x}) \cdot \frac{p_{\text{te}}(y|\boldsymbol{x}' \in \mathcal{N}_\epsilon(\boldsymbol{x}))}{p_{\text{tr}}(y|\boldsymbol{x}' \in \mathcal{N}_\epsilon(\boldsymbol{x}))}. \tag{A.10}$$

From the above relation, we know that $p_{\text{te}}(y|\boldsymbol{x}' \in \mathcal{N}_\epsilon(\boldsymbol{x}))/p_{\text{tr}}(y|\boldsymbol{x}' \in \mathcal{N}_\epsilon(\boldsymbol{x}))$ should be utilized to adjust the logits of samples. However, ensuring the same neighborhood sizes for all samples in practice is a challenging task due to variations in sample characteristics. Specifically, the distances between each sample and its surrounding samples vary. To this end, we employ K-nearest neighbors $\mathcal{N}_K(\boldsymbol{x})$ to replace $\mathcal{N}_\epsilon(\boldsymbol{x})$, while maintaining the underlying essence of $p_{\text{te}}(y|\boldsymbol{x}' \in \mathcal{N}_\epsilon(\boldsymbol{x}))/p_{\text{tr}}(y|\boldsymbol{x}' \in \mathcal{N}_\epsilon(\boldsymbol{x}))$.

B.II Algorithmic Details

The complete algorithm for Heuristic-CALA is delineated in Algorithm A-1. The value of the hyperparameter in MixUp can be directly set as 0.2, as recommended by the original paper (Zhang et al., 2018). During implementation, we employ the cosine distance between the deep features of training samples to compute the neighborhood of each sample. The reason for choosing cosine distance is its ability to be invariant to the feature space's dimensionality and its relatively low computational complexity. Moreover,

considering that the neighborhood information of samples may be susceptible to the border effect and outliers, we utilize the class averages of the corresponding values in the neighborhood to smooth the values of $p_{\mathrm{mx}}(y=c|\boldsymbol{x}' \in \mathcal{N}_K(\boldsymbol{x}))$ and $p_{\mathrm{pr}}(y=c|\boldsymbol{x}' \in \mathcal{N}_K(\boldsymbol{x}))$. Denote $\mathcal{N}_{K,c}(\boldsymbol{x})$ contains the samples with label c in $\mathcal{N}_K(\boldsymbol{x})$. For $p_{\mathrm{pr}}(y=c|\boldsymbol{x}' \in \mathcal{N}_K(\boldsymbol{x}))$, we use the following formula to smooth it:

$$\tilde{p}_{\mathrm{pr}}(y=c|\boldsymbol{x}' \in \mathcal{N}_K(\boldsymbol{x})) = \alpha \overline{p}_{\mathrm{pr}}(\boldsymbol{x}' \in \mathcal{N}_{K,c}(\boldsymbol{x})) + (1-\alpha) p_{\mathrm{pr}}(y=c|\boldsymbol{x}' \in \mathcal{N}_K(\boldsymbol{x})), \tag{A.11}$$

where $\overline{p}_{\mathrm{pr}}(\boldsymbol{x}' \in \mathcal{N}_{K,c}(\boldsymbol{x}))$ represents the average value of $p_{\mathrm{pr}}(y=c|\boldsymbol{x}' \in \mathcal{N}_K(\boldsymbol{x}))$ for all samples in $\mathcal{N}_{K,c}(\boldsymbol{x})$ and α is the smooth factor, which is fixed as 0.05 in our experiments. The value of $p_{\mathrm{mx}}(y=c|\boldsymbol{x}' \in \mathcal{N}_K(\boldsymbol{x}))$ is smoothed in the same way and $\tilde{p}_{\mathrm{mx}}(y=c|\boldsymbol{x}' \in \mathcal{N}_K(\boldsymbol{x}))$ can be obtained. Finally, the term $p_{\mathrm{pr}}(y=c|\boldsymbol{x}' \in \mathcal{N}_K(\boldsymbol{x}))/p_{\mathrm{mx}}(y=c|\boldsymbol{x}' \in \mathcal{N}_K(\boldsymbol{x}))$ is replaced by $\tilde{p}_{\mathrm{pr}}(y=c|\boldsymbol{x}' \in \mathcal{N}_K(\boldsymbol{x}))/\tilde{p}_{\mathrm{mx}}(y=c|\boldsymbol{x}' \in \mathcal{N}_K(\boldsymbol{x}))$. When $\alpha=1$, the adjustment terms for samples in a neighborhood are at the category level.

C More Details of Meta-CALA

C.I Extracted Training Characteristics

As stated in the main text, a total number of 42 characteristics are extracted from the classifier and input into the adjustment network to estimate $p_{\mathrm{tr}}(\boldsymbol{x}|y)/p_{\mathrm{te}}(\boldsymbol{x}|y)$. Denote the loss of sample $\boldsymbol{x}_i$ as ℓ_i; the logit vector is $\boldsymbol{a}_i = f(\boldsymbol{x}_i)$; and the probability vector is $\boldsymbol{q}_i = \mathrm{Softmax}(\boldsymbol{a}_i)$. The deep feature of sample $\boldsymbol{x}_i$ is denoted as $\boldsymbol{h}_i$, which is output by the previous layer of the logit. The average feature for all samples in the neighborhood of $\mathcal{N}_K(\boldsymbol{x}_i)$ is denoted as $\overline{\boldsymbol{h}}_{\mathcal{N}_K(\boldsymbol{x}_i)}$. Detailed calculations and descriptions of the extracted characteristics are presented in Table A-1. Moreover, the 21 characteristics presented in Table A-1 can be further expanded through the sequence, thereby obtaining $\zeta_{i,22}^t$ to $\zeta_{i,42}^t$. Specifically, we examine the differences in the characteristics between the current and previous iterations. For instance, for ℓ_i^t, its sequence-extended feature is denoted as $\zeta_{i,22}^t = \ell_i^t - \ell_i^{t-1}$.

C.II Algorithmic Details

We further elaborate on the optimization process of Meta-CALA, presenting more comprehensive formulas. First, a batch of training samples $\{\boldsymbol{x}_i, y_i\}_{i=1}^n$ is selected, where n is the batch size and the updating of $\boldsymbol{W}$ can be formulated as

$$\hat{\boldsymbol{W}}^{(t)} \leftarrow \boldsymbol{W}^{(t)} - \eta_1 \frac{1}{n} \sum\nolimits_{i=1}^{n} \nabla_{\boldsymbol{W}} \ell_{CALA}\left(f(\boldsymbol{x}_i), y_i; \boldsymbol{\delta}_i(\boldsymbol{\zeta}_i^{(t)}, \boldsymbol{\Omega}^{(t)})\right), \tag{A.12}$$

where η_1 is the step size; $\boldsymbol{\delta}_i$ and $\boldsymbol{\zeta}_i$ refer to the adjustment vector and the training characteristics of $\boldsymbol{x}_i$. After extracting the characteristics from the classifier, the parameters of the adjustment network $\boldsymbol{\Omega}$ can be updated on a minibatch of metadata $\{\boldsymbol{x}_i^{\mathrm{meta}}, y_i^{\mathrm{meta}}\}_{i=1}^m$, with the following formula:

$$\boldsymbol{\Omega}^{(t+1)} \leftarrow \boldsymbol{\Omega}^{(t)} - \eta_2 \frac{1}{m} \sum\nolimits_{i=1}^{m} \nabla_{\boldsymbol{\Omega}} \ell_{CE}\left(f_{\hat{\boldsymbol{W}}}(\boldsymbol{x}_i^{\mathrm{meta}}), y_i^{\mathrm{meta}}\right), \tag{A.13}$$

where m and η_2 are the minibatch size of metadata and the step size, respectively. Subsequently, the parameters of the classifier network are updated using the resulting adjustment terms

$$\boldsymbol{W}^{(t+1)} \leftarrow \boldsymbol{W}^{(t)} - \eta_1 \frac{1}{n} \sum\nolimits_{i=1}^{n} \nabla_{\boldsymbol{W}} \ell_{CALA}\left(f(\boldsymbol{x}_i), y_i; \boldsymbol{\delta}_i(\boldsymbol{\zeta}_i^{(t+1)}, \boldsymbol{\Omega}^{(t+1)})\right). \tag{A.14}$$

The algorithm of Meta-CALA is presented in Algorithm A-2.

Algorithm A-1 Algorithm of Heuristic-CALA

Input: Training data $\mathcal{D}^{\mathrm{tr}} = \{\boldsymbol{x}_i, y_i\}_{i=1}^N$, neighborhood size K, tuning parameters τ_1 and τ_2, #iteration $\mathcal{T}$, batch size n, and other training parameters;

Output: Trained classifier $f_{\boldsymbol{W}}$;

1: Initialize the classifier $f_{\boldsymbol{W}}$;
2: **for** $t = 1$ to $\mathcal{T}$ **do**
3: Sample a batch of data $\{\boldsymbol{x}_i, y_i\}_{i=1}^n$ from $\mathcal{D}^{\mathrm{tr}}$;
4: Augment $\{\boldsymbol{x}_i, y_i\}_{i=1}^n$ by MixUp;
5: Calculate the adjustment terms $u(y)$ and $v(\boldsymbol{x}, y)$ for all classes across the samples in the mini-batch;
6: Update $f_{\boldsymbol{W}}$ using ℓ_{CALA} with the SGD optimizer;
7: **end for**

Quantity	Formula	Description
$\zeta_{i,1}^t$	ℓ_i^t	Loss
$\zeta_{i,2}^t$	a_{i,y_i}^t	The logit vector corresponding to the ground-truth class
$\zeta_{i,3}^t$	$g_i^t = \|\boldsymbol{y}_i - \boldsymbol{q}_i^t\|$	The norm of loss gradient with respect to the logit vector
$\zeta_{i,4}^t$	q_{i,y_i}^t	The predicted probability corresponding to the ground-truth class
$\zeta_{i,5}^t$	$z_i^t = -\sum_{c\in[\mathcal{C}]} q_{i,c}^t \log(q_{i,c}^t)$	Uncertainty
$\zeta_{i,6}^t$	$\gamma_i^t = q_{i,y_i}^t - \max_{c\neq y_i} q_{i,c}^t$	Margin
$\zeta_{i,7}^t$	$\overline{\ell}_i^t = \frac{\sum_{\boldsymbol{x}_j\in\mathcal{N}_K(\boldsymbol{x}_i)} \ell_j^t}{K}$	The average loss for samples in $\mathcal{N}_K(\boldsymbol{x}_i)$
$\zeta_{i,8}^t$	$\overline{a}_i^t = \frac{\sum_{\boldsymbol{x}_j\in\mathcal{N}_K(\boldsymbol{x}_i)} a_{j,y_j}^t}{K}$	The average logit corresponding to the ground-truth class for samples in $\mathcal{N}_K(\boldsymbol{x}_i)$
$\zeta_{i,9}^t$	$\overline{g}_i^t = \frac{\sum_{\boldsymbol{x}_j\in\mathcal{N}_K(\boldsymbol{x}_i)} g_j^t}{K}$	The average loss gradient with respect to the logit vector for samples in $\mathcal{N}_K(\boldsymbol{x}_i)$
$\zeta_{i,10}^t$	$\overline{q}_i^t = \frac{\sum_{\boldsymbol{x}_j\in\mathcal{N}_K(\boldsymbol{x}_i)} q_{j,y_j}^t}{K}$	The average predicted probability corresponding to the ground-truth class for samples in $\mathcal{N}_K(\boldsymbol{x}_i)$
$\zeta_{i,11}^t$	$\overline{z}_i^t = \frac{\sum_{\boldsymbol{x}_j\in\mathcal{N}_K(\boldsymbol{x}_i)} z_j^t}{K}$	The average uncertainty for samples in $\mathcal{N}_K(\boldsymbol{x}_i)$
$\zeta_{i,12}^t$	$\overline{\gamma}_i^t = \frac{\sum_{\boldsymbol{x}_j\in\mathcal{N}_K(\boldsymbol{x}_i)} \gamma_j^t}{K}$	The average margin for samples in $\mathcal{N}_K(\boldsymbol{x}_i)$
$\zeta_{i,13}^t$	$\ell_i^t - \overline{\ell}_i^t$	The difference between ℓ_i^t and $\overline{\ell}_i^t$
$\zeta_{i,14}^t$	$a_{i,y_i}^t - \overline{a}_i^t$	The difference between a_{i,y_i}^t and $\overline{a}_i^t$
$\zeta_{i,15}^t$	$g_i^t - \overline{g}_i^t$	The difference between g_i^t and $\overline{g}_i^t$
$\zeta_{i,16}^t$	$q_{i,y_i}^t - \overline{q}_i^t$	The difference between q_{i,y_i}^t and $\overline{q}_i^t$
$\zeta_{i,17}^t$	$z_i^t - \overline{z}_i^t$	The difference between z_i^t and $\overline{z}_i^t$
$\zeta_{i,18}^t$	$\gamma_i^t - \overline{\gamma}_i^t$	The difference between γ_i^t and $\overline{\gamma}_i^t$
$\zeta_{i,19}^t$	$\|\mathcal{N}_{K,y_i}(\boldsymbol{x}_i)\|/K$	The ratio of samples sharing the same label with $\boldsymbol{x}_i$ in the neighborhood $\mathcal{N}_K(\boldsymbol{x}_i)$
$\zeta_{i,20}^t$	$\max_{c\neq y_i} \|\mathcal{N}_{K,c}(\boldsymbol{x}_i)\|/K$	The ratio of heterogeneous samples with the highest proportion in the neighborhood $\mathcal{N}_K(\boldsymbol{x}_i)$
$\zeta_{i,21}^t$	$\cos(\boldsymbol{h}_i, \overline{\boldsymbol{h}}_{\mathcal{N}_K(\boldsymbol{x}_i)})$	The cosine distance between the deep feature of $\boldsymbol{x}_i$ and the average feature of samples in the neighborhood $\mathcal{N}_K(\boldsymbol{x}_i)$

Table A-1: Formulas and descriptions of extracted training characteristics.

D More Details of Experimental Investigation

D.I Introduction of Utilized Datasets

First, we utilize three long-tail (LT) benchmarks: CIFAR-LT, iNaturalist 2018, and Places-LT, which are detailed as follows:

CIFAR-LT is the long-tailed version of CIFAR (Krizhevsky and Hinton, 2009) dataset. The original CIFAR10 (CIFAR100) dataset consists of 50,000 images drawn from 10 (100) classes with even data distribution. In other words, CIFAR10 (CIFAR100) has 5,000 (500) images per class. Following Cui et al. (2019), we discard some training samples to construct imbalanced datasets. Two training sets with imbalance ratios of 10:1 and 100:1 are compiled. As for test sets, we use the original balanced test sets.

iNaturalist (iNat) (Horn et al., 2018) is a large-scale dataset with images collected from the real world, which has an extremely imbalanced class distribution. The iNat 2017 includes 579,184 training images in 5,089 classes with an imbalance ratio of 3,919:9, while the iNat 2018 is composed of 435,713 images from 8,142 classes with an imbalance ratio of 500:1.

Algorithm A-2 Algorithm of Meta-CALA

Input: Training data $\mathcal{D}^{\text{tr}} = \{\boldsymbol{x}_i, y_i\}_{i=1}^{N}$, metadata $\mathcal{D}^{\text{meta}} = \{\boldsymbol{x}_i^{\text{meta}}, y_i^{\text{meta}}\}_{i=1}^{M}$, batch size n, meta batch size m, #iteration $\mathcal{T}$, tuning parameters τ_1 and τ_2, neighborhood size K, and other training parameters;
Output: Learned parametrs $\boldsymbol{W}$ and $\boldsymbol{\Omega}$;

1: Initialize $\boldsymbol{W}^{(1)}$ and $\boldsymbol{\Omega}^{(1)}$;
2: **for** $t = 1$ to $\mathcal{T}$ **do**
3: Sample n and m samples from $\mathcal{D}^{\text{tr}}$ and $\mathcal{D}^{\text{meta}}$;
4: Formulate $\hat{\boldsymbol{W}}^{(t)}$ by Eq. (A.12);
5: Update $\boldsymbol{\Omega}^{(t+1)}$ by Eq. (A.13);
6: Update $\boldsymbol{W}^{(t+1)}$ by Eq. (A.14) with the resulting adjustment vectors $\boldsymbol{\delta}$;
7: **end for**

Places-LT (Liu et al., 2019) features an imbalanced training dataset consisting of 62,500 images distributed across 365 classes. The class distribution adheres to a natural power law pattern, with individual classes having a maximum of 4,980 images and a minimum of just 5 images. In contrast, the validation and testing sets are meticulously balanced, each containing 20 and 100 images per class, respectively.

Moreover, three subpopulation shift datasets, including CMNIST, Waterbirds, and CelebA are employed. Following Yao et al. (2022), these datasets are introduced as follows:

CMNIST consists of digits classified into two classes, with class 0 containing the original digits (0, 1, 2, 3, 4), and class 1 containing the remaining digits (5, 6, 7, 8, 9). The color of the digits is considered a spurious attribute, with the proportion of red to green samples being 8:2 in class 0, and 2:8 in class 1. In the validation set, the proportion of green to red samples is 1:1 for both classes, while in the test set, the proportion of green to red samples is 1:9 in class 0 and 9:1 in class 1. The training, validation, and test sets comprise 30,000, 10,000, and 20,000 samples, respectively. Following the manner of Arjovsky et al. (2019), labels were randomly flipped with a probability of 0.25.

Waterbirds (Sagawa et al., 2020) aims to classify birds as either "waterbirds" or "landbirds," with the spurious attribute being the scene context of "water" or "land." The Waterbirds dataset is a synthetic dataset that comprises images composed of a bird image taken from the CUB dataset (Wah et al., 2011) superimposed on a background randomly sampled from the Places dataset (Zhou et al., 2017). The bird categories in CUB consist of both landbirds and waterbirds. Two groups, including ("land" background, "waterbird") and ("water" background, "landbird"), are considered minority groups. The training set comprises 4,795 samples, of which 56 samples are "waterbirds on land" and 184 samples are "landbirds on water." The remaining training samples consist of 3,498 samples of "landbirds on land" and 1,057 samples of "waterbirds on water."

CelebA (Liu et al., 2015) is a dataset that contains face images of celebrities, with the classification labels being the hair color of the individuals, including "blond" or "not blond." The spurious attribute in this dataset is gender, i.e., male or female. In CelebA, the minority groups are ("blond," male) and ("not blond," female), with the number of samples for each group being 71,629 (dark hair, female), 66,874 (dark hair, male), 22,880 (blond hair, female), and 1,387 (blond hair, male).

Additionally, two generalized long-tail (GLT) benchmarks, ImageNet-GLT and MSCOCO-GLT, are utilized. Following Tang et al. (2022), they are detailed as follows:

ImageNet-GLT is a long-tailed version of the ImageNet (Russakovsky et al., 2015) dataset, which has three protocols, including CLT, ALT, and GLT. Among them, CLT and GLT protocols share the same training set, i.e., Train-GLT, with 113k samples over 1k classes. ALT protocol adopts a class-wise balanced Train-CBL with 114k images. The evaluation splits {Val, Test-CBL, Test-GBL} have {30k, 60k, 60k} samples, respectively. The number of samples for each class in Train-GLT ranges from 570 to 4, while all classes have 114 samples in Train-CBL.

MSCOCO-GLT is a long-tailed subset of MSCOCO-Attribute (Patterson and Hays, 2016; Lin et al., 2014) with 196 different attributes. Tang et al. (2022) cropped each object with multi-label attributes as independent images. Under CLT and GLT protocols, there are {Train-GLT, Val, Test-CBL, Test-GBL} with {144k, 2.9k, 5.8k, 5.8k} samples over 29 classes, where the number of samples for each class ranges from 61k to 0.3k. The ALT protocol has {32k, 1.4k, 2.9k} images for {Train-CBL, Val, Test-GBL}.

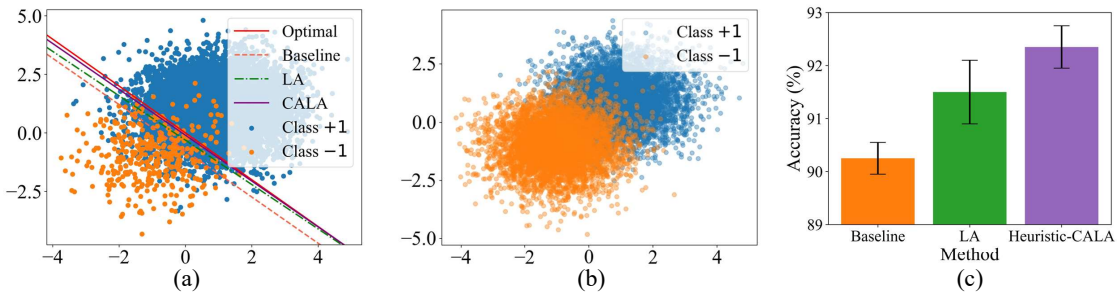

Figure A-1: (a): Distribution of the training data and classifiers of Baseline (CE loss), LA, and Heuristic-CALA. (b): Distribution of the test data. (c): Accuracy of the classifiers using the three methods.

D.II Introduction of Compared Baselines

Compared Methods for LT Benchmarks In the CIFAR-LT benchmark comparison, we evaluate both traditional and advanced LT baselines, including Class-Balanced CE loss (Cui et al., 2019), Class-Balanced Focal loss (Cui et al., 2019; Lin et al., 2017), Label-Distribution-Aware Margin with Data Re-Weighting strategy (LDAM-DRW) (Cao et al., 2019), Logit Adjustment (LA) (Menon et al., 2021), Adaptive Logit Adjustment (ALA) (Zhao et al., 2022), Balanced Knowledge Distillation (BKD) (Zhang et al., 2023), Mixup Shifted Label-Aware Smoothing model (MiSLAS) (Zhong et al., 2021), Label Distribution DisEntangling (LADE) (Hong et al., 2021), Gaussian Clouded Logit (GCL) (Li et al., 2022), Context Shift Augmentation (CSA) (Shi et al., 2023), and Sample-Adaptive Feature Augmentation with LDAM-DRW (LDAM-DRW-SAFA) (Hong et al., 2022). Additionally, we incorporate De-confound-TDE (Tang et al., 2020), which applies causal intervention during training and counterfactual reasoning during inference. Furthermore, we examine two meta-learning-based approaches, namely Meta-Weight-Net (Shu et al., 2019) and MetaSAug (Li et al., 2021), in the evaluation.

For iNat 2018 and Places-LT datasets, we conduct a comparison of several methods designed for LT learning. The compared methods include Decoupling (Kang et al., 2020), LA (Menon et al., 2021), Distribution Alignment (DisAlign) (Zhang et al., 2021), MiSLAS (Zhong et al., 2021), LADE (Hong et al., 2021), GCL (Li et al., 2022), LDAM-DRW-SAFA (Hong et al., 2022), and BKD (Zhang et al., 2023). Additionally, Meta-Weight-Net (Shu et al., 2019) and MetaSAug (Li et al., 2021) are also involved in the comparison.

Compared Methods for Subpopulation Shift Datasets In accordance with Yao et al. (2022), we conduct a comparative analysis of various robust methods for invariant feature learning. The methods include Invariant Risk Minimization (IRM) (Arjovsky et al., 2019), Information Bottleneck Invariant Risk Minimization (IB-IRM) (Ahuja et al., 2021), Variance Risk Extrapolation (V-REx) (Krueger et al., 2021), Correlation Alignment (CORAL) (Sun and Saenko, 2016), Group Distributionally Robust Optimization (GroupDRO) (Sagawa et al., 2020), Domain Mixup (DomainMix) (Xu et al., 2020), Fish (Shi et al., 2022), Learn Invariant Predictors via Selective Augmentation (LISA) (Yao et al., 2022), Confidence-based Model Selection (COSMOS) (Chen et al., 2023), and Progressive Data Expansion (PDE) (Deng et al., 2024).

Compared Methods for GLT Benchmarks The experimental comparisons involve several re-balancing methods designed to achieve better feature backbones. These methods include two-stage re-sampling approaches such as Classifier Re-training (cRT) (Kang et al., 2020), posthoc distribution adjustment techniques such as De-confound-TDE (Tang et al., 2020) and LA (Menon et al., 2021), multi-branch models with diverse sampling strategies such as Bilateral-Branch Network (BBN) (Zhou et al., 2020), invariant feature learning methods such as Invariant Feature Learning (IFL) (Tang et al., 2022), implicit semantic augmentation methods such as Reasoning-based Implicit Semantic Data Augmentation (RISDA) (Chen et al., 2022) and MetaSAug (Li et al., 2021), re-weighting loss functions such as Balanced Softmax (BLSoftmax) (Ren et al., 2020), LDAM (Cao et al., 2019), and BKD (Zhang et al., 2023), and data augmentation methods such as MixUp (Zhang et al., 2018), Random Augmentation (RandAug) (Cubuk et al., 2020), and Context Shift Augmentation (CSA) (Shi et al., 2023). These methods are compared with Heuristic-CALA and Meta-CALA to evaluate their effectiveness in GLT learning scenarios.

Dataset	Waterbirds		CMNIST		CelebA	
Method	Avg.	Worst	Avg.	Worst	Avg.	Worst
CORAL (Sun and Saenko, 2016)	90.3%	79.8%	71.8%	69.5%	93.8%	76.9%
IRM (Arjovsky et al., 2019)	87.5%	75.6%	72.1%	70.3%	94.0%	77.8%
GroupDRO (Sagawa et al., 2020)	91.8%	90.6%	72.3%	68.6%	92.1%	87.2%
DomainMix (Xu et al., 2020)	76.4%	53.0%	51.4%	48.0%	93.4%	65.6%
IB-IRM (Ahuja et al., 2021)	88.5%	76.5%	72.2%	70.7%	93.6%	85.0%
V-REx (Krueger et al., 2021)	88.0%	73.6%	71.7%	70.2%	92.2%	86.7%
Fish (Shi et al., 2022)	85.6%	64.0%	46.9%	35.6%	93.1%	61.2%
LISA (Yao et al., 2022)	91.8%	89.2%	74.0%	73.3%	92.4%	89.3%
COSMOS (Chen et al., 2023)	91.7%	89.3%	73.5%	72.4%	91.0%	88.6%
PDE (Deng et al., 2024)	92.4%	90.5%	78.1%	75.9%	92.1%	91.1%
Heuristic-CALA (Ours)	**94.3%**	**91.8%**	**79.5%**	**77.0%**	**94.4%**	**91.9%**

Table A-2: Comparison of the average and worst-group accuracy on three subpopulation shifts datasets.

D.III Experiments on Synthetic Dataset

To more intuitively demonstrate the effectiveness of the proposed CALA loss function, the assessment is also carried out on synthetic data, with consideration for binary classification. The training data have both class and attribute biases. For the class labeled "$+1$" in the training set, we generate the dataset by sampling from two two-dimension (2D) Gaussian distributions with means of $(+0.8, +0.8)$ and $(+1.2, +1.2)$, respectively. In contrast, for the class labeled "-1" in the training data, we sample the dataset from a Gaussian distribution with a mean of $(-1, -1)$. For the test dataset, we generate the data from a 2D Gaussian distribution with a mean of $(+1, +1)$ for class "$+1$," and from a 2D Gaussian distribution with a mean of $(-1, -1)$ for class "-1". The covariance matrices of all the aforementioned distributions are equal to $\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$.

It is obvious that there is attribute bias in the training and test data for class "$+1$". In order to simulate the long-tailed distribution of the training data, we set the class imbalance ratio to 24:1. To compare the performance of three loss functions, namely CE, LA, and CALA losses, a linear classifier of the form $f = \boldsymbol{w}\boldsymbol{x} + b$ is employed. The training and test datasets contain 10,000 samples in total, with 9,600 samples for class "$+1$" and 400 samples for class "-1" in the training data, and 5,000 samples for both classes in the test data.

Fig. A-1(a) depicts the distribution of the training data and the classifiers trained with the Baseline (CE loss), LA, and Heuristic-CALA. Fig. A-1(b) portrays the test data distribution. The accuracy of the classifiers trained using the three losses is presented in Fig. A-1(c). Remarkably, the classifier trained with Heuristic-CALA exhibits greater proximity to the Bayesian optimal classifier compared to those trained using LA and CE losses. Since LA is adept at mitigating class imbalances, it outperforms Baseline. Consequently, Heuristic-CALA yields the highest accuracy, whereas LA ranks second.

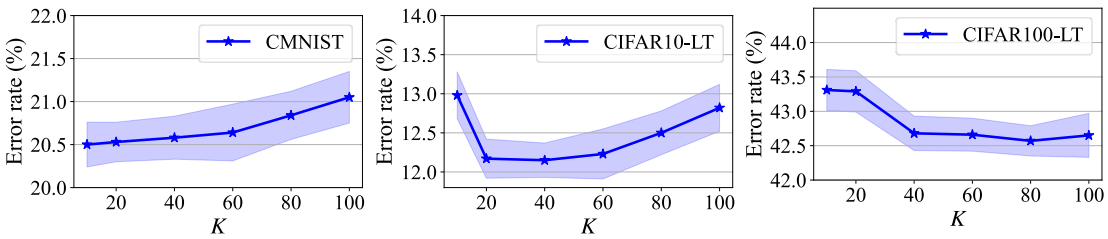

Figure A-2: Influence of different K values on CMNIST, CIFAR10-LT, and CIFAR100-LT.

Model	ResNet-110		Wide ResNet-28-10	
Dataset	CIFAR10	CIFAR100	CIFAR10	CIFAR100
Large Margin (Liu et al., 2016)	6.46%	28.00%	3.69%	18.48%
Disturb Label (Xie et al., 2016)	6.61%	28.46%	3.91%	18.56%
Focal loss (Lin et al., 2017)	6.68%	28.28%	3.62%	18.22%
Center loss (Wen et al., 2016)	6.38%	27.85%	3.76%	18.50%
L_q loss (Zhang and Sabuncu, 2018)	6.69%	28.78%	3.78%	18.43%
ISDA (Wang et al., 2019)	5.98%	26.35%	3.58%	17.98%
Heuristic-CALA (Ours)	**5.31%**	**25.02%**	**2.80%**	**17.25%**

Table A-3: Error rate of different methods on standard CIFAR datasets.

D.IV Complete Results on Subpopulation Shift Datasets

Table A-2 presents a comprehensive comparison of results across three subpopulation shift datasets. The results demonstrate that Heuristic-CALA consistently achieves the highest average and worst-group accuracy across these datasets. Specifically, Heuristic-CALA outperforms the second-best baselines by 1.23% and 1.03% in average and worst-group accuracy, respectively. This indicates its effectiveness in enhancing model generalization and improving performance for samples with rare attributes. Additionally, we apply the Wilcoxon signed-rank test to assess the significance of our performance improvement in terms of both average and worst accuracy. The resulting p-value of 0.03, which is below the threshold of 0.05, indicates statistically significant enhancement.

D.V More Sensitivity Analysis

Sensitivity analyses are conducted to investigate the impact of varying values of K, which determines the size of the neighborhood. The results for CMNIST, CIFAR10-LT, and CIFAR100-LT are presented in Fig. A-2. The optimal performance was obtained with $K = 10$ for CMNIST, $K = 40$ for CIFAR10-LT, and $K = 80$ for CIFAR100-LT. These findings suggest that as the label set size increases, a larger K is preferable. For all our experiments, K values were selected from the set {10, 20, 40, 60, 80, 100}, consistently yielding satisfactory results. Consequently, we recommend selecting K from this set.

D.VI Experiments on Standard CIFAR Datasets

We conduct experiments to evaluate the effectiveness of the CALA loss on the standard CIFAR datasets using two widely-used networks, including ResNet-110 (He et al., 2016) and Wide ResNet-28-10 (Zagoruyko and Komodakis, 2016). In this case, with the class distribution being balanced, only the adjustment term $v(\boldsymbol{x}, y)$ is effective. The compared methods follow those in the ISDA paper (Wang et al., 2019), including Large Margin (Liu et al., 2016), Disturb Label (Xie et al., 2016), Focal loss (Lin et al., 2017), Center loss (Wen et al., 2016), L_q loss (Zhang and Sabuncu, 2018), and ISDA (Wang et al., 2019). To ensure a

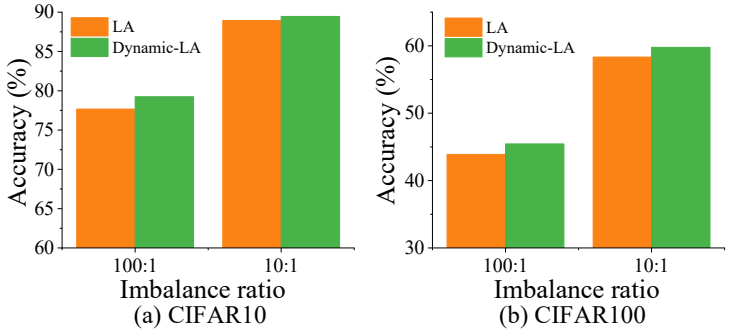

Figure A-3: Top-1 accuracy of LA and Dynamic-LA on CIFAR10-LT (a) and CIFAR100-LT (b) with the imbalance ratios of 10:1 and 100:1.

Networks	Parameters	Additional cost	
		CIFAR10	CIFAR100
ResNet-32	0.5M	6.6%	6.1%
ResNet-56	0.9M	6.2%	5.9%
ResNet-110	1.7M	1.9%	1.7%
DenseNet-BC-121	8.0M	0.7%	0.2%
DenseNet-BC-265	33.3M	0.5%	0.1%
Wide ResNet-16-8	11.0M	0.5%	0.1%
Wide ResNet-28-10	36.5M	0.4%	0.1%

Table A-4: Additional training time increased by Heuristic-CALA compared with CE loss.

Model	Parameters	Meta-Weight-Net	MetaSAug	Meta-CALA
ResNet-32	0.5M	8550	8612	8639
ResNet-56	0.9M	8701	8734	8766

Table A-5: Comparison of training time (s) when meta-learning is utilized in every iteration.

fair comparison, we only involve Heuristic-CALA in comparison as all compared methods do not rely on meta-learning.

We optimize the classifiers using SGD with an initial learning rate of 0.1 and a batch size of 128. For the ResNet model, we set the weight decay to 1×10^{-4} and decay the learning rate by 0.1 at the 80th and 120th epochs, with a total of 160 epochs. For the Wide ResNet model, we set the weight decay to 5×10^{-4} and decay the learning rate by 0.2 at the 60th, 120th, 160th, and 200th epochs, with a total of 240 epochs. Moreover, the values of τ_1 and τ_2 are both set to 1.

The comparison results are presented in Table A-3. It is evident that the Heuristic-CALA approach demonstrates superior performance and achieves state-of-the-art performance on the standard CIFAR datasets. These outcomes suggest the presence of attribute imbalances within the standard CIFAR datasets. Additionally, CALA loss is demonstrated to be an effective approach to address attribute imbalances and mitigate spurious correlations that may arise as a result of such imbalances.

D.VII Comparison between LA and Dynamic-LA

As discussed in the main text, when $K = +\infty$, Heuristic-CALA utilizes $p_{\mathrm{pr}}(y)$ to adjust the logits of samples for all classes. We refer to this adjusted approach as Dynamic-LA. The difference between LA and Dynamic-LA is that Dynamic-LA employs the class proportions of the predicted labels, which vary with model training. We compare the performance of these two methods on long-tailed CIFAR datasets, with the results presented in Fig. A-3. Dynamic-LA surpasses LA in all cases as the adjustment terms in Dynamic-LA can vary based on the model's performance during training. Indeed, the performance gap between classes cannot be attributed solely to the label frequency of the training data. For example, Xu et al. (2021) have demonstrated that classes with high variances tend to be more challenging and experience inferior performance. Thus, the variances of classes also affect their learning difficulty. A category with high label frequency and high variance may not outperform a category with low label frequency and low variance. Merely relying on class priors to adjust the training objective is insufficient since many other factors, besides class priors, affect the learning difficulty of classes. The label frequency of the predicted labels can more accurately reflect the relative learning difficulty of classes concerning the model.

D.VIII Results for Training Efficiency

We calculate the additional training time introduced by Heuristic-CALA compared with CE loss using various backbones on standard CIFAR data, including ResNet (He et al., 2016), Wide ResNet (Zagoruyko and Komodakis, 2016), and DenseNet (Huang et al., 2017). The experiments are conducted on a machine equipped with a single NVIDIA RTX 3090 GPU and 128 GB of RAM. The results, as reported in Table A-4, indicate that in comparison to the CE loss, Heuristic-CALA incurs only a marginal increase in

computational time. In addition, we calculate the training time of Meta-CALA and previous meta-learning-based approaches, including MetaSAug (Li et al., 2021) and Meta-Weight-Net (Shu et al., 2019). The comparison results are reported in Table A-5, verifying that the training time required for Meta-CALA is comparable with previous meta-learning-based algorithms.

References

Kartik Ahuja, Ethan Caballero, Dinghuai Zhang, Jean-Christophe Gagnon-Audet, Yoshua Bengio, Ioannis Mitliagkas, and Irina Rish. Invariance principle meets information bottleneck for out-of-distribution generalization. In *Proceedings of the Advances in Neural Information Processing Systems*, pages 3438–3450, 2021. 6, 7

Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019. 5, 6, 7

Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arechiga, and Tengyu Ma. Learning imbalanced datasets with label-distribution-aware margin loss. In *Proceedings of the Advances in Neural Information Processing Systems*, pages 1567–1578, 2019. 6

Annie S Chen, Yoonho Lee, Amrith Setlur, Sergey Levine, and Chelsea Finn. Confidence-based model selection: When to take shortcuts for subpopulation shifts. *arXiv preprint arXiv:2306.11120*, 2023. 6, 7

Xiaohua Chen, Yucan Zhou, Dayan Wu, Wanqian Zhang, Yu Zhou, Bo Li, and Weiping Wang. Imagine by reasoning: A reasoning-based implicit semantic data augmentation for long-tailed classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 356–364, 2022. 6

Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 3008–3017, 2020. 6

Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-balanced loss based on effective number of samples. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9260–9269, 2019. 4, 6

Yihe Deng, Yu Yang, Baharan Mirzasoleiman, and Quanquan Gu. Robust learning with progressive data expansion against spurious correlation. In *Proceedings of the Advances in Neural Information Processing Systems*, pages 1390–1402, 2024. 6, 7

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016. 8, 9

Youngkyu Hong, Seungju Han, Kwanghee Choi, Seokjun Seo, Beomsu Kim, and Buru Chang. Disentangling label distribution for long-tailed visual recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6626–6636, 2021. 6

Yan Hong, Jianfu Zhang, Zhongyi Sun, and Ke Yan. Safa: Sample-adaptive feature augmentation for long-tailed image classification. In *Proceedings of the European Conference on Computer Vision*, pages 587–603, 2022. 6

Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alex Shepard, Hartwig Adam, Pietro Perona, and Serge Belongie. The inaturalist species classification and detection dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8769–8778, 2018. 4

Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2261–2269, 2017. 9

Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, and Yannis Kalantidis. Decoupling representation and classifier for long-tailed recognition. In *Proceedings of the International Conference on Learning Representations*, 2020. 6

Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. Technical report, 2009. 4

David Krueger, Ethan Caballero, Joern-Henrik Jacobsen, Amy Zhang, Jonathan Binas, Dinghuai Zhang, Remi Le Priol, and Aaron Courville. Out-of-distribution generalization via risk extrapolation (rex). In *Proceedings of the International Conference on Machine Learning*, pages 5815–5826, 2021. 6, 7

Mengke Li, Yiu-ming Cheung, and Yang Lu. Long-tailed visual recognition via gaussian clouded logit adjustment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6929–6938, 2022. 6

Shuang Li, Kaixiong Gong, Chi Harold Liu, Yulin Wang, Feng Qiao, and Xinjing Cheng. Metasaug: Meta semantic augmentation for long-tailed visual recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5212–5221, 2021. 6, 10

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Proceedings of the European Conference on Computer Vision*, pages 740–755, 2014. 5

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2999–3007, 2017. 6, 8

Weiyang Liu, Yandong Wen, Zhiding Yu, and Meng Yang. Large-margin softmax loss for convolutional neural networks. In *Proceedings of the International Conference on Machine Learning*, pages 507–516, 2016. 8

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3730–3738, 2015. 5

Ziwei Liu, Zhongqi Miao, Xiaohang Zhan, Jiayun Wang, Boqing Gong, and Stella X Yu. Large-scale long-tailed recognition in an open world. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2537–2546, 2019. 5

Aditya Krishna Menon, Sadeep Jayasumana, Ankit Singh Rawat, Himanshu Jain, Andreas Veit, and Sanjiv Kumar. Long-tail learning via logit adjustment. In *Proceedings of the International Conference on Learning Representations*, 2021. 6

Genevieve Patterson and James Hays. Coco attributes: Attributes for people, animals, and objects. In *Proceedings of the ACM Conference on Computer and Communications Security*, pages 85–100, 2016. 5

Jiawei Ren, Cunjun Yu, Shunan Sheng, Xiao Ma, Haiyu Zhao, Shuai Yi, and Hongsheng Li. Balanced meta-softmax for long-tailed visual recognition. In *Proceedings of the Advances in Neural Information Processing Systems*, pages 4175–4186, 2020. 6

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015. 5

Shiori Sagawa, Pang Wei Koh, Tatsunori B Hashimoto, and Percy Liang. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. In *Proceedings of the International Conference on Learning Representations*, 2020. 5, 6, 7

Jiang-Xin Shi, Tong Wei, Yuke Xiang, and Yu-Feng Li. How re-sampling helps for long-tail learning? In *Proceedings of the Advances in Neural Information Processing Systems*, pages 75669–75687, 2023. 6

Yuge Shi, Jeffrey Seely, Philip HS Torr, N Siddharth, Awni Hannun, Nicolas Usunier, and Gabriel Synnaeve. Gradient matching for domain generalization. In *Proceedings of the International Conference on Learning Representations*, 2022. 6, 7

Jun Shu, Qi Xie, Lixuan Yi, Qian Zhao, Sanping Zhou, Zongben Xu, and Deyu Meng. Meta-weight-net: Learning an explicit mapping for sample weighting. In *Proceedings of the Advances in Neural Information Processing Systems*, pages 1919–1930, 2019. 6, 10

Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *Proceedings of the European Conference on Computer Vision Workshops*, pages 443–450, 2016. 6, 7

Kaihua Tang, Jianqiang Huang, and Hanwang Zhang. Long-tailed classification by keeping the good and removing the bad momentum causal effect. In *Proceedings of the Advances in Neural Information Processing Systems*, pages 1513–1524, 2020. 6

Kaihua Tang, Mingyuan Tao, Jiaxin Qi, Zhenguang Liu, and Hanwang Zhang. Invariant feature learning for generalized long-tailed classification. In *Proceedings of the European Conference on Computer Vision*, pages 709–726, 2022. 5, 6

Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011. 5

Yulin Wang, Xuran Pan, Shiji Song, Hong Zhang, Gao Huang, and Cheng Wu. Implicit semantic data augmentation for deep networks. In *Proceedings of the Advances in Neural Information Processing Systems*, pages 12635–12644, 2019. 8

Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao. A discriminative feature learning approach for deep face recognition. In *Proceedings of the ACM Conference on Computer and Communications Security*, pages 499–515, 2016. 8

Lingxi Xie, Jingdong Wang, Zhen Wei, Meng Wang, and Qi Tian. Disturblabel: Regularizing cnn on the loss layer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4753–4762, 2016. 8

Han Xu, Xiaorui Liu, Yaxin Li, Anil Jain, and Jiliang Tang. To be robust or to be fair: Towards fairness in adversarial training. In *Proceedings of the International Conference on Machine Learning*, pages 11492–11501, 2021. 9

Minghao Xu, Jian Zhang, Bingbing Ni, Teng Li, Chengjie Wang, Qi Tian, and Wenjun Zhang. Adversarial domain adaptation with domain mixup. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 6502–6509, 2020. 6, 7

Huaxiu Yao, Yu Wang, Sai Li, Linjun Zhang, Weixin Liang, James Zou, and Chelsea Finn. Improving out-of-distribution robustness via selective augmentation. In *Proceedings of the International Conference on Machine Learning*, pages 25407–25437, 2022. 5, 6, 7

Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In *Proceedings of the British Machine Vision Conference*, pages 87.1–87.12, 2016. 8, 9

Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. Mixup: Beyond empirical risk minimization. In *Proceedings of the International Conference on Learning Representations*, 2018. 2, 6

Songyang Zhang, Zeming Li, Shipeng Yan, Xuming He, and Jian Sun. Distribution alignment: A unified framework for long-tail visual recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2361–2370, 2021. 6

Shaoyu Zhang, Chen Chen, Xiyuan Hu, and Silong Peng. Balanced knowledge distillation for long-tailed learning. *Neurocomputing*, 527:36–46, 2023. 6

Zhilu Zhang and Mert Sabuncu. Generalized cross-entropy loss for training deep neural networks with noisy labels. In *Proceedings of the Advances in Neural Information Processing Systems*, pages 8792–8802, 2018. 8

Yan Zhao, Weicong Chen, Xu Tan, Kai Huang, and Jihong Zhu. Adaptive logit adjustment loss for long-tailed visual recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 3472–3480, 2022. 6

Zhisheng Zhong, Jiequan Cui, Shu Liu, and Jiaya Jia. Improving calibration for long-tailed recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16489–16498, 2021. 6

Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(6):1452–1464, 2017. 5

Boyan Zhou, Quan Cui, Xiushen Wei, and Zhaomin Chen. Bbn: Bilateral-branch network with cumulative learning for long-tailed visual recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9719–9728, 2020. 6