Balanced Product of Experts for Long-Tailed Recognition

Emanuel Sanchez Aimar[1], **Arvi Jonnarth**[1,*], **Michael Felsberg**[1,†], **Marco Kuhlmann**[2]
[1]Department of Electrical Engineering, Linköping University, Sweden
[2]Department of Computer and Information Science, Linköping University, Sweden
{emanuel.sanchez.aimar,arvi.jonnarth,
michael.felsberg,marco.kuhlmann}@liu.se

Abstract

Many real-world recognition problems suffer from an imbalanced or long-tailed label distribution. Those distributions make representation learning more challenging due to limited generalization over the tail classes. If the test distribution differs from the training distribution, e.g. uniform versus long-tailed, the problem of the distribution shift needs to be addressed. To this aim, recent works have extended softmax cross-entropy using margin modifications, inspired by Bayes' theorem. In this paper, we generalize several approaches with a Balanced Product of Experts (BalPoE), which combines a family of models with different test-time target distributions to tackle the imbalance in the data. The proposed experts are trained in a single stage, either jointly or independently, and fused seamlessly into a BalPoE. We show that BalPoE is Fisher consistent for minimizing the balanced error and perform extensive experiments to validate the effectiveness of our approach. Finally, we investigate the effect of Mixup in this setting, discovering that regularization is a key ingredient for learning calibrated experts. Our experiments show that a regularized BalPoE can perform remarkably well in test accuracy and calibration metrics, leading to state-of-the-art results on CIFAR-100-LT, ImageNet-LT, and iNaturalist-2018 datasets. The code will be made publicly available upon paper acceptance.

1 Introduction

Recent developments within the field of deep learning, enabled by large-scale datasets and vast computational resources, have significantly contributed to the progress in many computer vision tasks (Krizhevsky et al., 2012). However, there is a discrepancy between common evaluation protocols in benchmark datasets and the desired outcome for real-world problems. Many benchmark datasets assume a balanced label distribution with an enough number of samples per class. In this setting, empirical risk minimization (ERM) has been widely adopted to solve multi-class classification. Unfortunately, ERM is not well suited for imbalanced, or long-tailed (LT), datasets, in which *head classes* have many more samples than *tail classes*, and where an unbiased predictor is desired at test time. This is not an uncommon scenario for real-world problems in a variety of different domains, such as object detection (Lin et al., 2014), medical diagnosis (Grzymala-Busse et al., 2004) and fraud detection (Philip & Chan, 1998). On the one hand, extreme class imbalance biases the classifier towards head classes (Kang et al., 2019; Tang et al., 2020). On the other hand, the scarcity of data samples hinders learning good representations for less represented classes, especially in few-shot

[*]Affiliation: Husqvarna Group, Huskvarna, Sweden.
[†]Co-affiliation: University of KwaZulu-Natal, Durban, South Africa.

Preprint. Under review.

data regimes (Ye et al., 2020). Addressing class imbalance is also relevant from the perspective of algorithmic fairness, since incorporating unfair biases into the models can have life-changing consequences in real-world decision-making systems (Mehrabi et al., 2021).

Previous works have approached the problem of class imbalance by different means, including *data re-sampling* (Chawla et al., 2002; Buda et al., 2018), *cost-sensitive learning* (Xie & Manski, 1989; Akbani et al., 2004), and *margin modifications* (Cao et al., 2019; Tan et al., 2020). While intuitive, several of these methods are not without limitations. Over-sampling can lead to overfitting of rare classes (Cui et al., 2019), while under-sampling common classes might hinder feature learning due to the omitting of valuable information (Kang et al., 2019), and loss re-weighting can result in optimization instability (Cao et al., 2019). Complementary to this line of research, ensemble learning has empirically shown benefits over single-expert models in terms of generalization (Kuncheva, 2014) and predictive uncertainty (Lakshminarayanan et al., 2017) on balanced datasets. Recently, expert-based approaches (Wang et al., 2020; Cai et al., 2021) also show promising performance gains in the long-tailed setting, although model calibration is still an unexplored aspect of LT ensembles.

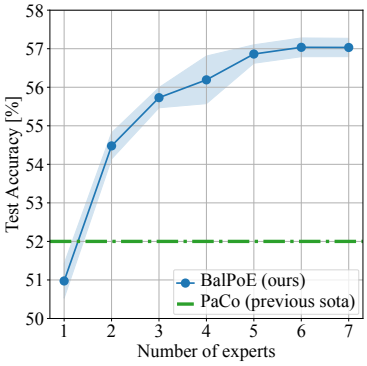


Figure 1: Test accuracy of BalPoE versus number of experts compared to PaCo (Cui et al., 2021) (currently best published method) on CIFAR-100-LT with an imbalance ratio of 100. Mean and standard deviation for BalPoE computed over 5 runs.

In this work, we investigate the following question: *In the presence of a skewed class distribution, how can we learn an ensemble classifier that is balanced, in the sense that it is Fisher consistent for minimizing the balanced error (Lin, 2002), and well-calibrated, in the sense that its predictions are reasonable probability estimates?* Our answer to this question is a new method that we call **Bal**anced **P**roduct **o**f **E**xperts (BalPoE).

Contributions. We extend the theoretical background for logit adjustment such that it can be applied to learning balanced expert ensembles. We derive a constraint for the target distributions, defined in terms of expert-specific biases, and prove that fulfilling the constraint yields consistency with the balanced error. We support our claim with extensive empirical studies, where we find that defining the ensemble in accordance with the constraint works very well in practice. Figure 1 shows that our balanced product of experts significantly improves over the previous best published method PaCo (Cui et al., 2021). Furthermore, we find that our expert ensembles are well calibrated, and investigate how this is affected by different target distributions and Mixup (Zhang et al., 2018). Our contributions can be summarized as follows:

- We extend the notion of logit adjustment based on label frequencies to learning balanced expert ensembles. We show that our approach is theoretically sound by proving that the expert ensemble is *Fisher consistent* for minimizing the balanced error.
- We conduct experiments investigating the effect of different expert target distributions and Mixup. We find that our method has stronger model performance and calibration compared to previous works and reaches a new state-of-the-art on long-tailed visual recognition.

2 Related work

Data resampling and loss re-weighting. Many approaches in literature resort to resampling the data or re-weighting the losses to achieve a more balanced distribution. The former includes under-sampling the majority classes (Drummond et al., 2003), over-sampling the minority classes (Chawla et al., 2002; Han et al., 2005), and class-balanced sampling (Huang et al., 2016; Wang et al., 2017). Among the latter approaches, some are based on class-level weights, e.g. proportional to the inverse class frequency (Huang et al., 2016) or by considering the *effective number of samples per class* (Cui et al., 2019), whereas other approaches propose sample-level weights (Lin et al., 2017).

Margin modifications. Enforcing large margins for minority classes has been shown to be an effective regularizer (Cao et al., 2019), improving model generalization under class imbalance (Tan et al., 2020; Ren et al., 2020). Analogously, recent *posthoc logit adjustments* can be seen as changing the class margins during inference time to favor tail classes (Menon et al., 2021; Tang et al., 2020).

Particularly, the approach proposed by Menon et al. (2021) is shown to be *Fisher-consistent* (Lin, 2002) for minimizing the balanced error. In addition, Hong et al. (2021) extend the previous approach to handle an arbitrary (known) test distribution.

Calibration. Modern over-parameterized neural networks are notorious for predicting uncalibrated probability estimates (Guo et al., 2017), being wrongly overconfident in the presence of out-of-distribution data (Li & Hoiem, 2020). These issues are further exacerbated under class imbalance (Zhong et al., 2021). Mixup (Zhang et al., 2018) and its extensions (Verma et al., 2019; Yun et al., 2019) have been shown very effective for improving model calibration (Thulasidasan et al., 2019) and generalization in the presence of balanced datasets. However, Mixup does not change the prior class distribution (Xu et al., 2021), thus other approaches propose modifications to the sampling procedure (Xu et al., 2021) and mixing strategy (Xu et al., 2021; Chou et al., 2020) for boosting the performance on tail classes.

Ensemble learning. The ensemble of multiple experts, such as Mixture of Experts (MoE) (Jacobs et al., 1991; Jordan & Jacobs, 1994), or Product of Experts (PoE) (Hinton, 2002), have empirically shown stronger generalization over its single-expert counterpart in the balanced setting (Szegedy et al., 2015; Kurutach et al., 2018). In addition, *deep ensembles* (Lakshminarayanan et al., 2017) and its extensions significantly improve calibration. These properties are typically attributed to model diversity, e.g. as a result of *making diverse mistakes* (Dietterich, 2000) or by exploring different local minima during training (Fort et al., 2019). In the long-tailed setting, expert models show promising results to improve the *head–tail trade-off* (Cai et al., 2021). Previous approaches typically enforce diversity with a joint optimization objective (Wang et al., 2020) or by learning on increasingly smaller data subsets (Xiang et al., 2020; Cai et al., 2021). However, the former are usually not tailored to addressing the head bias, whereas the latter present limited scalability due to the lack of data available for training tail specialists. Finally, Zhang et al. (2021) propose to tackle any unknown test distribution by first learning a fixed set of three diverse experts and then re-weighting them during inference time with a self-supervised procedure.

3 Preliminaries

In this section we present the theoretical background for our balanced product of experts, including a description of long-tailed recognition in Section 3.1, the motivation for *logit adjustment* from a distribution-shift perspective in Section 3.2, and a brief introduction to Mixup in Section 3.3.

3.1 Problem definition

In a multi-class classification problem, we are given an unknown distribution $\mathbb{P}$ over some data space $\mathcal{X} \times \mathcal{Y}$ of instances $\mathcal{X}$ and labels $\mathcal{Y} = \{1, 2, \dots, C\}$, and we want to find a mapping $f : \mathcal{X} \to \mathbb{R}^C$ that minimizes the misclassification error $\mathbb{P}_{x,y}\left(y \neq \arg\max_{j \in \mathcal{Y}} f_j(x)\right)$, where we denote $f(x) \equiv [f_1(x), ..., f_C(x)]$. In practice, given a sample $\mathcal{D} = \{x_i, y_i\}_{i=1}^N \sim \mathbb{P}$, we intend to minimize the empirical risk (Vapnik, 1991), $R_\delta(f) = \frac{1}{N} \sum_{i=1}^N \ell_{0/1}(y_i, f(x_i))$, where n_j is the number of samples for class j, $N = \sum_{j=1}^C n_j$ and $\ell_{0/1}$ denotes the per-sample misclassification error, also known as $0/1$-loss. As $\ell_{0/1}$ is not differentiable, it is typically approximated by the *softmax cross-entropy* (CE) criterion,

$$\ell(y, f(x)) = -\log \frac{e^{f_y(x)}}{\sum_{j \in \mathcal{Y}} e^{f_j(x)}} = \log \left[1 + \sum_{j \neq y} e^{f_j(x) - f_y(x)}\right]. \tag{1}$$

As observed by Hong et al. (2021), by minimizing the misclassification error (or its surrogate losses) in a long-tailed training set, we cannot disentangle the training prior from the data likelihood. For the rest of the paper, we assume a *label distribution shift* (Hong et al., 2021), s.t. the priors can be different, yet the likelihood remains unchanged, i.e.

$$\mathbb{P}^{\text{train}}(y) \neq \mathbb{P}^{\text{test}}(y) \qquad \mathbb{P}^{\text{train}}(x|y) = \mathbb{P}^{\text{test}}(x|y) \equiv \mathbb{P}(x|y). \tag{2}$$

When all classes are equally relevant, i.e. $\mathbb{P}^{\text{test}}(y) = \frac{1}{C} \equiv \mathbb{P}^{\text{bal}}(y)$, a more natural metric is the balanced error (BER), defined as $\text{BER}(f) \equiv \frac{1}{C} \sum_{y \in \mathcal{Y}} \mathbb{P}_{x|y}\left(y \neq \arg\max_{j \in \mathcal{Y}} f_j(x)\right)$ (Chan & Stolfo, 1998; Brodersen et al., 2010), which is consistent with minimizing the misclassification error under the uniform distribution.

3.2 Logit adjustment

From a statistical perspective, note that under a label distribution shift (2), we can write

$$\mathbb{P}^{\text{train}}(y|x) = \mathbb{P}^{\text{train}}(x|y)\frac{\mathbb{P}^{\text{train}}(y)}{\mathbb{P}^{\text{train}}(x)} \propto \mathbb{P}(x|y)\mathbb{P}^{\text{train}}(y), \tag{3}$$

which indicates that the training bias can be accounted for by a simple logit adjustment (Menon et al., 2021)

$$\frac{e^{f_y(x)}}{\mathbb{P}^{\text{train}}(y)} \propto \frac{\mathbb{P}^{\text{train}}(y|x)}{\mathbb{P}^{\text{train}}(y)} \propto \mathbb{P}(x|y) \propto \mathbb{P}^{\text{bal}}(y|x), \tag{4}$$

thus obtaining an *adjusted scorer* that is *Fisher consistent* for minimizing the balanced error (Menon et al., 2021). Hong et al. (2021) further generalize this idea to address an arbitrary label distribution shift, by observing we can swap the training prior for a desired test prior by

$$e^{f_y(x)}\frac{\mathbb{P}^{\text{test}}(y)}{\mathbb{P}^{\text{train}}(y)} \propto \mathbb{P}^{\text{train}}(y|x)\frac{\mathbb{P}^{\text{test}}(y)}{\mathbb{P}^{\text{train}}(y)} \propto \mathbb{P}(x|y)\mathbb{P}^{\text{test}}(y) \propto \mathbb{P}^{\text{test}}(y|x). \tag{5}$$

Finally, Xu et al. (2021) extend the logit adjusted loss (Menon et al., 2021) to accommodate the test distribution during training, by minimizing

$$\ell(y, f(x)) = -\log \frac{e^{f_y(x)+\log\frac{\mathbb{P}^{\text{train}}(y)}{\mathbb{P}^{\text{test}}(y)}}}{\sum_{j\in\mathcal{Y}} e^{f_j(x)+\log\frac{\mathbb{P}^{\text{train}}(j)}{\mathbb{P}^{\text{test}}(j)}}}. \tag{6}$$

3.3 Mixup

Previous works have observed overconfidence in over-parameterized neural networks, leading to poor calibration (Guo et al., 2017; Li & Hoiem, 2020). Although Mixup (Zhang et al., 2018) was introduced to alleviate memorization for over-parameterized models (Arpit et al., 2017), it was later shown to improve calibration in the balanced setting (Thulasidasan et al., 2019). Without loss of generality, let us denote $y \in \mathbb{R}^C$ as the *one-hot parameterization* of the corresponding label. Briefly, with Mixup we sample from a *vicinity distribution* $\mathcal{D}_\nu = \{\tilde{x}_i, \tilde{y}_i\}_{i=1}^{N'} \sim \mathbb{P}_\nu$, by creating virtual samples $(\tilde{x}, \tilde{y})$, where

$$\tilde{x} = \xi x_i + (1-\xi)x_j \qquad \tilde{y} = \xi y_i + (1-\xi)y_j \tag{7}$$

are convex combinations of random pairs $\{(x_i, y_i), (x_j, y_j)\} \sim \mathbb{P}^{\text{train}}_{x,y}$, $\xi \sim Beta(\alpha, \alpha)$ with $\alpha \in (0, \infty)$. Then, the model is trained by minimizing $R_\nu(f) = \frac{1}{N'}\sum_{i=1}^{N'} \ell(\tilde{y}_i, f(\tilde{x}_i))$, an approach known as *Vicinal Risk Minimization* (VRM).

4 Balanced product of experts

We will now introduce our main theoretical contribution. Motivated by the distribution shift perspective, let us revisit the logit adjustment formulation in Menon et al. (2021). For $\boldsymbol{\tau} \in \mathbb{R}^C$, note that

$$\arg\max_y f_y(x) - \boldsymbol{\tau}_y \log \mathbb{P}^{\text{train}}(y) = \arg\max_y f_y(x) - \log \mathbb{P}^{\text{train}}(y) + \log \mathbb{P}^{\text{train}}(y)^{1-\boldsymbol{\tau}_y} \tag{8}$$

$$= \arg\max_y f_y(x) - \log \mathbb{P}^{\text{train}}(y) + \log \frac{\mathbb{P}^{\text{train}}(y)^{1-\boldsymbol{\tau}_y}}{\sum_{j\in\mathcal{Y}} \mathbb{P}^{\text{train}}(y)^{1-\boldsymbol{\tau}_j}} \tag{9}$$

Let us denote $\boldsymbol{\lambda}_y = 1 - \boldsymbol{\tau}_y$ and $\mathbb{P}^{\text{test},\boldsymbol{\lambda}}(y) \equiv \frac{\mathbb{P}^{\text{train}}(y)^{\boldsymbol{\lambda}_y}}{\sum_{j\in\mathcal{Y}} \mathbb{P}^{\text{train}}(j)^{\boldsymbol{\lambda}_j}}$ for $\boldsymbol{\lambda} \in \mathbb{R}^C$. We observe that performing a logit adjustment with (8) can be interpreted as performing a distribution shift parameterized by $\boldsymbol{\lambda}$. In other words, the model shall accommodate for a target distribution $\mathbb{P}^{\text{test},\boldsymbol{\lambda}}(y)$. Thus, we define the *generalized logit adjusted* (gLA) loss as

$$\ell_{\boldsymbol{\tau}}(y, f(x)) = -\log \frac{e^{f_y(x)+\boldsymbol{\tau}_y \log \mathbb{P}^{\text{train}}(y)}}{\sum_{j\in\mathcal{Y}} e^{f_j(x)+\boldsymbol{\tau}_j \log \mathbb{P}^{\text{train}}(j)}} = \log\left[1 + \sum_{j\neq y} e^{f_j(x)-f_y(x)+\log\frac{\mathbb{P}^{\text{train}}(j)^{\boldsymbol{\tau}_j}}{\mathbb{P}^{\text{train}}(y)^{\boldsymbol{\tau}_y}}}\right] \tag{10}$$

where $\Delta_{yj}^{\boldsymbol{\tau}} = \log \frac{\mathbb{P}^{\text{train}}(j)^{\tau_j}}{\mathbb{P}^{\text{train}}(y)^{\tau_y}}$ defines a pairwise class margin. By training with $\ell_{\boldsymbol{\tau}}(y, f(x))$, the model is encouraged to incorporate the desired bias, as suggested by Xu et al. (2021). Intuitively, setting $\boldsymbol{\lambda} = \mathbb{1}$ ($\boldsymbol{\tau} = 0$) leads to the standard training procedure, whereas for $\boldsymbol{\lambda} = 0$ ($\boldsymbol{\tau} = \mathbb{1}$) we *model* the uniform distribution (Menon et al., 2021; Ren et al., 2020). Naturally, by increasing the pairwise margins further, e.g. for $\boldsymbol{\tau} = \mathbb{2}$ ($\boldsymbol{\lambda} = -\mathbb{1}$), the decision boundary is moved away from minority classes and towards majority classes, simulating a *reversed distribution*, as illustrated in Figure 2.

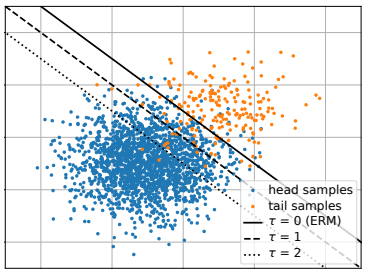


Figure 2: Illustration of how the margin moves towards the head class for increasing values of τ.

Following the previous observation, we raise the following question: *can we learn a stronger model, while still maintaining balancedness?* Inspired by *deep ensembles* (Lakshminarayanan et al., 2017), we propose *Balanced Product of Experts* to tackle long-tailed recognition. For our Balanced PoE, we combine a family of expert models, where each specializes in different test-time distributions, while the resultant PoE shall model the uniform distribution. Based on the distribution shift assumption (2), let us denote $\mathbb{P}^{\text{test},\boldsymbol{\lambda}}(x, y) \equiv \mathbb{P}(x|y)\mathbb{P}^{\text{test},\boldsymbol{\lambda}}(y)$. In the general case, for $M \geq 1$, we want to learn a set of expert scorers $\{f^{\boldsymbol{\lambda}}\}_{\boldsymbol{\lambda} \in S_\lambda}$ parameterized by $S_\lambda \in \mathbb{R}^{M \times C}$, s.t. $\exp f_y^{\boldsymbol{\lambda}}(x) \propto \mathbb{P}^{\text{test},\boldsymbol{\lambda}}(x, y)$. Then, we can define our (unnormalized) product of experts as

$$\overline{p}(x, y) \equiv \exp \overline{f}_y(x) = \prod_{\boldsymbol{\lambda} \in S_\lambda} \exp \left[f_y^{\boldsymbol{\lambda}}(x) \right]^{\frac{1}{M}}, \tag{11}$$

where $\overline{f}(x) \equiv \frac{1}{M} \sum_{\boldsymbol{\lambda} \in S_\lambda} f^{\boldsymbol{\lambda}}(x)$ denotes the *mean scorer*.

Let us denote $\overline{\boldsymbol{\lambda}} \equiv \frac{1}{M} \sum_{\boldsymbol{\lambda} \in S_\lambda}$. In Corollary 1.1 we show that $\overline{\boldsymbol{\lambda}} = 0$ is a sufficient condition to ensure that the mean scorer is *Fisher consistent* for minimizing the balanced error. Furthermore, Theorem 1 characterizes the distribution of the product of experts for any $\overline{\boldsymbol{\lambda}}$ and $M \geq 1$, providing a natural solution for learning individual experts (case $M = 1$). Hence, we propose to learn each $\boldsymbol{\lambda}$-expert by minimizing its respective $\ell_{1-\boldsymbol{\lambda}}$ loss. It then follows that the negative log-likelihood (NLL) for Balanced PoE can be approximated by (training experts independently and then) averaging the individual losses, i.e.

$$\ell_{1-\overline{\boldsymbol{\lambda}}}(y, \overline{f}(x)) \approx \frac{1}{M} \sum_{\boldsymbol{\lambda} \in S_\lambda} \ell_{1-\boldsymbol{\lambda}}(y, f^{\boldsymbol{\lambda}}(x)). \tag{12}$$

Theorem 1 (Distribution of Balanced PoE). *For $M \geq 1$ and $S_\lambda \in \mathbb{R}^{M \times C}$, assume sets of (unknown) scorer functions $\{f^{\boldsymbol{\lambda}}\}_{\boldsymbol{\lambda} \in S_\lambda}$ and $\{s^{\boldsymbol{\lambda}}\}_{\boldsymbol{\lambda} \in S_\lambda}$ with $f, s : \mathcal{X} \to \mathbb{R}^C$ s.t. $f_y^{\boldsymbol{\lambda}}(x) \equiv s_y^{\boldsymbol{\lambda}}(x) + (\boldsymbol{\lambda}_y - 1) \log \mathbb{P}^{\text{train}}(y)$ and $\mathbb{P}^{\text{train}}(y|x) \propto \exp s_y^{\boldsymbol{\lambda}}(x)$. Under the label distribution shift assumption in* (2)*, the product of experts as defined in* (11) *satisfies*

$$\overline{p}(x, y) \propto \mathbb{P}^{\text{test},\overline{\boldsymbol{\lambda}}}(x, y). \tag{13}$$

Proof. See supplementary material. □

Corollary 1.1 (Fisher consistency for Balanced PoE). *If $\overline{\boldsymbol{\lambda}} = 0$, then $\overline{f}$ is fisher-consistent for minimizing the balanced error.*

Proof. From Theorem 1, we have that for $\overline{\boldsymbol{\lambda}} = 0$,

$$\arg\max_y \overline{f}_y(x) = \arg\max_y \log \mathbb{P}(x|y)\mathbb{P}^{\text{test},\overline{\boldsymbol{\lambda}}}(y) = \arg\max_y \log \mathbb{P}(x|y)\frac{1}{C} = \arg\max_y \mathbb{P}(x|y). \tag{14}$$

Hence, it follows from (7) in Menon et al. (2021), that $\overline{f}_y(x)$ is *Bayes-optimal* for minimizing the balanced error. □

Interestingly, Corollary 1.1 does not assume any particular distribution for $\boldsymbol{\lambda}$'s, as long as the resultant PoE remains balanced. Moreover, note that our approach generalizes several works based on logit

adjustment (Menon et al., 2021; Hong et al., 2021; Xu et al., 2021) and can in principle accommodate for any known target distribution $\mathbb{P}^{\text{test}}(y)$, by setting $\overline{\boldsymbol{\lambda}}_y = \log \frac{\mathbb{P}^{\text{test}}(y)}{\mathbb{P}^{\text{train}}(y)}$. In our experiments, we assume a simplified setting where $\boldsymbol{\lambda}_i = \boldsymbol{\lambda}_j$ for $i \neq j$ (see Section 5). For the rest of the paper, we denote $\boldsymbol{\lambda}$ as λ and $\overline{\boldsymbol{\lambda}}$ as $\overline{\lambda}$.

5 Experiments

We validate our finding of Theorem 1 by measuring test accuracy when varying distinct aspects of the expert ensemble, including the number of experts, test-time target distributions in terms of λ, and backbone configuration. Furthermore, we measure model calibration of our expert ensemble and investigate how this is affected by Mixup. Finally, we compare our method with current state-of-the-art methods on several benchmark datasets.

5.1 Experimental setup

In this section, we describe our experiment setup, including the datasets used for long-tailed classification in Section 5.1.1, implementation details in Section 5.1.2, and evaluation protocol in Section 5.1.3.

5.1.1 Long-tailed datasets

CIFAR-100-LT. The original CIFAR-100 dataset (Krizhevsky, 2009) contains 60K images, 50K for training, and 10K for validation. It contains 100 categories and has a balanced number of samples per class. Following Cui et al. (2019); Cao et al. (2019), a long-tailed version, CIFAR-100-LT, is created by discarding samples according to exponential decay. The data imbalance is controlled by the imbalance ratio (IR) $\rho = \frac{\max_i n_i}{\min_i n_i}$, i.e. the ratio between the number of instances for the most populated and least populated classes. We conduct experiments with $\rho \in \{10, 50, 100\}$. For experiments on a long-tailed version of CIFAR-10 (Krizhevsky, 2009), see the supplementary material.

ImageNet-LT. Built from ImageNet-2012 (Deng et al., 2009) with 1K classes, a long-tailed version, ImageNet-LT, is obtained by sampling from a Pareto distribution with $\alpha = 6$ (Liu et al., 2019). The resultant dataset consists of 115.8K training, 20K validation, and 50K test images. The categories in the training set contain between 5 to 1280 samples, resulting in $\rho = 256$.

iNaturalist. The iNaturalist 2018 dataset (Horn et al., 2017) is a large-scale species classification dataset containing 437K images and 8142 classes in a hierarchical class structure of species. It was collected in a real-world setting and exhibits a natural imbalance ratio of $\rho = 500$.

5.1.2 Implementation details

For experiments with CIFAR, we follow the setup in Cao et al. (2019). We train a ResNet-32 backbone (He et al., 2016) for 200 epochs with the SGD, initial learning rate (LR) of 0.1, momentum rate set to 0.9, and a batch size of 128. A multi-step learning rate schedule decreases the LR by a factor of 10 at the 160th and 180th epochs. For ImageNet-LT and iNaturalist, we use a ResNet-50 backbone (He et al., 2016) and train for 180 and 100 epochs, with batch sizes 128 and 512, respectively. For these datasets, we use a cosine annealing learning rate schedule (Loshchilov & Hutter, 2017), with initial LR of 0.025 and 0.2, respectively. Additionally, we conduct experiments with the long-training setup proposed by Cui et al. (2021). All models are trained for 400 epochs with stronger data augmentation, namely RandAugment (Cubuk et al., 2020) for ImageNet and iNaturalist, and AutoAugment (Cubuk et al., 2019) for CIFAR. In the latter case, LR is decreased at epochs 320 and 360. For Mixup experiments, we set parameter α to 0.4, 0.2, and 0.2 for CIFAR-LT, ImageNet-LT and Inaturalist, respectively, unless stated otherwise. For BalPoE with $M = 3$ experts, we set λ's to $\{1, 0, -1\}$, $\{1, -0.25, -1.5\}$ and $\{2, 0, -2\}$, for CIFAR-LT, ImageNet-LT and Inaturalist, respectively, unless stated otherwise.

5.1.3 Evaluation protocol

Following Cao et al. (2019), we report accuracy on a balanced test set. Whenever confidence intervals are shown, they are acquired from five runs. In addition, we also investigate how model performance is affected by the number of samples in the training data. To this end, we report accuracy for three splits,

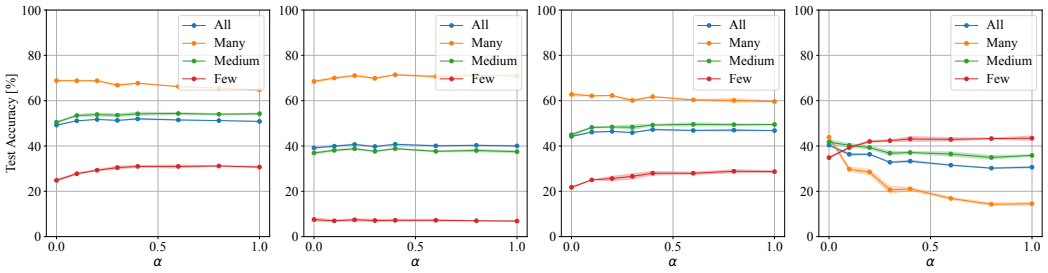


Figure 3: Test accuracy of a ResNet-32 trained on CIFAR-100-LT with IR=100, as a function of the Mixup parameter α, for (a) BalPoE, (b) head expert, (c) medium expert, and (d) tail expert.

where classes are grouped by number of samples, namely, many-shot (> 100 samples), medium-shot (20–100 samples), and few-shot (< 20 samples). Furthermore, we assess model calibration by estimating the expected calibration error (ECE), maximum calibration error (MCE) and visualizing reliability diagrams. See the supplementary material for definitions of these metrics.

5.2 Results and analysis

How does Mixup regularization affect expert specialization and ensemble performance? We hypothesize that training our BalPoE with ERM might still lead to uneven levels of overconfidence for different experts, thus affecting individual contribution in the overall ensemble. To verify this hypothesis, we evaluate the effect of Mixup regularization for long-trained learning by varying the parameter α. We train BalPoE with $\lambda's = \{1, 0, -1\}$ on CIFAR-100 with IR$= 100$. In Figure 3, we observe that Mixup promotes further expert specialization, especially for the tail expert which becomes a specialist at few-shot performance over other data regimes. Regularizing individual experts boosts the performance for the ensemble, reaching the highest performance around $\alpha = 0.2$-0.4. This is consistent with the study of Mixup under the balanced setting (Zhang et al., 2018), and previous findings suggesting that large α values lead to underfitting, due to *manifold intrusion* (Guo et al., 2019).

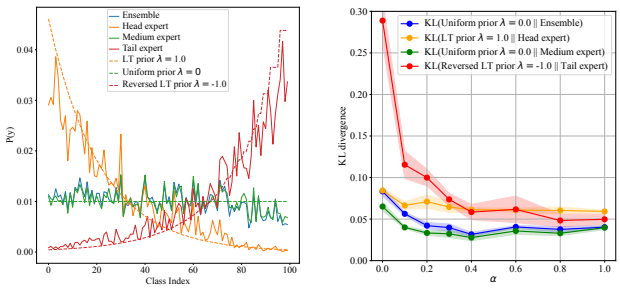
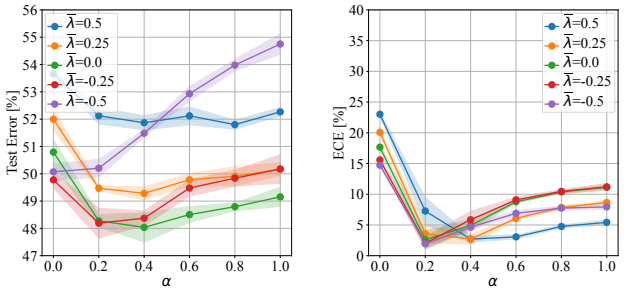


Figure 4: (a) Test priors against expected confidence, computed over CIFAR100-IR100 test split. (b) KL divergence of test priors against expected confidence, computed over CIFAR100-IR100 test split.

In Figure 4a we analyse the expert-specific biases for the different regularized λ-experts, where the λ-expert *test prior* is estimated by averaging predictive confidences over the balanced test data. This is compared to their *expected* test prior distributions $\mathbb{P}^{\text{test},\lambda}(y)$, whereas the prior of the ensemble is compared to the uniform distribution. We observe that the learned and expected priors match nicely.

Furthermore, we compute the Kullback-Leibler divergence (KL) between learned and expected priors as a function of α, see Figure 4b. We find that increasing regularization decreases the divergence for all experts, as well as for the ensemble up to $\alpha = 0.4$, where the divergence attends its minimum for the uniform distribution. This suggests that the regularized experts are both more specialized, and better calibrated for modelling their respective distributions. As expected, by setting $\overline{\lambda} = 0$ the ensemble becomes better at the balanced distribution, which is consistent with the observations in Figure 3.

Does our product of experts need to be balanced? To verify the validity of Corollary 1.1 in practical settings, we train our product of experts by varying the average bias $\overline{\lambda}$ and report the test error in Figure 5a. As expected, the optimal choice for $\overline{\lambda}$ is close to 0 and coincides exactly for



(a) (b)

Figure 5: (a) Test error and (b) expected calibration error (ECE) on the test split of CIFAR-100-LT, IR=100.

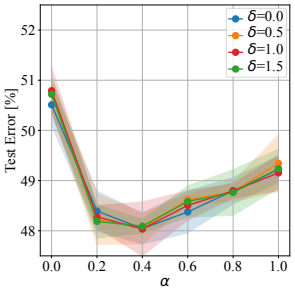


Figure 6: Test error when varying model diversity, defined as the spread of λ.

well-regularized models. For $\alpha = 0$, we hypothesize that the reason for the optimal $\overline{\lambda}$ not being 0, but instead -0.25, is due to violations of the assumptions made in Theorem 1, namely the *label distribution shift assumption*, as suggested by the low specialization and high KL divergence of the tail specialist, seen in Figures 3d and 4b, respectively.

Effect of λ distribution. Corollary 1.1 indicates that our proposed approach is Fisher consistent for minimizing the balanced error if $\overline{\lambda}$ is equal to 0, but it does not make assumptions on the distribution of λ's across different experts. We explore this aspect by varying the spread of λ's within the ensemble while keeping $\overline{\lambda} = 0$ fixed. Concretely, we train different ensemble models (with $M = 3$) by setting $\lambda' s = \{+\delta, 0, -\delta\}$, where δ controls the spread of aforementioned distribution. We perform experiments for $\delta \in \{0, 0.5, 1, 1.5\}$. Note that $\delta = 0$ corresponds to $\lambda = 0$ for all of experts, i.e. the individual experts all consider a uniform test-time target distribution. For $\delta = 1$, the ensemble includes a LT expert with $\lambda = 1$ (equivalent to biased training). Although unintuitive, we increase δ even further, thus biasing the experts to target test-time distributions that are more skewed than the original training prior. The test error as a function of α is presented in Figure 6. In the general case, we observe small variations in mean performance for different δ's, and no significant differences for the optimal $\alpha^* = 0.4$. This result empirically verifies that *balancedness* is the most relevant factor in our approach. Considering the robust performance with $\delta = 0$, we hypothesize that the main source of diversity must come from different (random) model initializations, leading to the exploration of diverse local minima in the loss landscape, as found by Fort et al. (2019).

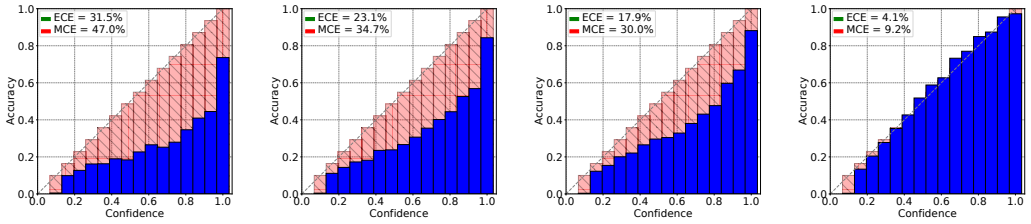
The performance of BalPoE consistently improves for an increasing number of experts. In this section, we investigate the scalability of our approach, see Figure 1. For simplicity, we set different λ's equidistantly spaced, with minimum and maximum values at -1.5 and 1.5 respectively, and $\lambda = 0$ for the single-model case. This ensures that the average is $\overline{\lambda} = 0$. We observe a graceful increase of test accuracy as more experts become part of the ensemble. Interestingly, the performance for even-numbered ensembles interpolates seamlessly, which indicates that a *uniform specialist* (with $\lambda = 0$) is not strictly required, as long as we maintain a balanced product of experts. We find similar improvements for the few-, medium-, and many-shot regimes separately, see the supplemental material for more details, where we also find minor improvements using independent backbones.

5.3 Comparison with state-of-the-art

We compare BalPoE with previous long-tailed approaches in Table 1. The results for ensemble methods, including BalPoE, are reported for three experts. For the standard setting (*short training*), we observe considerable improvements over previous methods. For CIFAR-100-LT we gain +1.5, +2.4, and +2.2 points over the previous state-of-the-art for imbalance ratios 10, 50, and 100, respectively. On ImageNet-LT we match the best previous method, and on iNaturalist we improve by +2.1 points. Finally, under the *long training* protocol we reach similar levels of improvement as in the standard setting. Moreover, as shown in Figure 1, the generalization performance can be further improved by adding more experts. Following this strategy, our approach achieves a peak at 57% test accuracy for an ensemble of seven experts on CIFAR-100-LT with IR=100.

Table 1: Test accuracy (%) on CIFAR-100-LT, ImageNet-LT, and iNaturalist 2018, for different imbalance ratios (IR), and backbones; Res=ResNet, ResX=ResNeXt. ⋆: Our reproduced results. †: From Xu et al. (2021). ‡: From Zhang et al. (2021). §: From Zhou et al. (2020).

		CIFAR-100-LT			ImageNet-LT		iNaturalist
	Backbone →	Res32	Res32	Res32	Res50	ResX50	Res50
Method ↓	IR →	10	50	100	256	256	500
Short training							
CE⋆		57.2	43.9	38.8	47.2	48.0‡	65.2
LDAM-DRW (Cao et al., 2019)		58.7	48.0†	42.0	45.8†	-	68.0
CB-Focal (Cui et al., 2019)		58.0	45.3	39.6	-	-	64.2
LA (Menon et al., 2021)		59.9†	49.8†	43.9	51.1	-	68.4
LADE (Hong et al., 2021)		61.7	50.5	45.4	-	53.0	70.0
Mixup§ (Zhang et al., 2018)		58.0	45.0	39.5	-	-	-
Remix-DRW (Chou et al., 2020)		61.2	-	46.8	-	-	70.5
MiSLAS (Zhong et al., 2021)		63.2	52.3	47.0	52.7	-	71.6
UniMix+Bayias (Xu et al., 2021)		61.3	51.1	45.5	48.4	-	69.2
RIDE (Wang et al., 2020)		61.8‡	51.7‡	48.0	54.9	56.4	72.2
ACE (Cai et al., 2021)		-	51.9	49.6	54.7	56.6	72.9
TADE (Zhang et al., 2021)		63.6	53.9	49.8	-	**58.8**	72.9
BalPoE (ours)		65.0	54.3	49.2	56.8	56.7	74.8
BalPoE + Mixup α^* (ours)		**65.1**	**56.3**	**52.0**	**57.9**	**58.8**	**75.0**
Long training							
PaCo (Cui et al., 2021)		64.2	56.0	52.0	57.0	58.2	73.2
TADE (Zhang et al., 2021)		65.3	57.3	52.2	-	**61.2**	74.5
BalPoE (ours)		**68.0**	**60.2**	**55.7**	**60.8**	61.0	**76.9**



(a) (b) (c) (d)

Figure 7: Reliability plots for (a) ERM with a single expert ($\overline{\lambda} = 1$), (b) LA loss with a single expert ($\overline{\lambda} = 0$), (c) BalPoE with three experts (λ's $= \{1, 0, -1\}$, $\overline{\lambda} = 0$), and (d) same BalPoE with Mixup ($\alpha = 0.4$).

5.4 *Is the balanced product of experts well-calibrated?*

Preferably, the confidence of a model reflects the true probability of the predicted class being the correct classification, i.e. the model is calibrated. As shown in the reliability plots in Figure 7, the ECE for a single model trained with ERM is 31.5%, which is reduced to 23.1% with the logit adjusted loss ($\lambda = 0$), and further reduced to 17.9% and 4.1% with a BalPoE of 3 experts ($\lambda's = \{1, 0, -1\}$) and Mixup, sequentially. We conclude that Mixup improves calibration significantly for our model, which we argue explains the large performance gain using it. We further investigate Mixup's effect on calibration in Figure 5b, where we provide additional evidence for calibration improvements. In Figures 5b and 5a, we observe a strong correlation between calibration and misclassification errors. This further supports our hypothesis: *calibration is a key ingredient for maximizing ensemble performance*. Since each expert is better calibrated, a fair weighting of individual *expert* can lead to better decisions by the overall ensemble.

6 Concluding discussion

In this paper, we extend the theoretical foundation for logit adjustment to be used for training a balanced product of experts (BalPoE). We show that the ensemble is consistent with the balanced error, given that a constraint for the expert-specific biases is fulfilled. We find that model calibration is important for ensemble performance since the experts need to be weighed against each other in a proper way. This is achieved with Mixup regularization. Our BalPoE reaches state-of-the-art results on three long-tailed benchmark datasets.

Limitations. First, we assume $\mathbb{P}^{\text{train}}(x|y) = \mathbb{P}^{\text{test}}(x|y)$, which is a fair assumption but may be violated in practice, e.g. in autonomous driving applications, where the model might be exposed to out-of-distribution data. Second, the prior $\mathbb{P}^{\text{train}}(y)$ is estimated empirically based on the number of training samples, which can be suboptimal for few-shot classes. To address this issue, considering the effective number of samples (Cui et al., 2019) could be an interesting venue for future research. Finally, our findings are limited to single-label multi-class classification, extending our balanced product of experts to multi-label classification or other detection tasks is left to future work.

7 Acknowledgements

This work was partially supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation. The computations were enabled by resources provided by the Swedish National Infrastructure for Computing (SNIC), partially funded by the Swedish Research Council (VR) through grant agreement no. 2018-05973.

References

Akbani, R., Kwek, S., and Japkowicz, N. Applying support vector machines to imbalanced datasets. In *European conference on machine learning*, pp. 39–50. Springer, 2004. 2

Arpit, D., Jastrzębski, S., Ballas, N., Krueger, D., Bengio, E., Kanwal, M. S., Maharaj, T., Fischer, A., Courville, A., Bengio, Y., et al. A closer look at memorization in deep networks. In *International conference on machine learning*, pp. 233–242. PMLR, 2017. 4

Brodersen, K. H., Ong, C. S., Stephan, K. E., and Buhmann, J. M. The balanced accuracy and its posterior distribution. In *2010 20th international conference on pattern recognition*, pp. 3121–3124. IEEE, 2010. 3

Buda, M., Maki, A., and Mazurowski, M. A. A systematic study of the class imbalance problem in convolutional neural networks. *Neural Networks*, 106:249–259, 2018. 2

Cai, J., Wang, Y., and Hwang, J.-N. Ace: Ally complementary experts for solving long-tailed recognition in one-shot. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 112–121, 2021. 2, 3, 9, 15, 16, 18

Cao, K., Wei, C., Gaidon, A., Arechiga, N., and Ma, T. Learning imbalanced datasets with label-distribution-aware margin loss. *Advances in Neural Information Processing Systems*, 32:1567–1578, 2019. 2, 6, 9, 15, 18

Chan, P. K. and Stolfo, S. J. Learning with non-uniform class and cost distributions: Effects and a distributed multi-classifier approach. In *In Workshop Notes KDD-98 Workshop on Distributed Data Mining*. Citeseer, 1998. 3

Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357, 2002. 2

Chou, H.-P., Chang, S.-C., Pan, J.-Y., Wei, W., and Juan, D.-C. Remix: rebalanced mixup. In *European Conference on Computer Vision*, pp. 95–110. Springer, 2020. 3, 9, 18, 19

Cubuk, E. D., Zoph, B., Mane, D., Vasudevan, V., and Le, Q. V. Autoaugment: Learning augmentation strategies from data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 6, 15

Cubuk, E. D., Zoph, B., Shlens, J., and Le, Q. V. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 702–703, 2020. 6

Cui, J., Zhong, Z., Liu, S., Yu, B., and Jia, J. Parametric contrastive learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 715–724, 2021. 2, 6, 9, 16, 17

Cui, Y., Jia, M., Lin, T.-Y., Song, Y., and Belongie, S. Class-balanced loss based on effective number of samples. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9268–9277, 2019. 2, 6, 9, 10, 18

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009. 6

Dietterich, T. G. Ensemble methods in machine learning. In *International workshop on multiple classifier systems*, pp. 1–15. Springer, 2000. 3, 16

Drummond, C., Holte, R. C., et al. C4. 5, class imbalance, and cost sensitivity: why under-sampling beats over-sampling. In *Workshop on learning from imbalanced datasets II*, volume 11, pp. 1–8. Citeseer, 2003. 2

Fort, S., Hu, H., and Lakshminarayanan, B. Deep ensembles: A loss landscape perspective. *arXiv preprint arXiv:1912.02757*, 2019. 3, 8, 16

Grzymala-Busse, J. W., Goodwin, L. K., Grzymala-Busse, W. J., and Zheng, X. An approach to imbalanced data sets based on changing rule strength. In *Rough-neural computing*, pp. 543–553. Springer, 2004. 1

Guo, C., Pleiss, G., Sun, Y., and Weinberger, K. Q. On calibration of modern neural networks. In *International Conference on Machine Learning*, pp. 1321–1330. PMLR, 2017. 3, 4, 17

Guo, H., Mao, Y., and Zhang, R. Mixup as locally linear out-of-manifold regularization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 3714–3722, 2019. 7

Han, H., Wang, W.-Y., and Mao, B.-H. Borderline-smote: a new over-sampling method in imbalanced data sets learning. In *International conference on intelligent computing*, pp. 878–887. Springer, 2005. 2

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016. 6

Hinton, G. E. Training products of experts by minimizing contrastive divergence. *Neural computation*, 14(8):1771–1800, 2002. 3

Hong, Y., Han, S., Choi, K., Seo, S., Kim, B., and Chang, B. Disentangling label distribution for long-tailed visual recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6626–6636, 2021. 3, 4, 6, 9, 16, 17

Horn, G. V., Aodha, O. M., Song, Y., Cui, Y., Sun, C., Shepard, A., Adam, H., Perona, P., and Belongie, S. The inaturalist species classification and detection dataset. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 8769–8778, 7 2017. ISSN 10636919. doi: 10.48550/arxiv.1707.06642. URL `https://arxiv.org/abs/1707.06642v2`. 6

Huang, C., Li, Y., Loy, C. C., and Tang, X. Learning deep representation for imbalanced classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5375–5384, 2016. 2

Jacobs, R. A., Jordan, M. I., Nowlan, S. J., and Hinton, G. E. Adaptive mixtures of local experts. *Neural computation*, 3(1):79–87, 1991. 3

Jordan, M. I. and Jacobs, R. A. Hierarchical mixtures of experts and the em algorithm. *Neural computation*, 6(2):181–214, 1994. 3

Kang, B., Xie, S., Rohrbach, M., Yan, Z., Gordo, A., Feng, J., and Kalantidis, Y. Decoupling representation and classifier for long-tailed recognition. In *International Conference on Learning Representations*, 2019. 1, 2, 19

Krizhevsky, A. Learning multiple layers of features from tiny images. Technical report, 2009. 6

Krizhevsky, A., Sutskever, I., and Hinton, G. E. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012. 1

Kuncheva, L. I. *Combining pattern classifiers: methods and algorithms*. John Wiley & Sons, 2014. 2

Kurutach, T., Clavera, I., Duan, Y., Tamar, A., and Abbeel, P. Model-ensemble trust-region policy optimization. In *International Conference on Learning Representations (ICLR)*, 2018. URL https://openreview.net/forum?id=SJJinbWRZ. 3

Lakshminarayanan, B., Pritzel, A., and Blundell, C. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in neural information processing systems*, 30, 2017. 2, 3, 5, 15

Li, Z. and Hoiem, D. Improving confidence estimates for unfamiliar examples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2686–2695, 2020. 3, 4

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*, pp. 740–755. Springer, 2014. 1

Lin, T.-Y., Goyal, P., Girshick, R., He, K., and Dollár, P. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pp. 2980–2988, 2017. 2

Lin, Y. A note on margin-based loss functions in classification by. 2002. 2, 3

Liu, Z., Miao, Z., Zhan, X., Wang, J., Gong, B., and Yu, S. X. Large-scale long-tailed recognition in an open world. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2537–2546, 2019. 6, 15

Loshchilov, I. and Hutter, F. SGDR: stochastic gradient descent with warm restarts. In *5th International Conference on Learning Representations, ICLR 2017*. OpenReview.net, 2017. URL https://openreview.net/forum?id=Skq89Scxx. 6

Mehrabi, N., Morstatter, F., Saxena, N., Lerman, K., and Galstyan, A. A survey on bias and fairness in machine learning. *ACM Computing Surveys (CSUR)*, 54(6):1–35, 2021. 2

Menon, A. K., Jayasumana, S., Rawat, A. S., Jain, H., Veit, A., and Kumar, S. Long-tail learning via logit adjustment. In *International Conference on Learning Representations (ICLR)*, 2021. URL https://openreview.net/forum?id=37nvvqkCo5. 2, 3, 4, 5, 6, 9, 18, 19

Philip, K. and Chan, S. Toward scalable learning with non-uniform class and cost distributions: A case study in credit card fraud detection. In *Proceeding of the Fourth International Conference on Knowledge Discovery and Data Mining*, pp. 164–168, 1998. 1

Ren, J., Yu, C., sheng, s., Ma, X., Zhao, H., Yi, S., and Li, h. Balanced meta-softmax for long-tailed visual recognition. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M. F., and Lin, H. (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 4175–4186. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/2ba61cc3a8f44143e1f2f13b2b729ab3-Paper.pdf. 2, 5, 16, 17

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1–9, 2015. 3

Tan, J., Wang, C., Li, B., Li, Q., Ouyang, W., Yin, C., and Yan, J. Equalization loss for long-tailed object recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11662–11671, 2020. 2

Tang, K., Huang, J., and Zhang, H. Long-tailed classification by keeping the good and removing the bad momentum causal effect. *Advances in Neural Information Processing Systems*, 33:1513–1524, 2020. 1, 2

Thulasidasan, S., Chennupati, G., Bilmes, J. A., Bhattacharya, T., and Michalak, S. On mixup training: Improved calibration and predictive uncertainty for deep neural networks. *Advances in Neural Information Processing Systems*, 32, 2019. 3, 4

Vapnik, V. Principles of risk minimization for learning theory. *Advances in neural information processing systems*, 4, 1991. 3

Verma, V., Lamb, A., Beckham, C., Najafi, A., Mitliagkas, I., Lopez-Paz, D., and Bengio, Y. Manifold mixup: Better representations by interpolating hidden states. In *International Conference on Machine Learning*, pp. 6438–6447. PMLR, 2019. 3

Wang, X., Lian, L., Miao, Z., Liu, Z., and Yu, S. Long-tailed recognition by routing diverse distribution-aware experts. In *International Conference on Learning Representations*, 2020. 2, 3, 9, 15, 16, 17, 18

Wang, Y.-X., Ramanan, D., and Hebert, M. Learning to model the tail. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pp. 7032–7042, 2017. 2

Xiang, L., Ding, G., and Han, J. Learning from multiple experts: Self-paced knowledge distillation for long-tailed classification. In *European Conference on Computer Vision*, pp. 247–263. Springer, 2020. 3

Xie, Y. and Manski, C. F. The logit model and response-based samples. *Sociological Methods & Research*, 17(3):283–302, 1989. 2

Xu, Z., Chai, Z., and Yuan, C. Towards calibrated model for long-tailed visual recognition from prior perspective. *Advances in Neural Information Processing Systems*, 34, 2021. 3, 4, 5, 6, 9, 18, 19

Ye, H.-J., Chen, H.-Y., Zhan, D.-C., and Chao, W.-L. Identifying and compensating for feature deviation in imbalanced deep learning. *arXiv preprint arXiv:2001.01385*, 2020. 2

Yun, S., Han, D., Oh, S. J., Chun, S., Choe, J., and Yoo, Y. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 6023–6032, 2019. 3

Zhang, H., Cisse, M., Dauphin, Y. N., and Lopez-Paz, D. mixup: Beyond empirical risk minimization. *International Conference on Learning Representations (ICLR)*, 2018. URL `https://openreview.net/forum?id=r1Ddp1-Rb`. 2, 3, 4, 7, 9, 18, 19

Zhang, Y., Hooi, B., Hong, L., and Feng, J. Test-agnostic long-tailed recognition by test-time aggregating diverse experts with self-supervision. *arXiv preprint arXiv:2107.09249*, 2021. 3, 9, 15, 16, 17, 18

Zhong, Z., Cui, J., Liu, S., and Jia, J. Improving calibration for long-tailed recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16489–16498, 2021. 3, 9, 16, 17, 18, 19

Zhou, B., Cui, Q., Wei, X.-S., and Chen, Z.-M. 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*, pp. 9719–9728, 2020. 9, 18

Supplementary Material for "Balanced Product of Experts for Long-Tailed Recognition"

A Missing proofs

Theorem 1 (Distribution of Balanced PoE)

For $M \geq 1$ and $S_\lambda \in \mathbb{R}^{M \times C}$, assume two sets of scorer functions, (unknown) $\{s^{\boldsymbol{\lambda}}\}_{\boldsymbol{\lambda} \in S_\lambda}$ and $\{f^{\boldsymbol{\lambda}}\}_{\boldsymbol{\lambda} \in S_\lambda}$ with $s^{\boldsymbol{\lambda}}, f^{\boldsymbol{\lambda}} : \mathcal{X} \to \mathbb{R}^C$, s.t.

$$f_y^{\boldsymbol{\lambda}}(x) \equiv s_y^{\boldsymbol{\lambda}}(x) + (\boldsymbol{\lambda}_y - 1) \log \mathbb{P}^{\text{train}}(y) \tag{15}$$

and

$$\mathbb{P}^{\text{train}}(y|x) \propto e^{s_y^{\boldsymbol{\lambda}}(x)}. \tag{16}$$

Then, under the label distribution shift assumption in (2), the (unnormalized) product of experts $\overline{p}(x, y)$ (11) satisfies

$$\overline{p}(x, y) \propto \mathbb{P}(x|y) \mathbb{P}^{\text{test}, \overline{\boldsymbol{\lambda}}}(y) \equiv \mathbb{P}^{\text{test}, \overline{\boldsymbol{\lambda}}}(x, y) \tag{17}$$

where $\overline{\boldsymbol{\lambda}} \equiv \frac{1}{M} \sum_{\boldsymbol{\lambda} \in S_\lambda} \boldsymbol{\lambda}$ and $\mathbb{P}^{\text{test}, \overline{\boldsymbol{\lambda}}}(y) \equiv \frac{\mathbb{P}^{\text{train}}(y)^{\overline{\boldsymbol{\lambda}}_y}}{\sum_{j \in \mathcal{Y}} \mathbb{P}^{\text{train}}(j)^{\overline{\boldsymbol{\lambda}}_j}}$.

Proof of Theorem 1 Without loss of generality, let us denote S_λ as the set of *row vectors* (with repetition) in the corresponding matrix. Given $x \in \mathcal{X}$ and $y \in \mathcal{Y}$, for each $\boldsymbol{\lambda} \in S_\lambda$ and its respective (training) scorer $s^{\boldsymbol{\lambda}}$, we have that

$$\mathbb{P}^{\text{train}}(y|x) = \frac{e^{s_y^{\boldsymbol{\lambda}}(x)}}{Z_x^{\boldsymbol{\lambda}}}, \qquad (16) \tag{18}$$

where $Z_x^{\boldsymbol{\lambda}} \in \mathbb{R}$ is an (unknown) normalizing factor. Then, our mean scorer $\overline{f}$ satisfies

$$\begin{aligned}
\overline{f}_y(x) &= \frac{1}{M} \sum_{\boldsymbol{\lambda} \in S_\lambda} \left[s_y^{\boldsymbol{\lambda}}(x) + (\boldsymbol{\lambda}_y - 1) \log \mathbb{P}^{\text{train}}(y) \right] && (11), (15) && (19) \\
&= \frac{1}{M} \sum_{\boldsymbol{\lambda} \in S_\lambda} s_y^{\boldsymbol{\lambda}}(x) + \left[\frac{1}{M} \sum_{\boldsymbol{\lambda} \in S_\lambda} \boldsymbol{\lambda}_y - 1 \right] \log \mathbb{P}^{\text{train}}(y) && && (20) \\
&= \frac{1}{M} \sum_{\boldsymbol{\lambda} \in S_\lambda} \log \left[\mathbb{P}^{\text{train}}(y|x) Z_x^{\boldsymbol{\lambda}} \right] + (\overline{\boldsymbol{\lambda}}_y - 1) \log \mathbb{P}^{\text{train}}(y) && (18) && (21) \\
&= \log \frac{\mathbb{P}^{\text{train}}(y|x)}{\mathbb{P}^{\text{train}}(y)} + \log \mathbb{P}^{\text{train}}(y)^{\overline{\boldsymbol{\lambda}}_y} + \frac{1}{M} \sum_{\boldsymbol{\lambda} \in S_\lambda} \log Z_x^{\boldsymbol{\lambda}} && && (22) \\
&= \log \mathbb{P}^{\text{train}}(x|y) + \log \mathbb{P}^{\text{test}, \overline{\boldsymbol{\lambda}}}(y) + \overline{C}_x^{\boldsymbol{\lambda}} && (3), (9), \text{(see definition of } \overline{C}_x^{\boldsymbol{\lambda}} \text{ below)} && (23) \\
&= \log \left[\mathbb{P}(x|y) \mathbb{P}^{\text{test}, \overline{\boldsymbol{\lambda}}}(y) \right] + \overline{C}_x^{\boldsymbol{\lambda}} && (2) && (24) \\
&= \log \mathbb{P}^{\text{test}, \overline{\boldsymbol{\lambda}}}(x, y) + \overline{C}_x^{\boldsymbol{\lambda}}, && (2) && (25)
\end{aligned}$$

where $\overline{C}_x^{\boldsymbol{\lambda}} = -\log \mathbb{P}^{\text{train}}(x) + \log \left[\sum_{j \in \mathcal{Y}} \mathbb{P}^{\text{train}}(j)^{\overline{\boldsymbol{\lambda}}_j} \right] + \frac{1}{M} \sum_{\boldsymbol{\lambda} \in S_\lambda} \log Z_x^{\boldsymbol{\lambda}}$ hides terms which are constant w.r.t. y. By re-arranging terms in (25) and applying *softmax*, $\overline{C}_x^{\boldsymbol{\lambda}}$ is cancelled out, obtaining

$$\frac{\mathbb{P}^{\text{test}, \overline{\boldsymbol{\lambda}}}(x, y)}{\sum_{j \in \mathcal{Y}} \mathbb{P}^{\text{test}, \overline{\boldsymbol{\lambda}}}(x, j)} = \frac{e^{\overline{f}_y(x) - \overline{C}_x^{\boldsymbol{\lambda}}}}{\sum_{j \in \mathcal{Y}} e^{\overline{f}_j(x) - \overline{C}_x^{\boldsymbol{\lambda}}}} = \frac{e^{\overline{f}_y(x)}}{\sum_{j \in \mathcal{Y}} e^{\overline{f}_j(x)}} = \frac{\overline{p}(x, y)}{\sum_{j \in \mathcal{Y}} \overline{p}(x, j)}. (11) \tag{26}$$

From (26) it follows that our PoE is proportional to a joint (test) distribution parameterized by $\overline{\boldsymbol{\lambda}}$, i.e. $\overline{p}(x, y) \propto \mathbb{P}^{\text{test}, \overline{\boldsymbol{\lambda}}}(x, y)$.

B Implementation details

B.1 Dataset summary

In Table 2 we include more details for the datasets used in this work.

Table 2: Summary of long-tailed datasets.

Dataset	# of classes	# of samples	Imbalance ratio
CIFAR10-LT	10	60K	{10, 50, 100}
CIFAR100-LT	100	60K	{10, 50, 100}
ImageNet-LT	1K	186K	256
iNaturalist	8K	437K	500

B.2 Training details

Following previous LT approaches (Cao et al., 2019; Liu et al., 2019), in our experiments we use cosine classifier, which is defined as $\psi(z, y) = \frac{\kappa w_y^T z}{||w_y||||z||}$, where w_y are learnable weights for class y, z denotes the output of a neural network and κ is a hyperparameter (set to 32). Our implementation is based on publicly available code by Zhang et al. (2021)[3]. For training our models, we utilize (up to) four Nvidia A100 40GB GPUs in an internal cluster.

C Additional experimental results

Here we present additional experiments and an extended analysis to further validate our approach.

C.1 Generalization performance and extended discussion

Can BalPoE improve the head-tail trade-off? In this section we provide a more detailed comparison with previous state-of-the-art approaches, by assessing test accuracy for classes in different data regimes. Tables 3, 4 and 5 present results for CIFAR100-LT-100, Inaturalist and ImageNet-LT, respectively. For CIFAR-100-LT-100, we observe that our regularized balanced product of experts significantly improves generalization under few-shot and medium-shot regimes, with only limited drops in head performance, effectively mitigating the elusive head-tail trade-off. Under the standard setting, we are close to RIDE for many-shot classes, while surpassing all baselines in few-shot, medium-shot and overall performance. Remarkably, our model is on-par with state-of-the-art approaches trained for twice as many iterations with additional data augmentation (Cubuk et al., 2019). Under similar training conditions, BalPoE achieves a new state-of-the-art, surpassing PACO and TADE by ~ 2.5 points in overall performance and obtaining further gains by leveraging seven experts (~ 5.0 points). As discussed in Section 5.3 of the paper, BalPoE can effectively tackle large-scale datasets. We achieve a new state-of-the-art for few-shot, medium-shot and overall performance for Inaturalist both under standard setting and long-training settings, see Table 4. Finally, for ImageNet-LT we obtain very promising results on-par with the current state-of-the-art, TADE, as observed in Table 5.

Do we need deep expert heads? Inspired by deep ensembles (*uniform mixture of deep experts*) (Lakshminarayanan et al., 2017), in this section we compare the performance between the typical expert-based architecture (Wang et al., 2020; Cai et al., 2021) against a *balanced product of independent experts*. The former is composed of a *shared backbone* and several *deep expert heads*, whereas for the latter, each expert is trained independently with different randomized conditions (e.g. model initialization, randomized batching order, etc.), denoted as *independent backbones*. In addition, we study the extreme case of a *product of linear experts*, where the feature representation is fully shared, and each expert head is reduced to a linear mapping (with a constraint on the weights). As seen in Figure 8, scaling up the number of *deep experts* brings consistent improvements in test accuracy over a single-expert model, unlike the case of *linear experts* whose performance remains almost constant. These results suggest that having different data representations might help in decorrelating expert

[3]Code adapted from (MIT license): `https://github.com/Vanint/TADE-AgnosticLT`

predictions, leading to more diverse mistakes (Dietterich, 2000). Our results are in line with findings for deep ensembles, where random initialization is found to be enough to promote expert diversity, by exploring different modes in function space (Fort et al., 2019). Remarkably, classes across all data regimes benefit from adding more experts, demonstrating the effectiveness of our balanced product of experts. Finally, it is worth mentioning that in cases where computational cost is of concern, one can choose to stick to the more computationally efficient shared-backbone solution.

Table 3: Test accuracy (%) of ResNet-32 trained on CIFAR-100-LT-100 for methods under comparison.

	CIFAR-100-LT-100			
Methods	Many	Medium	Few	All
Short training				
CE	67.6 ± 1.0	36.7 ± 1.2	7.6 ± 0.6	38.8 ± 0.6
LADE (Hong et al., 2021)	58.7	45.8	29.8	45.6
BALMS (Ren et al., 2020)	59.5	45.4	30.7	46.1
MiSLAS (Zhong et al., 2021)	60.4	49.6	26.6	47.0
RIDE (Wang et al., 2020)	68.1	49.2	23.9	48.0
TADE (Zhang et al., 2021)	65.4	49.3	29.3	49.8
BalPoE (ours)	$\mathbf{68.8 \pm 0.5}$	50.5 ± 0.5	24.8 ± 0.6	49.2 ± 0.5
BalPoE + Mixup (ours)	67.7 ± 0.3	$\mathbf{54.2 \pm 0.9}$	$\mathbf{31.0 \pm 0.6}$	$\mathbf{52.0 \pm 0.5}$
Long training with standard data augm.				
ACE (Cai et al., 2021)	66.1	55.7	23.5	49.4
Long training with strong data augm.				
PaCo (Cui et al., 2021)	-	-	-	52.0
TADE (Zhang et al., 2021)	-	-	-	52.2
BalPoE 3 experts + Mixup (ours)	71.6 ± 0.3	58 ± 0.7	34.6 ± 0.4	55.7 ± 0.3
BalPoE 7 experts + Mixup (ours)	$\mathbf{72.5 \pm 0.6}$	$\mathbf{59.3 \pm 0.6}$	$\mathbf{36.4 \pm 0.8}$	$\mathbf{57.0 \pm 0.2}$

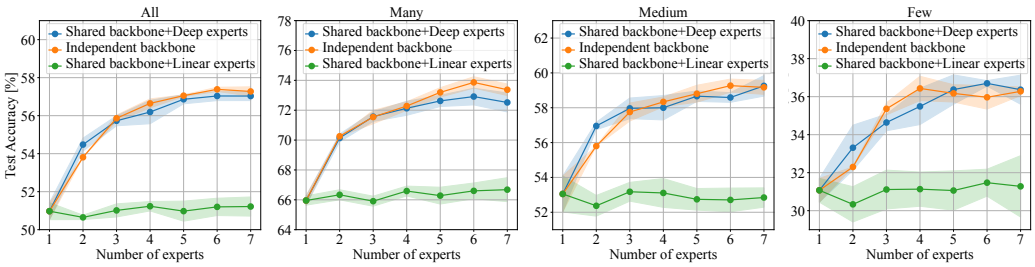
Table 4: Test accuracy (%) of ResNet-50 trained on Inaturalist-2018 for methods under comparison.

	Inaturalist			
Methods	Many	Medium	Few	All
Short training				
CE	**76.4**	66.8	60.1	65.2
BALMS (Ren et al., 2020)	70.9	70.7	70.4	70.6
MiSLAS (Zhong et al., 2021)	73.2	72.4	70.4	71.6
RIDE (Wang et al., 2020)	70.2	72.2	72.7	72.2
TADE (Zhang et al., 2021)	74.5	72.5	73.0	72.9
BalPoE (ours)	74.7	75.3	74.1	74.8
BalPoE + Mixup (ours)	73.2	**75.5**	**74.7**	**75.0**
Long training				
PaCo (Cui et al., 2021)	70.3	73.2	73.6	73.2
TADE (Zhang et al., 2021)	**75.5**	73.7	75.1	74.5
BalPoE + Mixup (ours)	75.0	**77.4**	**76.9**	**76.9**

Results on CIFAR-10-LT. We present additional results for CIFAR-10-LT in Table 6. For a brief description of this benchmark, see Table 2. We observe that our regularized balanced product of experts promotes a consistent boost in performance also under less extreme imbalance scenarios, where there is a limited number of classes with arguably enough data. Moreover, we demonstrate that, despite the lower difficulty of this task, BalPoE can still benefit from strong data augmentation and longer training, pushing the state-of-the-art on CIFAR-10-LT across all levels of class imbalance under evaluation.

Table 5: Test accuracy (%) of ResNet-50 / ResNeXt-50 trained on ImageNet-LT for methods under comparison.

	ImageNet-LT							
	ResNet50				ResNeXt50			
Methods	Many	Med.	Few	All	Many	Med.	Few	All
Short training								
CE	66.5	40.5	15.9	47.2	68.1	41.5	14.0	48.0
BALMS (Ren et al., 2020)	-	-	-	-	64.1	48.2	33.4	52.3
LADE (Hong et al., 2021)	-	-	-	-	65.1	48.9	33.4	53.0
MiSLAS (Zhong et al., 2021)	61.7	51.3	35.8	52.7	-	-	-	-
RIDE (Wang et al., 2020)	66.2	51.7	34.9	54.9	67.6	53.5	35.9	56.4
TADE (Zhang et al., 2021)	-	-	-	-	66.5	**57.0**	**43.5**	**58.8**
BalPoE (ours)	**67.0**	53.6	39.4	56.8	68.0	53.0	37.9	56.7
BalPoE + Mixup (ours)	66.7	**55.1**	**42.8**	**57.9**	**68.7**	55.5	42.2	**58.8**
Long training								
PaCo (Cui et al., 2021)	65.0	55.7	38.2	57.0	67.5	56.9	36.7	58.2
TADE (Zhang et al., 2021)	-	-	-	-	67.3	**60.4**	**46.4**	**61.2**
BalPoE + Mixup (ours)	**69.2**	**58.5**	**45.0**	**60.8**	**71.8**	57.6	42.6	61.0



(a) (b) (c) (d)

Figure 8: Test accuracy vs number of experts on CIFAR-100-LT-100 for (a) all, (b) many-shot, (c) medium-shot and (d) few-shot classes, under the long-training setting.

C.2 Calibration performance and extended discussion

Definition of calibration Intuitively, calibration is the measure of how well the model confidence reflects the true probability, i.e. when the model predicts a class with 90% confidence, it should be the correct class in 90% of the cases on average. Formally, a model is *perfectly calibrated* (Guo et al., 2017) when

$$\mathbb{P}(y = \hat{y}|p = \hat{p}) = p \qquad \forall p \in [0, 1] \tag{27}$$

where $\hat{y}$ is the predicted class and $\hat{p}$ is its associated confidence. In practice, we empirically estimate the difference between the terms on LHS and RHS of (27) over a discrete set of samples. Given a dataset $\mathcal{D} = \{x_i, y_i\}_{i=1}^N$, denote $\hat{p}_i$ the predicted confidence of sample x_i. Guo et al. (2017) propose to group predictions into M discrete intervals, and then calculate accuracy and confidence over the respective batch of samples. Let B_m denote the batch of indices in the m interval, we define the average accuracy of B_m as $acc(B_m) = \frac{1}{|B_m|} \sum_{i \in B_m} \mathbf{1}(\hat{y}_i = y_i)$. Similarly, the average confidence of B_m is defined as $conf(B_m) = \frac{1}{|B_m|} \sum_{i \in B_m} \hat{p}_i$.

Then we estimate the Expected Calibration Error (ECE) as a weighted average of the batch's differences between accuracy and confidence, i.e.

$$ECE = \sum_{m=1}^{M} \frac{|B_m|}{n} |acc(B_m) - conf(B_m)|, \tag{28}$$

Table 6: Test accuracy (%) of ResNet32 on CIFAR-10-LT for different imbalance ratios (IR). $\star$: Our reproduced results. †: From Xu et al. (2021). ‡: From Zhang et al. (2021). §: From Zhou et al. (2020).

Method ↓ IR →	CIFAR-10-LT 10	CIFAR-10-LT 50	CIFAR-10-LT 100
Short training			
CE*	87.2 ± 0.3	77.3 ± 0.4	71.3 ± 0.9
LDAM-DRW (Cao et al., 2019)	88.2	†81.8	77.1
CB-Focal (Cui et al., 2019)	84.4	79.3	74.6
LA (Menon et al., 2021)	†89.3	†83.4	79.9
Mixup§ (Zhang et al., 2018)	87.1	77.8	73.1
Remix-DRW (Chou et al., 2020)	89.0	-	79.8
MiSLAS (Zhong et al., 2021)	90.0	85.7	82.1
UniMix+Bayias (Xu et al., 2021)	89.7	84.3	82.8
RIDE‡ (Wang et al., 2020)	89.7	-	81.6
TADE (Zhang et al., 2021)	**90.8**	-	$\underline{83.8}$
BalPoE (ours)	90.3 ± 0.3	$\underline{84.6 \pm 0.3}$	80.5 ± 0.3
BalPoE + Mixup $\alpha = 0.8$ (ours)	$\underline{90.2 \pm 0.2}$	$\mathbf{86.2 \pm 0.2}$	$\mathbf{84.2 \pm 0.3}$
Long training with standard data augm.			
ACE (Cai et al., 2021)	-	84.3	81.2
Long training with strong data augm.			
BalPoE + Mixup with $\alpha = 0.8$ (ours)	$\mathbf{91.9 \pm 0.1}$	$\mathbf{88.5 \pm 0.2}$	$\mathbf{86.8 \pm 0.2}$

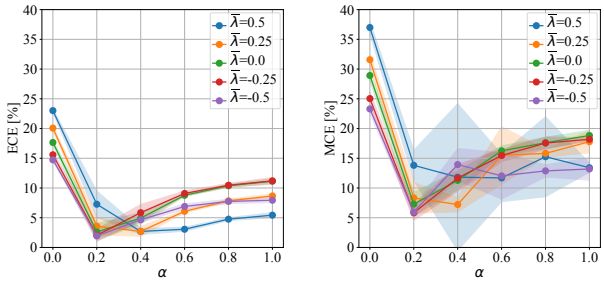
where n denotes the number of samples in each (equally spaced) interval. Analogously, the Maximum Calibration Error (MCE) describes the maximum difference between accuracy and confidence, i.e.

$$MCE = \max_{m \in \{1,..,M\}} |acc(B_m) - conf(B_m)|. \tag{29}$$

Table 7: Expected calibration error (ECE) and maximum calibration error (MCE) on CIFAR-100-LT-100 test split. $\star$: Our reproduced results. †: From Xu et al. (2021).

Methods	CIFAR-100-LT-100 ECE	CIFAR-100-LT-100 MCE
Without Mixup		
ERM / CE*	32.0 ± 0.4	47.3 ± 1.8
Bayias (Xu et al., 2021)	24.3	39.7
LA* (Menon et al., 2021)	$\underline{23.0 \pm 0.5}$	$\underline{35.0 \pm 1.3}$
BalPoE (ours)	$\mathbf{17.6 \pm 0.4}$	$\mathbf{28.9 \pm 0.9}$
Based on Mixup		
Remix† $\alpha = 1.0$ (Chou et al., 2020)	33.6	51.0
UniMix $\alpha = 0.5$ (Xu et al., 2021)	27.1	41.5
Bayias + UniMix $\alpha = 0.5$ (Xu et al., 2021)	23.0	37.4
Mixup* $\alpha = 0.4$ (Zhang et al., 2018)	9.6 ± 0.8	15.9 ± 1.5
MiSLAS + Mixup $\alpha = 0.2$ (Zhong et al., 2021)	**4.8**	-
BalPoE + Mixup $\alpha = 0.4$ (ours)	$\mathbf{4.9 \pm 1.0}$	$\mathbf{11.3 \pm 1.6}$

Comparison with state-of-the-art calibration methods. In this section we provide a comparison with previous methods for model calibration under the long-tailed setting, by assessing expected and maximum calibration errors. See Table 7 for ECE and MCE computed over CIFAR-100-LT-100 test split. As suggested by Xu et al. (2021), we observe that *fisher-consistent* approaches, such as the LA



(a) (b)

Figure 9: From left to right, (a) expected calibration error (ECE) and (b) maximum calibration error (MCE) on CIFAR-100-LT-100 test split.

loss (Menon et al., 2021) and the Bayias loss (Xu et al., 2021) improve calibration over empirical risk minimization. Our approach generalizes (Menon et al., 2021; Xu et al., 2021) to a balanced product of expert models, leading to lower calibration errors, as demonstrated in Table 7. Complementary to these methods, Mixup can reduce overconfidence and promote better predictive uncertainty by minimizing the expected risk on a *vicinal distribution* (Zhang et al., 2018). Unfortunately, Mixup's extensions for the long-tailed setting, such as Remix (Chou et al., 2020) and UniMix (Xu et al., 2021), seem to improve performance on tail classes (as shown in Table 1), by sacrificing model calibration compared to Mixup. Instead, we leverage Mixup to alleviate overconfidence, leading to stronger generalization and state-of-the-art calibration. Note that MiSLAS (Zhong et al., 2021) obtains similar calibration performance, by means of *decoupled learning* (Kang et al., 2019) and *label-aware smoothing*. Unlike MiSLAS, our approach is trained end-to-end in a single stage without incurring in data re-sampling. More importantly, BalPoE achieves state-of-the-art generalization without sacrificing model calibration, effectively keeping the best of both worlds.

Finally, for the sake of completeness, see Figure 9 for ECE and MCE as a function of Mixup's hyper-parameter α for our balanced product of experts.