De-biasing Neural Networks with Estimated Offset for Class Imbalanced Learning

Byungju Kim[1,2] Hyeong Gwon Hong[1] Junmo Kim[1]*
School of Electrical Engineering, KAIST, South Korea[1]
Mathpresso Inc[2]
{byungju.kim, honggudrnjs, junmo.kim}@kaist.ac.kr

Abstract

The imbalanced distribution of the training data makes the networks biased to the frequent classes. Existing methods to resolve the problem involve re-sampling, re-weighting, or cost-sensitive learning. Most of them anticipate that emphasizing the minority classes during the training would help the network to learn better representations. In this paper, we propose a method for reparameterizing softmax classifiers' offsets so that training is less sensitive to class imbalance. We first observe that the trained offset of the baseline linear classifier is biased toward the majority classes due to the imbalance. Instead of the trained offset, we define the estimated offset, and constrain it to be uniform over the classes. In experiments with long-tailed benchmarks, our method exhibits the best performance. These experiments verify that our proposed method effectively encourages the networks to learn better representations for minority classes while preserving the performance for the majority classes.

1. Introduction

Recently, numerous models based on deep neural networks (DNNs) have shown remarkable advances in various fields of the machine learning. Among them, the recognition is the most classic and the most important task. Networks trained for recognition are widely used as backbone networks for other tasks. A better performing architecture on the recognition often induces the improved performance by providing features of higher quality. Consequently, there has been intensive research for finding better-performing networks. The size and complexity of state-of-the-art architectures have been growing over time, which has led to the growth of model capacity. This grown capacity requires a large volume of the training data and, as a consequence, it is not surprising any more that a machine surpasses the human performance in recognition tasks. From this perspective, the outstanding performance of DNNs is grounded on the large volume of the training data. With a larger dataset, the same network would perform better[18], but if the provided data is insufficient, the over-fitting problem would deteriorate the model performance, despite its superior complexity and capacity.

The data deficiency problem arises in many tasks. In this study, we particularly aim to resolve the class imbalanced learning, which is a special case of the data deficiency. In class imbalanced learning, we assume there exists an extreme disparity among the number of samples, in which the frequent classes have sufficient amount of training samples, whereas the infrequent classes have only a few samples for training. The main challenge we confront is that a few majority classes dominate the whole training process [8]. If we use all the data we have, the over-fitting problem arises for the minority classes due to the insufficient training samples. As only a few samples are provided from the minority classes, the network can memorize the data, resulting in poor generalization. Conversely, if we discard samples from majority classes, the overall volume of the dataset shrinks so that we should employ smaller network architecture. It would result in worse performance. The simplest and straight-forward solution is to collect a large and well-balanced dataset. Unfortunately, it may not be available due to the nature of the various target domain. To resolve this problem, the importance of the decision boundary has been pointed out in [8, 10]. They showed that re-training and adjustment of the classifier can improve the overall performance. Specifically, [10] showed that regulation of the classifier can also positively affect the feature representation.

To this end, we further investigate the role of the classifier and propose a novel method to *de-bias* a neural network with a biased sample frequency. The high-level idea of our method is to draw a pair-wise decision boundary through the middle of the two feature distributions. When the prior distribution of the training data is highly imbalanced, it is known that the decision boundary is leaned toward the minority classes [4, 10]. Assuming that each class

is identically distributed, the ideal decision boundary passes through the middle of the two class-conditional distributions. Although numerous works have investigated this issue, the offset term of the classifier has not been highlighted in the literature, and the offset is often ignored or fixed as zero. However, we found that using a zero-fixed offset is not enough to resolve all the penalties caused by the imbalance. To remove the unintended penalty, we define an *empirical offset* for the top layer using the feature mean of each class. Unlike the trained offset, the empirical offset is independent of the prior distribution that possesses a severe imbalance. In addition, we regulate them to have the uniform value. We geometrically show that the proposed structure of the offset can achieve the goal of our high-level idea: decision boundary in the middle. Then, we modify the empirical offset into the *estimated offset*, which is a learnable alternative of the empirical offset. The overall concept of our high-level idea is similar to that in [10]. The main difference lies in the methodology of inducing the decision boundary to be drawn in the middle. In [10], they force the weight vectors to have the same magnitude. However, we regulate the boundary with an additional loss term.

In summary, our main contributions are as follows: 1) we show that when an offset term of the linear classifier is not used, an advantage is in fact given to the majority classes, 2) we propose a novel method to induce uniform offset term independent of the class cardinality, and 3) we experimentally show that our method outperforms the existing methods on real image datasets with a long-tailed distribution.

2. Method

In this section, we describe our detailed method. We investigate the offset term of the softmax classifier on the top of the trained neural network. Using Gaussian discriminative analysis, we show how the imbalanced sample frequency affects the offset. Then, we introduce an alternative approach estimating the offset and propose the estimated offsets to have a uniform value for all the classes. We also provide a geometrical interpretation of our proposed method.

2.1. Preliminary

Suppose we have a training dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$ of N data-label pairs, where $x_i \in \mathbb{R}^n$ denotes the n dimensional input data, and $y_i \in \{1, ..., K\}$ denotes the corresponding label; we consider a K-class categorization problem. We assume that each data has a single label. Therefore, $\mathcal{D}$ can be further segmented into K subsets; $\mathcal{D} = \bigcup_{j=1}^{K} \mathcal{D}_j$, where $\mathcal{D}_j$ is a collection of data that has j as its label. Without loss of generality, we can set the order of the class cardinality as $|\mathcal{D}_1| \geq ... \geq |\mathcal{D}_K|$. Given the dataset, a conventional framework to train a model is the empirical risk minimization (ERM). With an appropriate loss function $\mathcal{L}(x_i, y_i; \theta)$, the goal of the ERM is to minimize the overall risk:

$$\min_{\theta} \mathbb{E}_{(x,y)\sim\mathcal{D}}[\mathcal{L}(x, y; \theta)], \tag{1}$$

where θ denotes the learnable parameters. Let $f : \mathbb{R}^n \rightarrow \mathbb{R}^d$ be a neural network followed by a softmax classifier, where d denotes the feature dimension. The softmax classifier is a linear classifier with weight vectors and offset. The neural network estimates the posterior probability of x_i being categorized as class k; $\hat{Pr}(y_i = k|x_i) = \frac{\exp(w_k^T f(x_i)+w_{k0})}{\sum_j \exp(w_j^T f(x_i)+w_{j0})}$, where w_j and w_{j0} denote, respectively, the weight vector and the offset for the j-th node of the softmax classifier.

In this study, we further assume that the class-conditional feature distribution follows the multivariate Gaussian distribution. More specifically, we assume that each class-conditional distribution of the feature vector $f(x)$ shares the same covariance matrix. In fact, this assumption is already lain underneath the softmax classifier because the linear decision boundary implies a tied covariance matrix in the Gaussian discriminant analysis. Therefore, we can convert the softmax classifier into a *generative classifier* without an additional training process [1].

Then, we obtain a solution for the the weight vectors and offsets in closed form.

$$\begin{aligned} w_k &= \mathbf{\Sigma}^{-1}\mu_k \\ w_{k0} &= \ln Pr(y_i = k) - \frac{1}{2}\mu_k^T \mathbf{\Sigma}^{-1}\mu_k, \end{aligned} \tag{2}$$

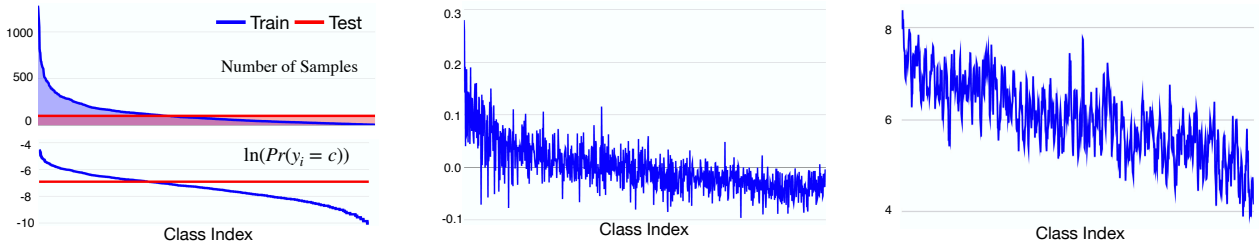
where $\mathbf{\Sigma}$ denotes the shared covariance matrix, and μ_k denotes the mean of the class-conditional distribution for k-th class. Specifically, when Eq.(2) holds, the following equation is satisfied:

$$\frac{\exp(w_k^T f(x_i) + w_{k0})}{\sum_j \exp(w_j^T f(x_i) + w_{j0})} = \frac{Pr(x_i|y_i = k)Pr(y_i = k)}{\Sigma_j Pr(x_i|y_i = j)Pr(y_i = j)}. \tag{3}$$

In Eq.(3), the left hand side is the probability of x_i being classified into the k^{th} class in the softmax classifier, whereas the probability in the generative classifier is located on the right hand side. The related theoretical details are reported in [1]. Under the ERM framework, we rarely investigate the mean and covariance matrix of a class-conditional distribution. However, Eq.(2) implies that they are closely related with the trained parameters.

2.2. How the class imbalance affects classifier offset

In class imbalanced learning, the major challenge is the disparity between the prior distributions of the training set and the test set. The prior distribution is often regarded as a uniform distribution in the test time, whereas highly imbalanced in the training time [2, 4, 8]. Fig.1(a) shows the difference of the prior distributions between training and test



(a) Prior distribution (b) Trained offset (c) Empirical offset

Figure 1. On ImageNet-LT dataset with ResNet-10 (a)Prior distribution, (b)Trained offsets, and (c)Empirical offset($\frac{1}{2}\tilde{\mu}_k^T w_k$). The network is trained with ERM. Figures show that the trained offset follows prior distribution. The trend of the empirical offset also follows the prior distribution.

time. As the network learns directly from the data distribution, the network would learn the imbalanced prior distribution of the training data. Note that the class indices are sorted by the class cardinality; $Pr(y_i = j) > Pr(y_i = k)$ if $j < k$. In other words, the learned prior distribution in Eq.(2) would follow the prior distribution of the training set. The offset term of Eq.(2) shows how the imbalance deteriorates the deep neural networks. We can notice that the first term of the offset ($\ln Pr(y_i = k)$) would be affected directly from the prior distribution that the network has learned. This implies that the neural network is penalizing the minority classes by using the offset term. Fig.1(a) illustrates $\ln Pr(y_i = k)$ and the number of samples and Fig.1(b) shows the trained offset w_{k0} for each class. They show that the trained offset follows the sample frequency.

A simple, yet oppressive method to avoid the penalty that comes from the imbalanced prior distribution $Pr(y_i = k)$ in Eq.(2) is not to use the offset in the softmax layer. That is, fix w_{k0} as zero for all k. By employing the zero-fixed offset, the classifier outputs a uniform posterior distribution for a zero signal. Further, several studies do not use the offset of their softmax classifier [2, 8, 10, 13]. The common underlying assumption in these prior works and in this study is that $Pr(y_i = k)$ should have the same value for all classes in the test scenario.

However, Eq.(2) shows that the offset contains an additional term (-$\frac{1}{2}\mu_k^T \mathbf{\Sigma}^{-1}\mu_k$). Therefore, the zero-fixed offset can be interpreted in two different ways: 1) the variation among $\mu_k^T \mathbf{\Sigma}^{-1}\mu_k$ for all $k \in \mathcal{Y}$ is negligible, and, thus, we are ignoring this term; otherwise 2) we are adding $\frac{1}{2}\mu_k^T \mathbf{\Sigma}^{-1}\mu_k$ for each class. To investigate which interpretation is correct, we need to estimate $\mu_k^T \mathbf{\Sigma}^{-1}\mu_k$. However, it contains matrix inversion operation, which is computationally inefficient and unstable. Furthermore, it is highly uncertain for the minority classes owing to their low sampling rate. Instead, we can find an alternative way for the estimation using Eq.(2). Considering the formulation of the trained weight vector, we can write $\mu_k^T \mathbf{\Sigma}^{-1}\mu_k = \mu_k^T w_k$. Using the feature mean, we can approximate $\mu_k^T \mathbf{\Sigma}^{-1}\mu_k \approx \tilde{\mu}_k^T w_k$, where $\tilde{\mu}_k = \frac{1}{|\mathcal{D}_k|}\sum_{x_i \in \mathcal{D}_k} f(x_i)$. Here, we define the *empirical offset* as $w_{k0}^{emp} = \frac{1}{2}\tilde{\mu}_k^T w_k$. Fig.1 (c) presents the empirical offset for each class. They are calculated with a network trained by ERM, which verifies that the first interpretation barely holds. It is clearly *not* negligible. It shows definite trend following the class cardinality.

Consequently, we should employ the second interpretation: discarding the offset term is equivalent to adding $\frac{1}{2}\mu_k^T \mathbf{\Sigma}^{-1}\mu_k$ for each class. One can notice that $\frac{1}{2}\mu_k^T \mathbf{\Sigma}^{-1}\mu_k$ has larger value for the majority classes in Fig.1(c). That is, discarding the offset term in the classifier is, in fact, the same as adding a larger value for the majority classes, penalizing the minority classes. In other words, the addition of $\frac{1}{2}\mu_k^T \mathbf{\Sigma}^{-1}\mu_k$ is an opportunity cost of resolving the problem of imbalanced prior distribution. To cancel out the penalty, we reformulate the posterior probability with the empirical offset: $Pr(y_i = k|x_i) = \frac{\exp(w_k^T f(x_i) - w_{k0}^{emp})}{\sum_j \exp(w_j^T f(x_i) - w_{j0}^{emp})}$. Note that the empirical offset should be subtracted, not added. By employing the *empirical offset*, we can discard the penalty caused by the prior information from the classifier.

In addition, the empirical offset makes the decision boundary being drawn in the middle. If we define d_{jk} as a distance from μ_j to the decision boundary between class j and k, we can calculate d_{jk} with the weight vectors and class centers. By substituting $\Sigma^{-1}\mu_j$ for w_j, we can calculate d_{jk} as

$$\begin{aligned} d_{jk} &= \frac{|\mu_j^T(w_j - w_k) - w_{j0}^{emp} + w_{k0}^{emp}|}{||w_j - w_k||_2} \\ &= \frac{|\frac{1}{2}(\mu_j^T \Sigma^{-1}\mu_j + \mu_k^T \Sigma^{-1}\mu_k) - \mu_j^T \Sigma^{-1}\mu_k|}{||w_j - w_k||_2}. \end{aligned} \tag{4}$$

As the covariance matrix is symmetric, one can notice that d_{jk} and d_{kj} are the same; the decision boundary goes through the middle of μ_1 and μ_2. The decision boundary in the middle implies that we are treating the two classes equally, independent of the sample frequency [10].

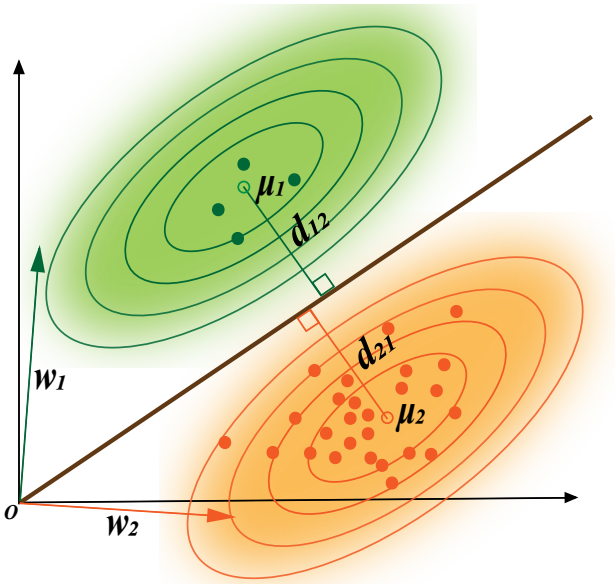


Figure 2. Assuming the class-conditional distribution of the features follows the multivariate Gaussian distribution, the optimal decision boundary passes through the center of their means. The proposed empirical offset makes d_{12} and d_{21} to be the same. It indicates that the decision boundary goes through the middle of μ_1 and μ_2 by geometrical similarity. Moreover, the identical empirical offset is the same as zero offset for all classes. Geometrically, zero offset means the decision boundary passes through the origin as well. It also implies that the classifier is de-biased, so it outputs the uniform posterior distribution for zero signal.

2.3. Uniform Constraint for Empirical Offset

Going back to the beginning, zero-fixing the offset is advantageous from the viewpoint of the classifier bias. A uniform offset means that the classifier is unbiased; if a zero signal is given to the classifier, it will output a uniform prediction. However, we cannot arbitrarily force the empirical offsets to have the same value. To this end, we introduce an additional uniform constraint using softmax and the entropy function:

$$\mathcal{L}_{uniform}(w_0^{emp}) = -H(softmax(-w_0^{emp})), \quad (5)$$

where w_0^{emp} denotes a vector form of the empirical offset, and $H(\cdot)$ denotes the entropy function. The uniform constraint evokes the first interpretation for the fixed offset: the variation among $\mu_k^T \mathbf{\Sigma}^{-1} \mu_k$ is negligible. The major difference is that we induce the variation to be negligible using the uniform constraint, not arbitrarily assume that it is negligible. Therefore, with the uniform constraint, the distribution of feature vectors changes accordingly through the training process. Fig.2 briefly illustrates the feature points and weight vectors. By the definition of the empirical offset, the uniform constraint, $\mathcal{L}_{uniform}(w_0^{emp})$, encourages the inner products to have the same value for all classes.

2.4. Estimated Offset

To train the network with the empirical offset, we need $\tilde{\mu}_k$, which requires to feed-forward the whole dataset $\mathcal{D}$. It is computationally inefficient. One can estimate $\tilde{\mu}_k$ within a mini-batch. However, in class imbalanced learning, it is unreliable owing to the low sample frequency of the minority classes. Some mini-batches may not even contain a sample from a minority class. To address this problem, we introduce an *estimated offset*, $\tilde{w}_{k0} = \frac{1}{2} c_k^T w_k$, where c_k is a learnable parametrization for $\tilde{\mu}_k$. Consequently, the posterior distribution becomes $Pr(y_i = k|x_i) = \frac{\exp(w_k^T f(x_i) - \tilde{w}_{k0})}{\sum_j \exp(w_j^T f(x_i) - \tilde{w}_{j0})}$. As a result, our final objective for optimization is as follows:

$$\begin{aligned}\mathcal{L}_{final}(\mathcal{D}) =& \mathbb{E}_{x_i \sim \mathcal{D}}[\mathcal{L}_{cls}(x_i, y_i) + \lambda_1 ||f(x_i) - c_{y_i}||_2^2] \\ &+ \lambda_2 \mathcal{L}_{uniform}(\tilde{w}_0),\end{aligned} \quad (6)$$

where $\mathcal{L}_{cls}(\cdot, \cdot)$ denotes the softmax cross-entropy loss with the estimated offset and λs are hyper-parameters to be tuned to balance the loss terms. The second term is also known as the center loss [21] in the literature. In [21], Wen *et al.* mainly focus on the discriminative power of the learned features. They show that the center loss enlarges the inter-class feature difference, while reducing the intra-class feature variation. Similarly, we expect the l_2 loss to encourages c_k to become the feature mean. Then, the neural network is trained to be unbiased with the uniform constraint with the estimated offset. To train c_k in Eq.6, we follow the protocol presented in [21].

3. Related Works

Numerous methods for the class imbalanced learning have been proposed in the literature. To resolve this problem, a network should be trained to extract better feature representations of the minority classes while preserving the successful performance of the majority classes. The primal approach for the class imbalanced learning is to re-sample the minority classes [3, 4, 7, 15, 20]. Equivalent to the re-sampling method, samples can be re-weighted during the training. In this approach, the samples are often weighted by the reciprocal of the volume of each class. Although the re-sampling and re-weighting approach attenuate the dominance of the majority classes, they were not effective enough for training deep neural networks. The networks still suffer from the over-fitting problem, since the samples from minority classes appear repeatedly over the training process. Cui *et al.* claim that if the number of samples is sufficiently large, additional data is a near-duplicate of existing samples with high probability [4]. Considering the overlap among the training samples, authors define the *effective number* of samples and proposed a class-balanced

loss, which adjusted the weight depending on the volume of each class.

Another line of research is to design a novel loss function, specialized for class imbalanced learning [2, 9]. An important feature that differentiates these approaches from the sampling based methods mentioned above is that these approaches focus on the margin. Considering the imbalance of the sample frequency, more margin is given to the minority classes. The underlying anticipation of these approaches is that by penalizing the majority classes and encouraging the network to focus more on the minority classes, the network would be helped to learn more representative features. Cao *et al.* proposed a label-distribution-aware margin (LDAM) loss [2]. The authors modified the soft margin loss [19] by encouraging the minority classes to have a larger margin considering the volume of each class. The margin is theoretically induced from the generalization error bounds. Combined with new optimization scheduling, the LDAM loss significantly improves the performance on class imbalanced learning. Khan *et al.* claim that the feature representations of the minority classes show a highly concentrated distribution, resulting in high uncertainty[9]. By measuring the uncertainty of each class, the authors proposed a novel loss function for class imbalanced learning.

Empirical studies of class imbalanced learning have also been conducted [8, 10]. In [8], the authors experimentally show that joint learning of the representation and decision boundary is more effective than the end-to-end learning in long-tailed recognition. For better representation, they proposed to additionally employ a sampling strategy, such as class-balanced sampling [7, 20], square root sampling [14, 15], and progressively balanced sampling [2, 4]. Several weight re-balancing techniques, such as τ-normalized and learnable weight scaling (LWS), also improve the long-tailed recognition accuracy [8]. In [10], the authors focused on how the features form the clusters in the feature space, and how they are affected by the generalization performance of the networks. The trained model shows remarkable performance for the majority classes with great generalization. Similar to the weight re-balancing technique proposed in [8], they proposed to adjust the weight norms of the linear classifier to find better decision boundary. In a similar way to the approach in our study, both works [8, 10] attempted to resolve the class imbalanced learning by regulating the classifier. Although the authors of [8, 10] have claimed their own perspectives, the effect of their algorithm is similar to those of margin based approaches [2, 9]. The main differentiating feature of our work is that we focus on the role of the offset, whereas previous works mainly focused on the weight vectors. We claim that the importance of the offset term is underestimated, and we experimentally show the effectiveness of our method in the following section.

4. Experiments

4.1. Overview

To verify our proposed method, we evaluated it with real world datasets: ImageNet-LT, Places2-LT [13], and SUN397 [22]. ImageNet-LT and Places2-LT are long-tailed versions of the ImageNet [17] and Places2 [25] datasets. In [13], Liu *et al.* introduced these datasets. Both datasets are modified by sampling the images following Pareto distribution. The main difference between the two datasets is that ImageNet-LT is an object-centric dataset, whereas the Places2-LT is a scene-centric dataset. Network architectures and the training frameworks are the same for both datasets. However, the context which the networks should learn is different. Therefore, we can verify if the algorithm can be generalized by using these datasets. The ImageNet-LT dataset contains 115.8K images from 1,000 classes. The most frequent class has 1,280 training samples, whereas the least frequent class has 5 training samples. The Places2-LT dataset contains 184.5K training samples from 365 classes. The disparity between the most and the least frequent class is severer in Places2-LT, where the most frequent class has 4,980 samples, while the least frequent class has only 5 samples for training.

SUN-397 [22] is another long-tailed dataset with smaller number of samples. Despite its volume, this dataset suffers from an even more imbalanced distribution. The least frequent class has only a single training image, whereas the most frequent class has 1,132 images. Although these datasets have a long-tailed distribution, their test sets are constructed to have the same number of samples for each class. Therefore, the overall accuracy indicates the balanced accuracy, which equally emphasizes all the classes; thus, we can verify if a method trains a network to recognize every class independent of their sample frequency.

For the ImageNet-LT dataset, the ResNet-10 [6] architecture is used for comparison, while ResNet-152 [6] is employed for the Places2-LT and SUN397 datasets. Except for the experiments with ImageNet-LT, the networks are pretrained with the original ImageNet [17] dataset. All networks are trained with the SGD optimizer with momentum 0.9. For data augmentation, we follow the protocol of [11], with image resolution of 224x224. For all experiments, λ_1 and λ_2 are fixed to 1.

4.2. Evaluation Results

We summarize the overall performance and comparison of the proposed method with ImageNet-LT and Places-LT in Table 1. As our baseline, a network is trained for classification with ERM. The results show that our model performs the best in terms of overall performance. More importantly, the performance of our method is better balanced compared to that of the other methods. In Table 1, the results show that

	Many-shot >100	**Medium-shot** $\leq$ 100 & >20	**Few-shot** <20	**All**	**Many-shot** >100	**Medium-shot** $\leq$ 100 & >20	**Few-shot** <20	**All**
	ImageNet-LT				**Places2-LT**			
Baseline[6]	47.4	7.9	0.1	21.9	45.2	23.8	8.1	28.2
Lifted Loss[16]	35.8	30.4	17.9	30.8	41.1	35.4	24.0	35.2
Focal Loss[12]	36.4	29.9	16.0	30.5	41.1	34.8	22.4	34.6
Range Loss[24]	35.8	30.3	17.6	30.7	41.1	35.4	23.2	35.1
FSLwF[5]	40.9	22.1	15.0	28.4	43.9	29.9	29.5	34.9
OLTR[13]	43.2	35.1	18.5	35.6	44.7	37.0	25.3	35.9
Baseline+RS[10]	46.6	36.9	22.2	38.5	40.8	38.3	25.8	36.6
WVN+RS[10]	47.8	38.8	**34.0**	41.6	41.2	39.4	28.8	38.2
cRT[8]	-	-	-	41.8	42.0	37.6	24.9	36.7
τ-norm [8]	-	-	-	40.6	37.8	**40.7**	31.8	37.9
LWS[8]	-	-	-	41.4	40.6	39.1	28.6	37.6
Ours w/o UC	**56.6**	30.8	9.4	37.7	**45.9**	35.2	21.1	36.0
Ours	49.9	**40.6**	31.4	**42.9**	38.0	39.9	**40.7**	**39.4**

Table 1. Performance evaluation on ImageNet-LT and Places2-LT. We also report many-shot, medium-shot, and few-shot performance separately. It shows that our method performs the best in overall accuracy. Even without the uniform constraint, the model with estimated offset significantly improves the performance.

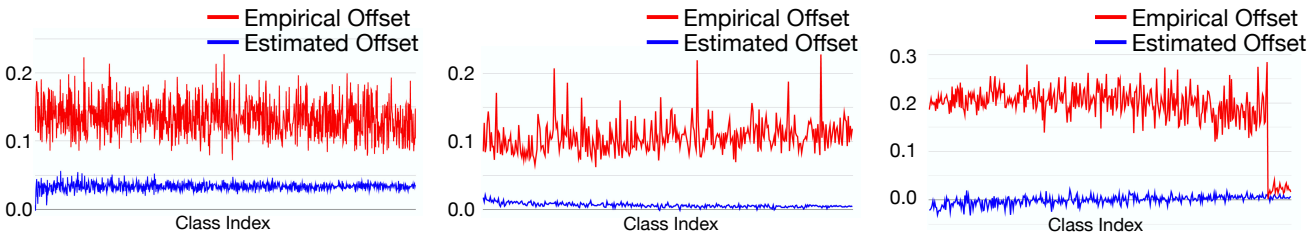
Method	ResNet-10	ResNeXt-50	ResNeXt-152
Baseline[6, 23]	21.9	39.5	43.1
OLTR[13]	37.3	46.3	50.3
WVN+RS [10]	41.6	47.7	52.2
cRT [8]	41.8	49.5	52.4
τ-norm [8]	40.6	49.4	52.8
LWS [8]	41.4	49.9	53.3
Ours	**42.9**	**51.6**	**57.1**

Method	Acc
Baseline [6]	48.0
Cost-Sensitive [7]	52.4
WVN+RS [10]	53.0
Model Reg. [20]	54.7
MetaModelNet [20]	57.3
OLTR [13]	58.7
Ours	**58.9**

Table 2. (*left*) Performance with deeper model on ImageNet-LT. Our method shows even better performance with deeper networks. For ResNet-10 architecture, the performance gain of our algorithm compare to the second best model is 1.1%. On the other hand, with ResNeXt-152 architecture, our method outperforms the second best model with 3.8% margin. (*right*) With SUN397 dataset, our method shows the best performance as well.

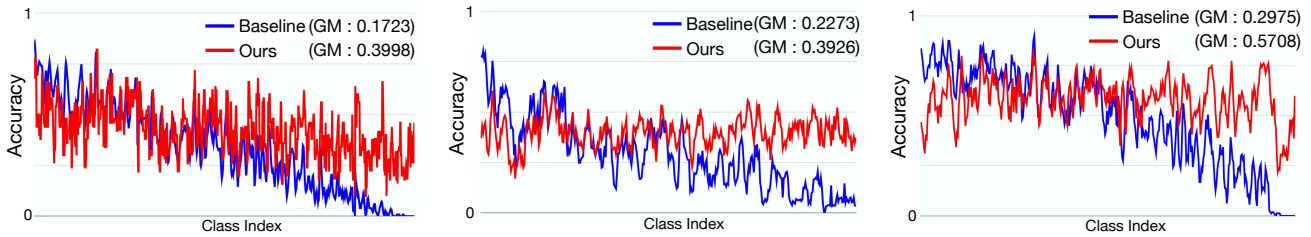
the baseline performance is highly imbalanced. The accuracy for the many-shot classes is significantly better than the accuracy for the few-shot classes, resulting in disappointing overall accuracy. The table implies that even without the uniform constraint, the performance is improved across the board. However, the accuracy for many-shot classes still outperforms the accuracy for few-shot classes. The performance of the few-shot classes needs further improvement compared to the previous works. By employing the uniform constraint, the networks, marked as Ours, are trained to recognize the infrequent classes better. However, the accuracy for the many-shot classes is degraded as its rebound. Despite the degradation, the performance is still comparable to that of other methods. Especially in ImageNet-LT, our methods show the best performance in both many-shot and medium-shot classes. It implies that our algorithm not only reinforces the classification capability for the few-shot classes, but also improves the generalization of the network.

We provide the performance on the ImageNet-LT dataset with a deeper model in Table 2 (left). Kang *et al.* observed that the deeper architectures significantly improve the performance with ImageNet-LT [8]; the table verifies the effect of network depth. Following [8], we evaluated our method with the ResNeXt-50 and ResNeXt-152 architectures [23]. The results show that our algorithm is still effective in training deeper models. Interestingly, the improvement by employing our method is more significant with a deeper model. Among the architectures with different depth, the performance of the deepest model, ResNeXt-152, outperforms the second model with the largest margin. Deeper depth of the networks often represents deeper understanding about the training data. Therefore, deeper networks can learn the imbalanced prior distribution better than the relatively shallow networks. By minimizing the negative impact of the imbalance, our algorithm performs even better with the deeper networks. Table 2 (right) presents the overall performance on the SUN397 dataset. Although it is marginal, our method shows the best performance.



(a) ImageNet-LT (b) Places2-LT (c) SUN397

Figure 3. Empirical offset and estimated offset on real-image dataset with long-tailed distribution. Figures show the estimated offset is well representing the empirical offset. Moreover, the consistent magnitude implies the effectiveness of the proposed uniform constraint. It suggests that the decision boundary is drawn in the middle of the feature clusters.



(a) ImageNet-LT (b) Places2-LT (c) SUN397

Figure 4. Class-wise accuracy and geometric mean of the performance for our model and baseline. It shows that the performance of our method is more balanced in both quantitatively and qualitatively. With our algorithm, the networks are trained to recognize minority classes as good as the majority classes.

4.3. Empirical Offset vs. Estimated Offset

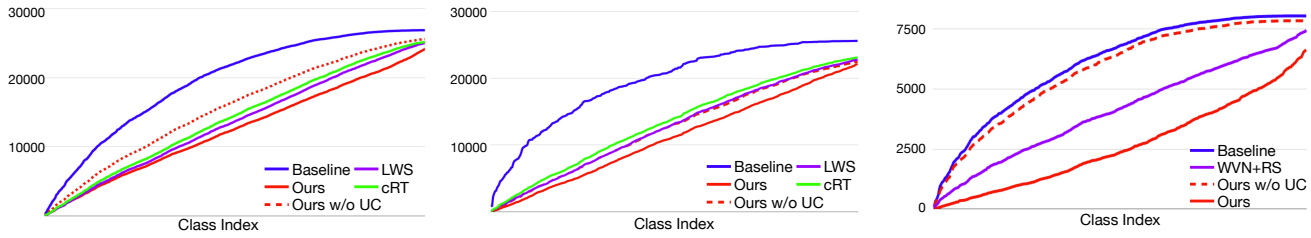
In Fig.3, we present the magnitude of the empirical offsets and the estimated offsets. Both offsets show consistency over classes. Compared to the situation in Fig.1 (c), the uniformity validates the effectiveness of the uniform constraint. Moreover, the gap between the empirical and the estimated offsets is also consistent. Note that the addition of an arbitrary constant to the estimated offset does not modify its softmax output. Therefore, the consistent gap verifies that the estimated offset represents well the empirical offset. Interestingly, the least frequent classes have significantly small empirical offset in the SUN397 dataset. As we subtract the offset, the smaller magnitude implies that the network is still penalizing the least frequent classes. The drop in magnitude does not occur with the estimated offset. It implies that the estimated offset is not biased against the minority classes, even when the empirical offset is.

4.4. Balance in Performance

For more detailed analysis, Fig.4 presents the class-wise accuracy. Qualitatively, the figures imply that our method encourages the network to have a more balanced accuracy over the classes. It also provides the geometric mean (GM) of the class-wise accuracy. The GM is defined as $(\prod_{i=1}^{K} acc_i)^{1/K}$, where acc_i denotes the accuracy for class i. As we can recognize from the definition, if there is a zero-valued element, the GM becomes zero. In the case of the baseline model, there are several classes with zero accuracy. It is not desirable for the analysis to obtain a zero GM owing to the few zero-accuracy classes. Thus, we calculated GM by replacing the zero-accuracy with $\epsilon = 0.001$. In Fig.4, both the GM and the graph show that our method reinforces the balance of the class-wise accuracy. Compared to the algebraic mean presented in Table 1 and Table 2 (right), the GM of the baseline drops significantly. However, the geometric performances of our method are comparable with the algebraic performances, which suggests that the performance of our method is well-balanced.

Additionally, we present a cumulative false positive in Fig.5. As the baseline network is biased to the majority classes, numerous samples from the minority classes are misclassified to the majority classes, which makes the graph concave. Conversely, if we excessively penalize the majority classes, so the networks misclassify numerous samples to the minority classes, the graph would become convex shape. The ideal shape is a linear graph. In Fig.5 (a) and (b), compared to the other methods, our method shows higher linearity. It implies that the networks trained with our method is better balanced. Meanwhile, Fig.5 (c) implies that our method is over-penalizing the majority classes. Despite its convexity, it shows the best overall performance.

The balanced performance is an important feature of our work. As it is shown in Fig.4, the baseline network is



(a) ImageNet-LT (b) Places2-LT (c) SUN397

Figure 5. Cumulative false positive. If the classifier is not biased to neither majority nor minority classes, the graph would be linearly increasing. The figures suggest that our method is better-balanced. Moreover, they also suggest that the estimated offset even without uniform constraint improves the balance in class-wise performance.

	Many-shot >100	**Medium-shot** ≤ 100 & >20	**Few-shot** <20	**All**	**Many-shot** >100	**Medium-shot** ≤ 100 & >20	**Few-shot** <20	**All**
	ImageNet-LT				**Places2-LT**			
Baseline	47.4	7.9	0.1	21.9	45.2	23.8	8.1	28.2
UC only	56.5	28.3	9.1	36.5	43.2	28.5	24.2	32.9
Ours w/o UC	**56.6**	30.8	9.4	37.7	**45.9**	35.2	21.1	36.0
Ours	49.9	**40.6**	31.4	**42.9**	38.0	**39.9**	**40.7**	**39.4**

Table 3. Ablation study with ImageNet-LT and Places2-LT. The results show that the estimated offset is more critical. However, the networks perform the best when we employ both methods. Especially for the few-shot classes, the performance is remarkably improved by employing both methods. Ablating one of the two methods severely deteriorates the performance. It implies that these methods are complementary to each other.

trained to recognize majority classes better than the minority classes. It is owing to the imbalance of the class cardinality. In several applications, this may arise a discrimination issue against the minorities. The results presented in Fig.4 and Fig.5 imply that the network is treating all classes equally importantly; the networks are categorizing test samples without prior knowledge.

4.5. Ablation Study

This work proposed two methods for class imbalanced learning: estimated offset, and uniform constraint. To analyze the contribution of each method, we ablated one of the two methods. Table 3 shows the evaluation performance of networks trained with single method. In the table, Ours w/o UC means that the model is trained with the estimated offset without the uniform constraint. On the contrary, UC only means that the model is trained with the uniform constraint only. Although the uniform constraint improves the performance by itself, the results imply that the estimated offset is more effective for class imbalanced learning than the uniform constraint. By employing both methods, the overall performance is significantly improved compared to the performance of either method. Especially, the balance in performance is greatly degraded by both of the ablations. The results suggest that these methods are complementary to each other.

5. Conclusion

In this paper, we have shown that a linear classifier without offset term penalizes the minority classes. To resolve the involuntary penalty, we defined the *empirical offset*, which requires an access to class-wise feature mean. The advantage of the empirical offset is that it is not biased owing to the imbalanced prior distribution of the training data. We also defined the *estimated offset* as a learnable alternative of the empirical offset and experimentally showed that it represents well the empirical offset. Furthermore, we proposed the *uniform constraint* for the estimated offset, which makes the classifier unbiased. We also introduced the geometric interpretation of the proposed method. It encourages the classifier to draw the decision boundary in the middle of the class-conditional distribution. The benefit of the proposed approach was verified by experiments with large-scale real-image datasets.

Acknowledgement

This work was supported by Institute for Information & communications Technology Promotion(IITP) grant funded by the Korea government(MSIT) (No.2016-0-00563, Research on Adaptive Machine Learning Technology Development for Intelligent Autonomous Digital Companion).

References

[1] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*, pages 196–203. Springer, 1 edition, 2007.

[2] Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arechiga, and Tengyu Ma. Learning imbalanced datasets with label-distribution-aware margin loss. In *Advances in Neural Information Processing Systems*, pages 1565–1576, 2019.

[3] 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.

[4] 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 9268–9277, 2019.

[5] Spyros Gidaris and Nikos Komodakis. Dynamic few-shot visual learning without forgetting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4367–4375, 2018.

[6] 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.

[7] Chen Huang, Yining Li, Chen Change Loy, and Xiaoou Tang. Learning deep representation for imbalanced classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5375–5384, 2016.

[8] 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.

[9] Salman Khan, Munawar Hayat, Syed Waqas Zamir, Jianbing Shen, and Ling Shao. Striking the right balance with uncertainty. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 103–112, 2019.

[10] B. Kim and J. Kim. Adjusting decision boundary for class imbalanced learning. *IEEE Access*, 8:81674–81685, 2020.

[11] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.

[12] 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.

[13] 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.

[14] Dhruv Mahajan, Ross Girshick, Vignesh Ramanathan, Kaiming He, Manohar Paluri, Yixuan Li, Ashwin Bharambe, and Laurens van der Maaten. Exploring the limits of weakly supervised pretraining. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 181–196, 2018.

[15] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

[16] Hyun Oh Song, Yu Xiang, Stefanie Jegelka, and Silvio Savarese. Deep metric learning via lifted structured feature embedding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4004–4012, 2016.

[17] 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 (IJCV)*, 115(3):211–252, 2015.

[18] Chen Sun, Abhinav Shrivastava, Saurabh Singh, and Abhinav Gupta. Revisiting unreasonable effectiveness of data in deep learning era. In *Proceedings of the IEEE international conference on computer vision*, pages 843–852, 2017.

[19] Feng Wang, Weiyang Liu, Haijun Liu, and Jian Cheng. Additive margin softmax for face verification. *CoRR*, abs/1801.05599, 2018.

[20] Yu-Xiong Wang and Martial Hebert. Learning to learn: Model regression networks for easy small sample learning. In *European Conference on Computer Vision*, pages 616–634. Springer, 2016.

[21] Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao. A discriminative feature learning approach for deep face recognition. In *European conference on computer vision*, pages 499–515. Springer, 2016.

[22] Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 3485–3492. IEEE, 2010.

[23] 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.

[24] Xiao Zhang, Zhiyuan Fang, Yandong Wen, Zhifeng Li, and Yu Qiao. Range loss for deep face recognition with long-tailed training data. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5409–5418, 2017.

[25] 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.