Novel Regularization via Logit Weight Repulsion for Long-Tailed Classification

TaeGil Ha
clfdydmlarnl@snu.ac.kr
SeulKi Park
seulki.park@snu.ac.kr
Jin Young Choi
jychoi@snu.ac.kr

Department of ECE, ASRI,
Seoul National University,
Seoul, Korea

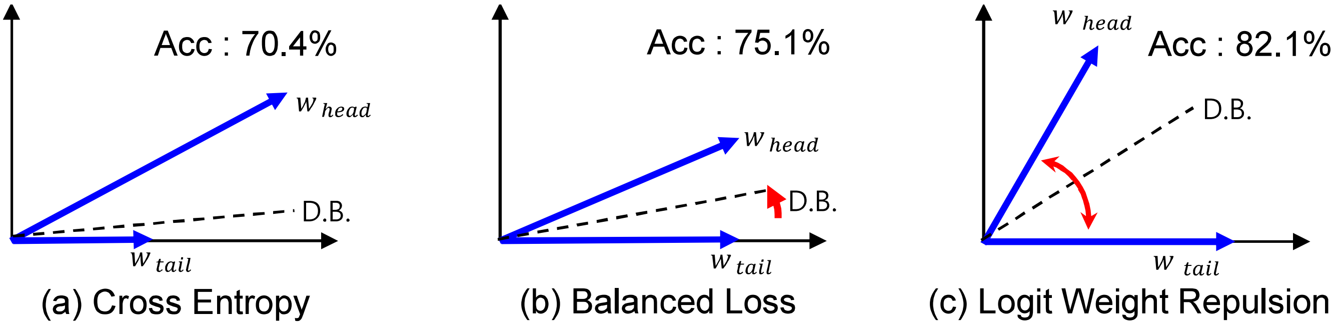
Abstract

Research to address the class imbalance problem aims to balance the impact of each class on the loss function because logit weight vectors tend to favor a majority class. To this end, researchers have introduced balanced losses such as margin-based loss and logit adjustment. The balanced losses succeed to classify the minority class better than the conventional loss. However, the balanced loss focuses on balancing the norm of logit weight, but overlooks the direction of logit weight vectors. As a result, the balanced loss sacrifices the head class performance by shrinking the region between the logit vectors. In this paper, we delve into the behavior of the gradient of the balanced loss and clarify how it shrinks the decision plane of each class from two perspectives. First, balanced loss pushes the decision boundary from tail to head within limited space, shrinking the decision plane of the head class. Second, balanced loss does not prevent the logit vectors to have a similar direction to each other during the update, shrinking region between logit vectors. Based on this study, we propose a new regularization called Logit Weight Repulsion (LWR), which encourages a logit weight vector for a class to repel those for other classes. This repulsion enlarges the region between the logit vectors for each class. The proposed LWR regularizer has been evaluated on benchmark datasets where ours achieves the state-of-the-art performance for long-tailed classification. Notably, LWR achieves performance improvements in minority classes without sacrificing the performance in majority classes.

1 Introduction

In real-world scenarios, highly imbalanced datasets [22, 23, 35] are common where some classes have a large number of samples while others have very few. Models trained on such data struggle to accurately predict the minority classes because the decision boundary between the minority class and the majority class is severely biased to predict the majority class well while sacrificing the minority class. Long-tailed visual recognition aims to address this problem [9, 10, 17, 28, 31].

One of the most popular approach for training a robust classifier in the imbalanced setting is training with a balanced loss. The balanced loss applies class-wise weights or introduces



(a) Cross Entropy (b) Balanced Loss (c) Logit Weight Repulsion

Figure 1: Brief comparison of logit weight vector and decision boundary (D.B.) trained by different training objectives. (a) Standard cross entropy results in decision boundary favored to the head class. (b) Balanced loss [5, 10, 26, 31] relaxes this problem by balancing logit weight magnitude. However, there exists a sacrifice of the head class region, and it cannot prevent logit weights to have a similar direction during the training, resulting in the angular distance between w_{head} and w_{tail} decreased. (c) Proposed logit weight repulsion (LWR) regularizer with balanced loss enlarges the narrowed angular distance. 'Acc' denotes accuracy on CIFAR-10 LT [5, 18] whose imbalance factor of 100.

adjustment terms, to relax the negative influence caused by class imbalance [3, 5, 10, 11, 26, 28, 31]. When training with imbalanced data using the conventional cross-entropy softmax loss, the norm of the classifier (logit) weight of the majority class tends to be significantly larger than that of the minority class, resulting in poor prediction performance for the minority class (Figure 1 (a)). The balanced loss balances the norm of the weight to improve the prediction performance of the minority class (Figure 1 (b)).

In this paper, we first investigate the limitation of balanced loss mentioned so far, and explain why this is problematic in long-tailed classification from two perspectives. First, balanced loss pushes the decision boundary from the tail class to the head class, shrinking the decision plane of the majority class as shown in Figure 1 (b). Second, the balanced loss does not prevent logit weight vectors to have a large cosine similarity to each other during the update, shrinking the decision plane of each class.

To mitigate this issue, we propose a novel regularizer named Logit Weight Repulsion that makes logit vectors for each class be repulsive from each other. Unlike balanced loss only considers the magnitude(norm) of the logit weight vector and overlooks its direction, our regularizer considers directions. As shown in Figure 1 (c), the adjusted logit vectors by our regularizer produce large output space between classes mitigating the misclassification of the tail class without sacrificing head class accuracy. Our experiments show performance improvement in both head and tail classes in long-tailed classification benchmarks [5, 23], and we demonstrate the effectiveness of the proposed method through ablation studies.

Our main contributions are summarized as follows:

- We investigate the limitation of balanced loss, which only tries to balance the magnitude of the logit weight vectors, but overlooks their direction. Balanced loss makes region between logit vectors of classes shrink.

- To mitigate the limitation of the balanced loss, we propose a Logit Weight Repulsion(LWR) regularizer, which controls the direction of the logit weight vector. LWR makes a logit weight vector for a class be repelled from the others to reduce their cosine similarity, enlarging the region between logit vectors of each class.

- We achieve considerable performance gain, outperforming the state-of-the-art methods without sacrificing the performance in the majority class.

2 Related Works

Balanced Loss Balanced loss aims to balance the negative influence caused by class imbalance. The re-weighting approach tries to adjust the impact of each class by multiplying weight, which is inversely proportional to the number of samples [16, 39], to the loss. Instead of weighting by simple inverse class frequency, Class-Balanced loss [10] proposed the effective number, which theoretically calculates the size of samples that can cover the class. On the other hand, instance-level re-weighting methods [20, 28] re-weight each sample based on its different impact. For example, Influence-balanced loss [28] re-weights samples with the influence of each sample on the model. Balanced Softmax loss [31] modifies the prediction logits by multiplying them with the training label frequencies to alleviate bias. Meanwhile, Label-Distribution-Aware Margin (LDAM) loss encourages larger margins for minority classes to shift the biased decision boundary. Recently, logit adjustment methods handle the long-tailed problem by adjusting the logits of a class prediction [15, 26].

Balanced loss methods show a prominent effect in balancing the magnitude(norm) of the logit weight vector, where imbalanced norm results in ill-conditioned decision boundary [21, 40, 43]. However, the balanced loss considers only the magnitude of the logit weight vectors, overlooking the direction of them.

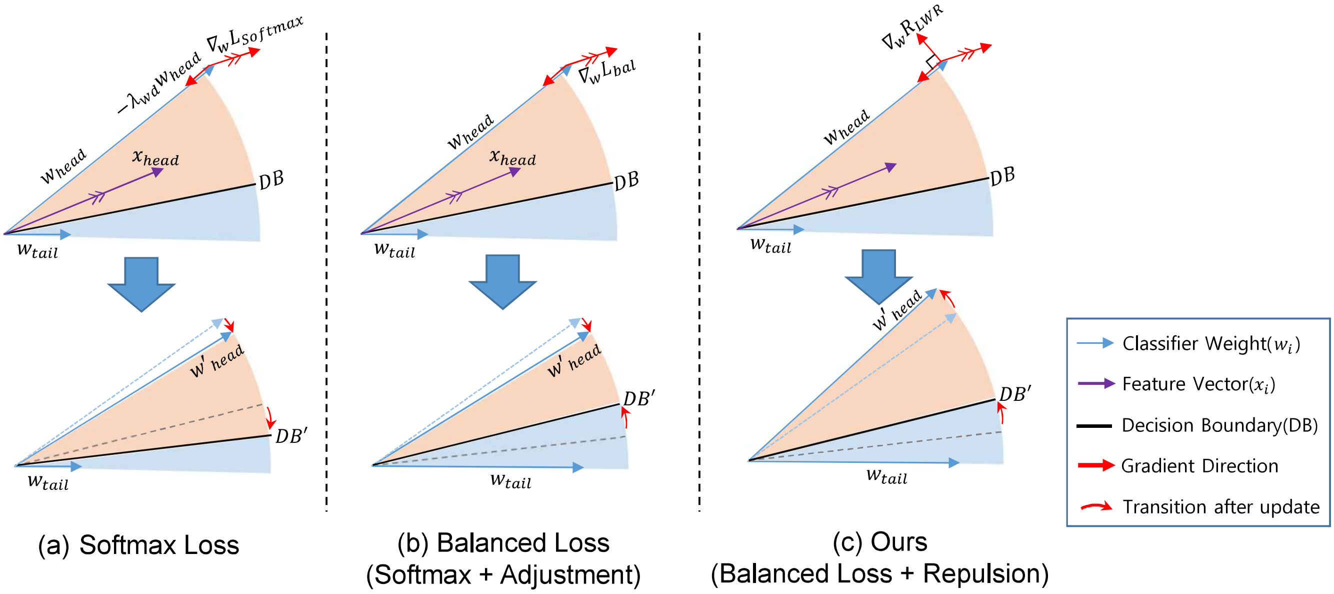
Re-sampling Re-sampling aims to train the model with more class-balanced data than the original data, by oversampling minority class data [1, 37] or undersampling majority class data [25, 36]. Advanced re-sampling methods generate synthetic minority class samples [2, 6, 13, 27, 29], or augment minority data in feature space [8, 44].

Advanced model There is approaches using multiple networks and ensemble them to solve imbalanced classification. BBN [46] trains two classifier heads with different samplers, respectively. RIDE [38] trains multiple classifiers to reduce model variance. ACE [4] trains multiple networks using data with different distributions, and ensemble them via optimization. In addition, to learn robust representations, PACO [9] uses contrastive learning [7], and recent works [24, 33] adopt large-scale vision-language pretraining models. For a more thorough study on this topic, we refer the reader to survey [45].

3 Methods

In this section, we provide the motivation for the proposed logit weight repulsion(LWR) regularizer and details, along with the training strategy. We begin with the mathematical analysis behind balanced losses, which show that their gradients only affect the decision boundary between the head and tail classes, balancing their norms (Section 3.1). Based on this analysis, we propose Logit Weight Repulsion (LWR) regularizer, which makes the logit weight vectors be distant by considering the gradient direction in Section 3.2. We present the training process in Section 3.3.

Notations. We provide notations that we use throughout this paper. The class-wise logit weight vectors are denoted by column vector w_i, and X_i denotes the set of feature vectors from samples belonging to class i. Direction vector of w_i is $\overline{w}_i = w_i/||w_i||_2$ where $||w_i||_2$ is L_2 norm(magnitude) of w_i.



(a) Softmax Loss

(b) Balanced Loss (Softmax + Adjustment)

(c) Ours (Balanced Loss + Repulsion)

Figure 2: Illustration of gradient directions and change of logit weight vectors and decision boundary between head and tail classes. (a) On imbalanced data, the norm of logit weight for the head class is much larger than that of the tail class in the original softmax loss. (b) Balanced Loss aims to balance the norm of the logit weight and pushes the biased decision boundary from the tail class, but this can shrink the decision plane for the head class. (c) Our proposed Logit Weight Repulsion Regularizer R_{LWR} encourages the weights of the head and tail classes to become farther apart from each other by taking into account the gradient direction.

3.1 Analysis of Balanced Loss

In this section, we discuss the potential limitation of balanced loss in imbalanced data by analyzing it from the perspective of gradient direction. The most commonly used balanced losses [3, 5, 10, 26, 31, 32] can be generalized as follows:

$$L_{\text{bal}} = -C_i \log \frac{\exp(w_i \cdot x_i + \delta_i)}{\sum_j \exp(w_j \cdot x_i + \delta_{i,j})} + \lambda_{wd} ||w_i||_2^2, \tag{1}$$

where $x_i \in X_i$ is the feature, C_i is a class-wise re-weighing coefficient, δ_i and $\delta_{i,j}$ are margin or adjustment term, and λ_{wd} is a hyper-parameter for weight decay [19].

Then, its gradient with respect to w_i can be written as:

$$\begin{aligned} -\nabla_{w_i} L_{\text{bal}} =& C_i x_i \left(1 - \frac{\exp\left(w_i \cdot x_i + \delta_{i,i}\right)}{\sum_j \exp(w_j \cdot x_i + \delta_{i,j})}\right) \\ &+ C_i \nabla_{w_i} \delta_i - C_i \sum_j \nabla_{w_i} \delta_{i,j} \frac{\exp\left(w_j \cdot x_i + \delta_{i,j}\right)}{\sum_j \exp\left(w_j \cdot x_i + \delta_{i,j}\right)} - 2\lambda_{wd} w_i. \end{aligned} \tag{2}$$

In the Balanced Loss methods, various designs of δ_i and $\delta_{i,j}$ are used to adjust the importance of tail and head classes, which results in different gradient magnitudes for updating logit weight vector w_i. This helps to push the decision boundary from the tail to the head classes by assigning more importance in the gradient update to the tail class than that of the head class as in Figure 2 (b).

In our work, we delve into the behavior of the gradient of the balanced loss as follows. Since δ_i and $\delta_{i,j}$ are usually designed as a function of the class frequency [5, 26, 31], $\nabla_{w_i} \delta_i$ and $\nabla_{w_i} \delta_{i,j}$ becomes 0 since δ_i and $\delta_{i,j}$ are not the function of w_i. Then, the negative gradients with respect to w_i can be decomposed into two directions: x_i and $-w_i$. It means the direction of the logit weight vector w_i is updated by the feature x_i, and its magnitude is reduced by itself, $-w_i$, by weight decay regularization.

The update by these directions leads to the following issues.

- First, pushing the decision boundary from the tail class to the head class makes decision plane of the head class shrink. Though this change in the decision boundary improves performance in the tail class, it sacrifices the head class performance.

- Second, when a hard example appears near the decision boundary, the logit weight vectors are updated to pull each other. For example, consider the case where a head-class sample x_{head} updates the logit weight vector w_{head} in Figure 2 (b). Since the gradient of $\nabla_{w_i} L_{\text{bal}}$ is a weighted addition of x_{head} and w_{head}, the direction of the gradient points between x_{head} and w_{head} (Figure 2 (b) Upper). As a result, the updated logit weight vector w'_{head} have moved closer to the other logit weight vector, causing the region between logit vectors to shrink when hard example appears near decision boundary. In conclusion, hard examples in confusing classes can be misclassified due to the shrink of region between logit vectors for each confusing class pairs.

To mitigate the aforementioned problem, we propose a Logit Weight Repulsion regularizer in the following section.

3.2 Logit Weight Repulsion (LWR)

As depicted in Figure 2 (c), our regularizer is designed to enlarge the region between logit vectors by increasing the angular distance between the logit weight vectors of different classes. It is achieved by updating w_i in the direction repelled from other w_j, $j \neq i$. From this motivation, we propose logit weight repulsion regularizer R_{LWR} that can increase the difference between logit weight vectors as:

$$R_{LWR} = -\sum_i \log \frac{\exp(\overline{w_i} \cdot \overline{w_j})}{\sum_j \exp(\overline{w_i} \cdot \overline{w_j})}, \tag{3}$$

Then, the derivative of R_{LWR} yields the following.

$$-\nabla_{w_i} R_{LWR} = -\sum_{j \neq i} C_{i,j} \left(\frac{\overline{w_j} - (\overline{w_i} \cdot \overline{w_j})\overline{w_i}}{||w_i||_2} \right) \tag{4}$$

The LWR regularizer takes into account the direction of other logit weight vectors w_j when updating w_i. Intuitively R_{LWR} leads to repulsion between pairs of logits by reducing cosine similarity $\overline{w_i} \cdot \overline{w_j}$. Analyzing the gradient of R_{LWR}, the update direction $(\overline{w_j} - (\overline{w_i} \cdot \overline{w_j})\overline{w_i})$, $j \neq i$ is orthogonal to $\overline{w_i}$) since $w_i \cdot (\overline{w_j} - (\overline{w_i} \cdot \overline{w_j})\overline{w_i}) = 0$. This update increases the angular distance between w_i and w_j, which enlarges the region between logit vectors of both i and j. This mechanism not only reduces the risk of misclassification of the tail class but also prevents degradation of the head class performance.

To enhance the effectiveness of repulsion, we apply the repulsion to a pair of logit weight vectors with high cosine similarity to each other. To this end, we define the final R_{LWR} as follows:

$$R_{LWR} = -\sum_i \left(\log \frac{\exp(1_{i,j,\theta} \overline{w_i} \cdot \overline{w_j})}{\sum_j \exp(1_{i,j,\theta} \overline{w_j} \cdot \overline{w_j})} \right), \tag{5}$$

where

$$1_{i,j,\theta} = \begin{cases} 1 & \text{if } \overline{w}_i \cdot \overline{w}_j > \theta \\ 0 & \text{otherwise.} \end{cases} \tag{6}$$

3.3 Training Schemes

3.3.1 Adaptive Weight Regularization

Optimizing regularization weight is crucial for effective training. In this section, we propose an adaptive regularization scheme that adjusts the weight of the logit weight regularization according to the training progress. It is designed to give strong repulsion initially to reduce the similarity between weights and gradually decrease the influence of LWR to focus on fine-tuning the classifier. For the total training epoch T, $\lambda(t)$ for the current epoch t is calculated by:

$$\lambda(t) = (1 - t/T)^2. \tag{7}$$

We validate the effectiveness of adaptive weight regularization in the Experimental section.

Then, our final training objective becomes:

$$L_{bal} + \lambda(t) R_{LWR}. \tag{8}$$

3.3.2 Implementation Trick

To adjust the scale of the cosine similarity [7], we introduced temperature $\tau = 0.1$ replacing $\overline{w_i} \cdot \overline{w_j}$ to $\overline{w_i} \cdot \overline{w_j}/\tau$ in Equation (5). And, for better generalization [12], we design the loss to repulse between logit weight vectors projected by trainable multi-layer perceptrons with one hidden layer whose hidden dimension is the same as an input logit weight vector's dimension.

4 Experiments

Implementation Details. We implemented all our methods with PyTorch [30] library. We evaluate our model on the most commonly used benchmark datasets: CIFAR100-LT [5] with various imbalance factors and ImageNet-LT [23], which is a large-scaled classification benchmark. We use Balanced Softmax loss [31] for the balanced loss (L_{bal}) in our experiments. For all datasets, we use stochastic Gradient Descent as the optimizer with the momentum of 0.9 and follow the augmentation strategy from [32]. For, CIFAR100-LT, we use ResNet-32 [14] as the backbone and train our network for 200 epochs. With the initial learning rate of 0.1, the learning rate decays at the 140th and 170th epoch by a factor of 0.1. We set weight decay to $5e-4$. For ImageNet-LT, we use resnet-50. The network is trained for 100 epochs with an initial learning rate of 0.1. The learning rate decays at the 60th and 80th epochs by 0.1. We set the weight decay as $2e-4$. We fix $\theta = 0.4$ for all experiments except Table 5.

Metrics. To provide a quantitative comparison between methods, we report classification accuracy. In addition, following [23], we report the accuracy for three disjoint subsets: many-shot classes with more than 100 training samples, medium-shot classes with 20 to 100 training samples, and few-shot classes with less than 20 samples. Additionally, we present the results for three imbalance factors (IFs) for CIFAR-100 LT, namely IF=100, IF=50, and

Methods	All			IF=100			
	IF=100	IF=50	IF=10	Few	Medium	Many	All
Softmax	38.6	44.0	56.4	8.7	37.6	65.3	38.6
CBL† [10]	39.6	45.4	58.0	-	-	-	39.6
IB-Loss [28]	39.8	46.4	50.4	20.4	44.9	50.3	39.8
Focal [20]	41.9	48.2	59.8	10.9	41.3	**68.7**	41.9
BBN†[46]	42.6	47.0	59.1	-	-	-	42.6
UNO-IC† [34]	43.1	-	58.6	-	-	-	43.1
SEQL† [32]	43.4	-	-	-	-	-	43.4
LFME† [41]	43.8	-	-	28.0	-	59.5	43.8
BS[31]	46.3	51.2	61.5	25.0	46.7	63.9	46.3
LA [26]	46.5	-	-	24.4	47.1	63.6	46.5
LDAM-DRW[5]	46.6	51.2	59.6	22.8	48.5	64.4	46.6
Ours(R_{LWR} + BS [31])	**50.7**	**54.3**	**63.6**	**32.2**	**50.0**	66.6	**50.7**

Table 1: Comparisons with state-of-the-art methods on CIFAR100-LT trained for 200 epochs with ResNet-32 [14] backbone. † are copied from their original paper.

Methods	Architecture	All	Few	Medium	Many
Focal [20]	ResNet-50	38.0	11.2	31.0	56.3
BBN§ [46]	ResNeXt	41.2	40.8	43.3	40
Softmax	ResNet-50	41.6	5.8	33.8	64.0
UNO-IC§ [34]	ResNeXt	45.7	9.3	38.7	**66.3**
OLTR§ [22]	ResNeXt	46.7	19.5	45.5	58.2
LFME§ [41]	ResNeXt	47.0	22.0	43.5	60.6
ESQL§ [32]	ResNeXt	47.3	15.7	44	62.5
cRT$^\#$ [17]	ResNet-50	47.3	26.1	44.0	58.8
CE-DRW [5]	ResNet-50	47.6	28.0	44.9	57.6
LWS$^\#$ [17]	ResNet-50	47.7	29.3	45.2	57.1
LA [26]	ResNet-50	48.0	29.1	44.4	58.5
BS [31]	ResNet-50	48.7	24.0	46.2	60.5
LDAM-DRW$^\#$ [5]	ResNet-50	49.8	**30.7**	46.9	60.4
Ours(R_{LWR} + BS [31])	ResNet-50	**51.5**	**30.7**	**49.2**	62.6

Table 2: Comparisons with state-of-the-art methods on ImageNet-LT [23]. For a fair comparison, we compare the baselines trained with ResNet-50 [14]/ResNext [42] for 90 or 100 epochs. # and § denote the results from [29] and [45], respectively.

IF=10, where IF is defined as the ratio of the number of training samples of the largest class to that of the smallest class.

4.1 Benchmark results

CIFAR100-LT Table 1 displays the performance evaluated on CIFAR-100LT [5]. We compare our method with various state-of-the-art methods [5, 10, 20, 26, 28, 31, 32, 34, 41, 46]. Applying our proposed regularizer R_{LWR} outperforms the other baselines in overall accuracy along with the accuracy in Few-shot, and Medium-shot classes. Notably, applying LWR

Methods	All	Few	Medium	Many
CE-DRW [5]	47.6	28.0	44.9	57.6
$+R_{LWR}$	50.4(+2.8)	30.0(+2.0)	47.7(+2.8)	60.4(+2.8)
BS [31]	48.7	24.0	46.2	60.5
$+R_{LWR}$	51.5(+2.8)	30.7(+6.7)	49.2(+3.0)	62.6(+2.1)
Focal [20]	38.0	11.2	31.0	56.3
$+R_{LWR}$	41.0(+3.0)	13.7(+2.5)	34.1(+3.1)	58.6(+2.3)
Logit Adjustment (LA) [26]	48.0	29.1	44.4	58.5
$+R_{LWR}$	50.1(+2.1)	31.5(+2.4)	46.9(+2.5)	60.4(+1.9)

Table 3: The impact of applying R_{LWR} to different losses. We add R_{LWR} with various balanced losses L_{bal} to verify its effectiveness on the large-scale dataset, ImageNet-LT[23]. Applying R_{LWR} brings significant performance improvements across all many-shot, medium-shot, and few-shot categories.

Methods	Few	Medium	Many	All
Balanced Softmax [31]	25.0	46.7	63.9	46.3
$+R_{LWR}$	28.8	**51.6**	64.0	49.3
$+R_{LWR}$ w/ proj	<u>30.0</u>	49.9	<u>65.3</u>	<u>49.5</u>
$+\lambda(t)R_{LWR}$ w/ proj	**32.2**	<u>50.0</u>	**66.6**	**50.7**

Table 4: Ablation study on CIFAR100-LT(IF=100). Our baseline is Balanced Softmax [31] reproduced by us. w/ proj denotes that we use projection layers explained in Section 3.3.

shows considerable improvement of accuracy in the Few-shot class from its baseline balanced loss, BS. Furthermore, it is noteworthy that applying our regularizer further improves the accuracies of the Medium and Many-shot classes, not compensating their performances.
Imagenet-LT The performance evaluated on Imagenet-LT [23] is displayed Table 2, respectively. Again, when applying our method to the Balanced Softmax loss [31], we observe significant performance improvements not only in few-shot classes but also in many-shot and medium-shot classes.

4.2 Ablation Studies

4.2.1 LWR with Other Balanced Losses

Our proposed LWR can widen the region between logit vectors trained with balanced loss, allowing for improved performance across all classes without sacrificing the performance of the majority class. To verify this, we apply our regularizer R_{LWR} to other types of balanced losses [5, 20, 26, 31] in Table 3. As shown in Table 3, we consistently observe significant performance improvements across all many-shot, medium-shot, and few-shot categories. This demonstrates that our regularizer can synergize with other balaced losses, further improving performance.

4.2.2 Effectiveness of LWR

To verify the effectiveness of the LWR, we conduct experiments on CIFAR-100 LT with IF=100. We apply the proposed LWR regularizer on the baseline [31] and training schemes

θ	Few	Medium	Many	All
-1.0	30.0	50.3	65.3	50.1
0	29.9	51.3	65.5	50.0
0.2	30.3	**51.4**	65.5	50.2
0.4	**32.2**	50.0	**66.6**	**50.7**
0.6	31.3	49.7	65.5	49.9
1.0	25.0	46.7	63.9	46.3

Table 5: Ablation study on CIFAR-100LT(IF=100). to verify the influence of repulsion threshold θ. Note that $\theta = -1.0$ is equal to threshold is not working, and $\theta = 1.0$ is equal to the baseline since $-1.0 \leq \overline{w_i} \cdot \overline{w_J} \leq 1.0$

explained throughout Section 3.2 and 3.3. The results of these experiments are displayed in Table 4. In Table 4, '$+R_{LWR}$' denotes that we trained our model with training objective in Equation (8) with fixed $\lambda(t) = 1$. This configuration shows +3.8% accuracy gain in Few and +4.9% gain in Medium, +0.1% gain in Many, and +3.0% gain in overall accuracy. '$+R_{LWR}$ w/ proj' denotes the use of projection MLP explained in Section3.3., and it further achieves performance gain in overall accuracy(+0.2%), Few(+1.2%), and Many(+1.3%). Finally, applying $\lambda(t)$ in Equation (7) results of our proposed method. Compared to the baseline, LWR regularizer achieves +7.2% in Few, +3.3% in Medium, +2.7% in Many, and +4.4% in overall accuracy. Notably, R_{LWR} does not compromise the accuracy of the majority class.

4.2.3 Influence of the repulsion threshold θ

To evaluate the effectiveness of the repulsion threshold θ in the indicator function $1_{i,j,\theta}$, we conduct experiments with different values of θ as shown in Table 5. Note that θ is compared with the cosine similarity between $\hat{w}_i$ and $\hat{w}_j$. When $\theta = -1.0$, no threshold is applied since $1_{i,j,\theta} = 1$ for any i, j, and $\theta = 1.0$ indicates no logit weight repulsion occurs. It shows the best performance when $\theta = 0.4$, and we use this value for all the other experiments.

5 Conclusion

In this paper, we have discussed the limitation of the balanced loss functions designed for the class-imbalance problem. The balanced loss pushes the decision boundary from tail class to head class within a limited region between logit vectors, and thus performance gain in the tail class leads to performance sacrifice of the head class. In addition, updating logit weight vectors by the balanced loss may shrink the region between logit vectors for each class, which leads to performance degradation of each class. To mitigate this problem, we proposed logit weight repulsion (LWR) regularizer for long-tailed classification. Through experiments and ablation studies, we have shown LWR synergizes the balanced losses, achieving state-of-the-art performance. We verify that the proposed LWR seamlessly cooperate with various balanced losses, facilitating performance enhancement of tail classes along with the head classes.

Limitations and Future work. The proposed R_{LWR} requires repulsion threshold θ. Throughout the experiments, we used a fixed value of θ rather than optimizing it depending on class pairs i and j. We expect that designing adaptive repulsion threshold $\theta_{i,j}$ depending on class i and j will show better performance than using a fixed value. We leave it to future work.

Acknowledgement

This work was supported by IITP grant funded by Korea government(MSIT) [No.B0101-15-0266, Development of High Performance Visual BigData Discovery Platform for Large-Scale Realtime Data Analysis; NO.2021-0-01343, Artificial Intelligence Graduate School Program (Seoul National University)]

References

[1] Mateusz Buda, Atsuto Maki, and Maciej A Mazurowski. A systematic study of the class imbalance problem in convolutional neural networks. *Neural networks*, 106:249–259, 2018.

[2] Chumphol Bunkhumpornpat, Krung Sinapiromsaran, and Chidchanok Lursinsap. Safe-level-smote: Safe-level-synthetic minority over-sampling technique for handling the class imbalanced problem. In *Advances in Knowledge Discovery and Data Mining: 13th Pacific-Asia Conference, PAKDD 2009 Bangkok, Thailand, April 27-30, 2009 Proceedings 13*, pages 475–482. Springer, 2009.

[3] Jonathon Byrd and Zachary Lipton. What is the effect of importance weighting in deep learning? In *International conference on machine learning*, pages 872–881. PMLR, 2019.

[4] Jiarui Cai, Yizhou Wang, and Jenq-Neng Hwang. Ace: Ally complementary experts for solving long-tailed recognition in one-shot. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 112–121, 2021.

[5] Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arechiga, and Tengyu Ma. Learning imbalanced datasets with label-distribution-aware margin loss. *Advances in neural information processing systems*, 32, 2019.

[6] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357, 2002.

[7] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[8] Peng Chu, Xiao Bian, Shaopeng Liu, and Haibin Ling. Feature space augmentation for long-tailed data. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIX 16*, pages 694–710. Springer, 2020.

[9] Jiequan Cui, Zhisheng Zhong, Shu Liu, Bei Yu, and Jiaya Jia. Parametric contrastive learning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 715–724, 2021.

[10] 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/CVF conference on computer vision and pattern recognition*, pages 9268–9277, 2019.

[11] Qi Dong, Shaogang Gong, and Xiatian Zhu. Imbalanced deep learning by minority class incremental rectification. *IEEE transactions on pattern analysis and machine intelligence*, 41(6):1367–1381, 2018.

[12] Kartik Gupta, Thalaiyasingam Ajanthan, Anton van den Hengel, and Stephen Gould. Understanding and improving the role of projection head in self-supervised learning. *arXiv preprint arXiv:2212.11491*, 2022.

[13] Hui Han, Wen-Yuan Wang, and Bing-Huan Mao. Borderline-smote: a new over-sampling method in imbalanced data sets learning. In *Advances in Intelligent Computing: International Conference on Intelligent Computing, ICIC 2005, Hefei, China, August 23-26, 2005, Proceedings, Part I 1*, pages 878–887. Springer, 2005.

[14] 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.

[15] 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/CVF conference on computer vision and pattern recognition*, pages 6626–6636, 2021.

[16] C. Huang, Y. Li, C. C. Loy, and X. Tang. Learning deep representation for imbalanced classification. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[17] Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, and Yannis Kalantidis. Decoupling representation and classifier for long-tailed recognition. *arXiv preprint arXiv:1910.09217*, 2019.

[18] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[19] Anders Krogh and John Hertz. A simple weight decay can improve generalization. *Advances in neural information processing systems*, 4, 1991.

[20] 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 2980–2988, 2017.

[21] Jialun Liu, Yifan Sun, Chuchu Han, Zhaopeng Dou, and Wenhui Li. Deep representation learning on long-tailed data: A learnable embedding augmentation perspective. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2970–2979, 2020.

[22] 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/CVF conference on computer vision and pattern recognition*, pages 2537–2546, 2019.

[23] 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/CVF conference on computer vision and pattern recognition*, pages 2537–2546, 2019.

[24] Alexander Long, Wei Yin, Thalaiyasingam Ajanthan, Vu Nguyen, Pulak Purkait, Ravi Garg, Alan Blair, Chunhua Shen, and Anton van den Hengel. Retrieval augmented classification for long-tail visual recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6959–6969, 2022.

[25] Inderjeet Mani and I Zhang. knn approach to unbalanced data distributions: a case study involving information extraction. In *Proceedings of workshop on learning from imbalanced datasets*, volume 126, pages 1–7. ICML, 2003.

[26] Aditya Krishna Menon, Sadeep Jayasumana, Ankit Singh Rawat, Himanshu Jain, Andreas Veit, and Sanjiv Kumar. Long-tail learning via logit adjustment. *arXiv preprint arXiv:2007.07314*, 2020.

[27] Sankha Subhra Mullick, Shounak Datta, and Swagatam Das. Generative adversarial minority oversampling. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1695–1704, 2019.

[28] Seulki Park, Jongin Lim, Younghan Jeon, and Jin Young Choi. Influence-balanced loss for imbalanced visual classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[29] Seulki Park, Youngkyu Hong, Byeongho Heo, Sangdoo Yun, and Jin Young Choi. The majority can help the minority: Context-rich minority oversampling for long-tailed classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6887–6896, 2022.

[30] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

[31] Jiawei Ren, Cunjun Yu, Xiao Ma, Haiyu Zhao, Shuai Yi, et al. Balanced meta-softmax for long-tailed visual recognition. *Advances in neural information processing systems*, 33:4175–4186, 2020.

[32] Jingru Tan, Changbao Wang, Buyu Li, Quanquan Li, Wanli Ouyang, Changqing Yin, and Junjie Yan. Equalization loss for long-tailed object recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11662–11671, 2020.

[33] Changyao Tian, Wenhai Wang, Xizhou Zhu, Jifeng Dai, and Yu Qiao. Vl-ltr: Learning class-wise visual-linguistic representation for long-tailed visual recognition. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXV*, pages 73–91. Springer, 2022.

[34] Junjiao Tian, Yen-Cheng Liu, Nathaniel Glaser, Yen-Chang Hsu, and Zsolt Kira. Posterior re-calibration for imbalanced datasets. *Advances in Neural Information Processing Systems*, 33:8101–8113, 2020.

[35] 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.

[36] Jason Van Hulse, Taghi M Khoshgoftaar, and Amri Napolitano. Experimental perspectives on learning from imbalanced data. In *Proceedings of the 24th international conference on Machine learning*, pages 935–942, 2007.

[37] Vikas Verma, Alex Lamb, Christopher Beckham, Amir Najafi, Ioannis Mitliagkas, David Lopez-Paz, and Yoshua Bengio. Manifold mixup: Better representations by interpolating hidden states. In *International conference on machine learning*, pages 6438–6447. PMLR, 2019.

[38] Xudong Wang, Long Lian, Zhongqi Miao, Ziwei Liu, and Stella X Yu. Long-tailed recognition by routing diverse distribution-aware experts. *arXiv preprint arXiv:2010.01809*, 2020.

[39] Yu-Xiong Wang, Deva Ramanan, and Martial Hebert. Learning to model the tail. In *Advances in Neural Information Processing Systems*, 2017.

[40] Tong Wu, Ziwei Liu, Qingqiu Huang, Yu Wang, and Dahua Lin. Adversarial robustness under long-tailed distribution. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8659–8668, 2021.

[41] Liuyu Xiang, Guiguang Ding, and Jungong Han. Learning from multiple experts: Self-paced knowledge distillation for long-tailed classification. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part V 16*, pages 247–263. Springer, 2020.

[42] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1492–1500, 2017.

[43] Xi Yin, Xiang Yu, Kihyuk Sohn, Xiaoming Liu, and Manmohan Chandraker. Feature transfer learning for face recognition with under-represented data. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5704–5713, 2019.

[44] Yuhang Zang, Chen Huang, and Chen Change Loy. Fasa: Feature augmentation and sampling adaptation for long-tailed instance segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3457–3466, 2021.

[45] Yifan Zhang, Bingyi Kang, Bryan Hooi, Shuicheng Yan, and Jiashi Feng. Deep long-tailed learning: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

[46] Boyan Zhou, Quan Cui, Xiu-Shen Wei, and Zhao-Min Chen. Bbn: Bilateral-branch network with cumulative learning for long-tailed visual recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9719–9728, 2020.