ON THE CALIBRATION SET DIFFICULTY AND OUT-OF-DISTRIBUTION CALIBRATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Model calibration usually requires optimizing some parameters (*e.g.*, temperature) *w.r.t* an objective function (*e.g.*, negative log-likelihood). In this paper, we report a plain fact that the objective function is influenced by calibration set difficulty, *i.e.*, the ratio of the number of incorrectly classified samples to that of correctly classified samples [1]. If a test set has a drastically different difficulty level from the calibration set, a phenomenon out-of-distribution (OOD) data often exhibit: the optimal calibration parameters of the two datasets would be different, rendering an *optimal* calibrator on the calibration set *suboptimal* on the OOD test set and thus degraded calibration performance. With this knowledge, we propose a simple and effective method named adaptive calibrator ensemble (ACE) to calibrate OOD datasets whose difficulty is usually higher than the calibration set. Specifically, two calibration functions are trained, one for in-distribution data (low difficulty), and the other for severely OOD data (high difficulty). To achieve desirable calibration on a new OOD dataset, ACE uses an adaptive weighting method that strikes a balance between the two extreme functions. When plugged in, ACE generally improves the performance of a few state-of-the-art calibration schemes on a series of OOD benchmarks. Importantly, such improvement does *not* come at the cost of the in-distribution calibration accuracy.

1 INTRODUCTION

Model calibration aims to connect the neural network output with uncertainty. A common practice is to find optimal parameters against certain objective functions on a held-out calibration set, to obtain an optimized *calibrator*. In this paper, we focus on post-hoc calibration methods, which require training a calibration mapping function to rescale the confidence scores of a trained neural network to make it calibrated (Guo et al., 2017; Gupta et al., 2020; Kull et al., 2019). A popular technique is Temperature Scaling (Guo et al., 2017), which optimizes model temperature by minimizing the negative log-likelihood (NLL) loss.

Post-hoc calibration methods generally work well when calibrating in-distribution test sets. However, oftentimes their calibration performance drops significantly when being tested on an out-of-distribution (OOD) test set (Ovadia et al., 2019). For example, temperature scaling has shown to be ineffective under distribution shift in some scenarios (Ovadia et al., 2019). This problem happens because the test environment (OOD) is different from the training environment (in-distribution) due to factors like sample bias and non-stationarity. This paper thus aims to improve post-hoc calibration methods by producing reliable and predictive uncertainty under distribution shifts.

In the community, there exist a few works studying the OOD calibration problem (Salvador et al., 2021; Tomani et al., 2021; Wang et al., 2020). They typically aim to make amendments to the calibration set to let it approximate the OOD data in certain aspects (Salvador et al., 2021; Tomani et al., 2021). Nevertheless, these techniques are typically not adaptive to the test dataset, that is, the calibration set transformation process cannot automatically adjust to the test set. In our experiment, we observe that they improve calibration on some OOD datasets but significantly lead to decreased in-distribution calibration accuracy. In this regard, while TransCal (Wang et al., 2020) can perform domain adaptation according to the test domain, it needs to be re-trained for every new test set.

[1]To possibly facilitate reader understanding, we point out that the difficulty of a dataset (with respect to a classifier) shares the same meaning of classifier accuracy on this dataset.

In this paper, our contributions are mainly in two aspects. **First, we provide a new perspective to understand calibration failure on out-of-distribution datasets.** Specifically, we show that the calibration objective is dependent on the dataset difficulty, *i.e.,* the ratio of the number of incorrectly classified samples to that of correctly classified samples. When the calibration set and test set have the same distribution, they have a similar difficulty, and thus the calibrator learned on the calibration set would be effective on the test set (Guo et al., 2017; Gupta et al., 2020; Kull et al., 2019). However, out-of-distribution test sets usually exhibit a different (in fact, higher) difficulty level compared with the calibration set because of the distribution gap. Under this circumstance, the optimal calibration functions are different between the calibration set and OOD test sets. In other words, a calibrator that optimized on the calibration set would not be optimal on OOD data.

Second, to achieve robust calibration under distribution shifts, we propose a simple but effective method named adaptive calibrator ensemble (ACE). It adaptively integrates two predefined calibrators: 1) one trained on an easy in-distribution dataset, and 2) the other trained on a severely OOD data set with high difficulty. By estimating how much a new test set deviates from the high-difficulty calibration set, we compute a test adaptive weight to balance the force between the two calibrators. We show that our proposed ACE method improves three post-hoc calibration algorithms such as Spline (Gupta et al., 2020) on commonly used OOD benchmarks. Moreover, our method does *not* have compromised in-distribution calibration performance.

2 Related Work

Post-hoc calibration calibrates a trained neural network by rescaling confidence scores (Allikivi & Kull, 2019; Guo et al., 2017; Gupta et al., 2020; Kull et al., 2017; 2019; Liu et al., 2020; Naeini & Cooper, 2016; Nixon et al., 2019; Rahimi et al., 2020; Tian et al., 2021; Van Amersfoort et al., 2020; Wenger et al., 2020; Zadrozny & Elkan, 2001; 2002; Joy et al., 2022). For example, as a multi-class extension of Platt scaling, vector scaling and matrix scaling (Guo et al., 2017) introduce a linear layer to transform the logits vector to calibrate the network outputs. Gupta et al. (2020) obtain a recalibration function via spline-fitting, which directly maps the classifier outputs to the calibrated probabilities. Kull et al. (2019) propose a multi-class calibration method, derived from Dirichlet distributions. Rahimi et al. (2020) propose a general post-hoc calibration function that can preserve the top-k predictions of any deep network via intra order-preserving function. Our work seeks to improve the OOD performance of existing post-hoc calibrators such as vector scaling, temperature scaling, and spline, through an ensemble mechanism.

Out-of-distribution calibration. A few works study calibration under distribution shift (Salvador et al., 2021; Wang et al., 2020). To improve the post-hoc calibration under distribution shift, Salvador et al. (2021) and Tomani et al. (2021) propose to modify the calibration set to represent a generic distribution shift. Moreover, prediction uncertainty is studied in (Krishnan & Tickoo, 2020). Based on the uncertainty, an "accuracy versus uncertainty" calibration loss is proposed to encourage a model to be certain on correctly classified samples and uncertain on inaccurate samples. In comparison, our method is based on *whether samples are correctly or incorrectly classified (i.e., difficulty) rather than uncertainty*. We find difficulty is an important factor for OOD calibration failure. Furthermore, TransCal (Wang et al., 2020) uses unsupervised domain adaptation to improve temperature scaling. This method has a high computational cost because, 1) it needs an additional domain adaptation training process, and 2) every time it meets a new test set, the domain adaptation model needs to be re-trained. We contribute from a different perspective to the existing literature. We provide insight into the role of dataset difficulty on the failure of existing algorithms on OOD data. We then propose a simple and effective ensemble strategy to improve post-hoc calibrators in a test set adaptive manner.

3 Methodology

3.1 Preliminaries

Neural network notations. Considering the task of calibrating neural networks for n-way classification, let us define $[n] \coloneqq \{1, \ldots, n\}$, $\mathcal{X} \subseteq \mathbb{R}^d$ be the domain, $\mathcal{Y} = [n]$ be the label space, and Δ_n denote the $n-1$ dimensional unit simplex. Given a training dataset $\mathcal{D}_{tr}$ of independent and identically distributed (i.i.d.) samples drawn from an unknown distribution π on $\mathcal{X} \times \mathcal{Y}$, we learn a probabilistic predictor $\phi : \mathbb{R}^d \rightarrow \Delta_n$. We assume that ϕ can be expressed as the composition

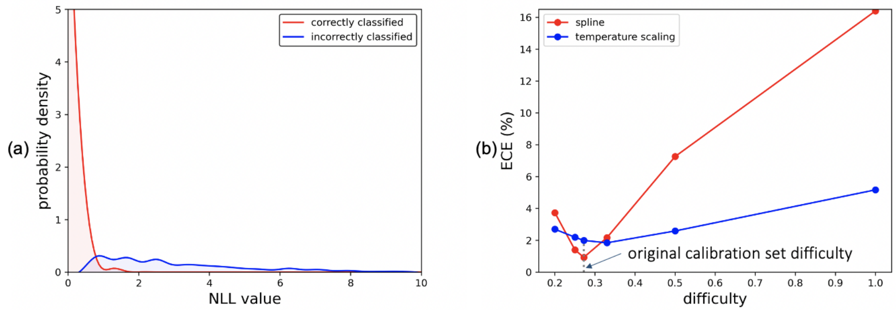

Figure 1: **(a) NLL values of correctly and incorrectly classified samples.** We use ResNet-152 on the in-distribution ImageNet calibration set (described in Section 4.1) and plot NLL probability density of the two types of samples. We clearly observe that correctly classified samples generally have a much lower NLL value. **(b) The impact of calibration set difficulty ($\frac{N_F}{N_T}$) on calibration performance (ECE).** We manually select images from the in-distribution calibration set to create new calibration sets of various difficulty levels. The difficulty of the original calibration set is 0.2723. We use the ResNet-152 model and an in-distribution test set from ImageNet. We evaluate two methods (temperature scaling, and Spline), and at the same time, mark the difficulty of the original in-distribution calibration set (gray vertical dotted line). We find the calibration sets having similar difficulty to the original will lead to good calibration performance and vice versa.

$\phi =: \mathbf{sm} \circ \mathbf{g}$, with $\mathbf{g} : \mathbb{R}^d \rightarrow \mathbb{R}^n$ being a non-probabilistic n-way classifier and $\mathbf{sm} : \mathbb{R}^n \rightarrow \Delta_n$ being the softmax operator $\mathbf{sm}_i(\mathbf{z}) = \frac{\exp(\mathbf{z}_i)}{\sum_{j=1}^n \exp(\mathbf{z}_j)}$, for $i \in \mathcal{Y}$, where the subscript $_i$ denotes the i-th element of a vector. We say $\mathbf{g}(\mathbf{x})$ is the *logits* of $\mathbf{x}$ with respect to ϕ.

Definition of a calibrated network. When queried at $(\mathbf{x}, y) \in \mathcal{X} \times \mathcal{Y}$ sampled from an unknown distribution π, the probabilistic predictor ϕ returns $\hat{y} =: \arg\max_i \phi_i(\mathbf{x})$ as the predicted label and $\hat{p} =: \max_i \phi_i(\mathbf{x})$ as the associated confidence score. We say ϕ is *perfectly calibrated* with respect to π, if $\hat{p}$ is expected to represent the true probability of correctness. Formally, a perfectly calibrated model satisfies $\mathbb{P}(\hat{y} = y | \hat{p} = p) = p$ for any $p \in [0, 1]$. In practice, we commonly use the Expected Calibration Error (ECE) as the calibration performance metric. It first groups all samples into M equally interval bins $\{B_m\}_{m=1}^M$ with respect to their confidence scores, and then calculates the expected difference between the accuracy and average confidence: $\text{ECE} = \sum_{m=1}^M \frac{|B_m|}{n} |\text{acc}(B_m) - \text{avgConf}(B_m)|$, where n denotes the number of samples.

Post-hoc calibration learns a post-hoc calibration function $\mathbf{f} : \mathcal{R}^n \rightarrow \mathcal{R}^n$ such that the new probabilistic predictor $\phi_c := \mathbf{sm} \circ \mathbf{f} \circ \mathbf{g}$ is better calibrated *and* tries to keep a similar (or same) accuracy of the original network ϕ.

3.2 Post-hoc Calibration Function Is Influenced by Calibration Set Difficulty

Post-hoc calibration loss function. Assume we have a held-out calibration dataset $\mathcal{D}_c = \{(\mathbf{x}^i, y^i)\}_{i=1}^N$ with i.i.d samples from the unknown distribution π on $\mathcal{X} \times \mathcal{Y}$ and a calibration function $\mathbf{f}$ parameterized by some vector θ. The empirical calibration loss is generally defined as,

$$\frac{1}{N} \sum_{i=1}^N \ell(y^i, \mathbf{f}(\mathbf{z}^i)) + \frac{\lambda}{2} ||\theta||^2, \tag{1}$$

where $\mathbf{z}^i = \mathbf{g}(\mathbf{x}^i)$, $\ell : \mathcal{Y} \times \mathcal{R}^n \rightarrow \mathcal{R}$ is a cost function, and $\lambda \geq 0$ is the regularization weight. $\ell(\cdot, \cdot)$ is the network classification loss. Following existing literature, we employ the commonly used negative log-likelihood (NLL) loss:

$$\ell(y, \mathbf{f}(\mathbf{z})) = -\log(\mathbf{sm}_y(\mathbf{f}(\mathbf{g}(\mathbf{x})))), \tag{2}$$

where $\mathbf{sm}$ is softmax operator, and $\mathbf{sm}_y$ is its y-th element.

Plain fact: individual samples matter in the classification loss. Apparently, a major component in the calibration objective (Eq. 1) is the model classification loss (*e.g.,* the commonly used NLL loss, Eq. 2). If a sample is correctly classified, the classification loss will likely return a small value; If a sample is incorrectly classified, there will likely be a high loss value. Therefore, *whether an individual sample is correctly classified or not would lead to different classification loss values*.

We conduct an empirical analysis to verify this conclusion. Specifically, we use a ResNet-152 classifier trained on the ImageNet dataset (Deng et al., 2009). The NLL values of these samples are computed on the calibration set (described in Section 4.1), and summarily drawn in Fig. 1(a). It is clearly shown that the NLL values of correctly classified samples are close to 0 while those of incorrectly classified samples are significantly greater.

Collectively, calibration set difficulty influences calibration optimization. To illustrate this point, we use NLL as an example, which is a commonly used classification loss. Given that the two types of samples have different NLL values, we decompose the NLL loss into two parts:

$$\ell_T(y, \mathbf{f}(\mathbf{z})) = -\frac{1}{N_T}\sum_{i}^{N_T}\log(\mathbf{sm}_{y^i}(\mathbf{f}(\mathbf{g}(\mathbf{x}^i)))), \tag{3}$$

where $\arg\max \mathbf{sm}(\mathbf{g}(\mathbf{x}^i)) = y^i$, and,

$$\ell_F(y, \mathbf{f}(\mathbf{z})) = -\frac{1}{N_F}\sum_{i}^{N_F}\log(\mathbf{sm}_{y^i}(\mathbf{f}(\mathbf{g}(\mathbf{x}^i)))), \tag{4}$$

where $\arg\max \mathbf{sm}(\mathbf{g}(\mathbf{x}^i)) \neq y^i$. In Eq. 3 and Eq. 4, N_T and N_F note the numbers of correctly and incorrectly classified samples, respectively. By adjusting $\frac{N_F}{N_T}$, the overall NLL value changes, which will affect the optimized calibration parameters $\boldsymbol{\theta}$ (*a.k.a.* the calibration function).

Formally, we define the *difficulty* of a dataset as $\frac{N_F}{N_T}$. Note that, the difficulty of a dataset (with respect to a classifier) shares the same meaning as classifier accuracy on this dataset. The above analysis indicates that *optimized calibration parameters are affected by the difficulty of the calibration set*: **1)** θ trained on a more difficult calibration set tends to have a larger classification loss values (Eq. 2) and thus a larger calibration loss (Eq. 1). **2)** θ trained on an easier calibration set likely corresponds to a smaller classification loss (Eq. 2) and thus a lower calibration loss (Eq. 1).

We empirically verify the above conclusion in Fig. 1(b), where we create calibration sets with various levels of difficulty ($\frac{N_F}{N_T}$) and mark the difficulty level of the original calibration set. It indicates that calibration set difficulty indeed influences ECE of two calibration methods: temperature scaling (NLL) (Guo et al., 2017), Spline (KS-error) (Gupta et al., 2020). Moreover, when testing the original in-distribution data, if the difficulty of the created calibration dataset is similar, the two calibration methods generally have good calibration performance. However, calibration accuracy is poorer when the *difficulty* of created calibration dataset is very different from that of the original calibration dataset.

The above analysis mostly uses the NLL loss as an example, but can also apply to some other classification loss functions (*e.g.*, the KS-error used in Spline is verified in Fig. 1(b), the cross-entropy loss and the focal loss). These loss functions are usually influenced by individual samples, and thus collectively the dataset difficulty would eventually impact the calibration performance.

3.3 Calibration Set Difficulty Influences Out-of-distribution Calibration

Having analyzed that calibration set difficulty influences the calibration performance on in-distribution test sets, we provide a tentative explanation of *why calibrators trained on in-distribution data fail on OOD test sets*. Essentially, an OOD test set usually has a different difficulty level from the in-distribution calibration set. In fact, the OOD difficulty level is usually higher, *i.e.,* there is a higher percentage of incorrectly classified samples, because of the domain gap problem (Ben-David et al., 2010; Deng & Zheng, 2021; Mansour et al., 2009). Therefore, the calibration mapping function $\mathbf{f}$ that an OOD test set needs is different from the in-distribution calibration set. When training a calibrator on in-distribution data, its performance would thus be suboptimal on the OOD test data.

Moreover, our reasoning also helps understand why some existing OOD calibration methods have compromised calibration performance on in-distribution test sets. Specifically, these methods, *e.g.,* Perturbation by Tomani et al. (2021), obtain their mapping functions on some modified in-distribution calibration set (*e.g.,* adding Gaussian noise), which to some extent mimics the OOD test set. However, this modification operation is not adaptive, that is, they do not change *w.r.t* the test set. When the test set changes to an in-distribution one, its optimal calibration parameters would be different from those obtained from the modified calibration set. This is possible because of different difficulty levels.

Based on the above consideration, we propose ACE (adaptive calibrator ensemble) method to achieve robust calibration. In the next section, we will introduce our method that 1) ensembles the outputs of two extreme calibrators, 2) is used to improve post-hoc calibration algorithms on OOD data, and 3) reduces to in-distribution calibration effect when tested on in-distribution data.

3.4 Robust Calibration with Adaptive Calibrator Ensemble

Overview. To achieve desirable calibration under distribution shifts, we propose a simple and effective method called Adaptive Calibrator Ensemble (ACE). Using an in-distribution calibration set as input, ACE outputs an OOD calibrator as if having been trained on a calibration set with a proper difficulty level. To do so, we first seek two calibration sets with extreme difficulty levels: an in-distribution difficulty level (easy) and a high difficulty level (hard). We then use an adaptive weighting scheme to fuse the output of calibrators trained on the two extreme calibration sets.

Finding two datasets with extreme difficulty levels. Straightforwardly, we secure the "easy" one as the in-distribution calibration set itself $\mathcal{D}_o$. To obtain the "hard" calibration set $\mathcal{D}_h$, we perform sampling on $\mathcal{D}_o$ aiming to increase the difficulty. Specifically, we apply the classifier on the in-distribution calibration set to find correctly classified samples, incorrectly classified samples, and their numbers N_T^o and N_F^o (N_T^o is usually greater than N_F^o). To create $\mathcal{D}_h$, we calculate its N_T and N_F as follows, $N_F^h = N_F^o, N_T^h = N_F^o/d$, where $d \in (0, \infty)$ is a pre-defined difficulty level (hyperparameter). We then randomly sample $\mathcal{D}_o$ to achieve this difficulty level. When d is relatively large[2], the calibration set contains many more incorrectly samples than correct ones, allowing us to have the desired calibration set $\mathcal{D}_h$, which is considered seriously out-of-distribution and hard.

Training two calibrators on the two extreme datasets. On *each* of the obtained the easy and the hard calibration sets $\mathcal{D}_o$ and $\mathcal{D}_h$, we train a calibrator. Let $\mathbf{g}$ denote the deep learning model. For calibration dataset $\mathcal{D}_o = \{(\mathbf{x}^i, y^i)\}_{i=1}^{N_o}$, where N_o means the number of samples of $\mathcal{D}_o$, we train a calibration function $\mathbf{f}_o$, and the calibrated confidence scores are denoted as $\mathbf{q}_o^i = \mathbf{f}_o(\mathbf{z}^i)$. Here, $\mathbf{z}^i = \mathbf{g}(\mathbf{x}^i), \mathbf{x}^i \in \mathcal{D}_{test}$, where $\mathcal{D}_{test}$ is a new test set that is either in-distribution or OOD. Similarly, for calibration set $\mathcal{D}_h = \{(\mathbf{x}^i, y^i)\}_{i=1}^{N_h}$, where N_h means the number of samples of $\mathcal{D}_h$, we train a calibration function $\mathbf{f}_h$, the calibrated confidence score is $\mathbf{q}_h^i = \mathbf{f}_h(\mathbf{z}^i)$.

An adaptive method to ensemble outputs of the two calibrators. Given $\mathcal{D}_o$ and $\mathcal{D}_h$, we intuitively speculate that the difficulty of a usual out-of-distribution test set would be positioned in between. As such, we propose to compute an adaptive weight α to balance the difficulty of these two models:

$$\mathbf{q} = \alpha \mathbf{q}_o + (1 - \alpha)\mathbf{q}_h. \tag{5}$$

In designing a reasonable weight α, we request it to be test-set-adaptive. **First**, when the distribution of an OOD test set is similar to the original calibration set $\mathcal{D}_o$, $\alpha \to 1$, so that the system reduces to in-distribution calibrator $\mathbf{q}_o$; **Second**, when a test set is seriously out-of-distribution, $\alpha \to 0$. Moreover, Guillory et al. (2021) suggest that the average confidence score could serve as an unsupervised indicator of the degree of how out-of-distribution a test set is. So given an unlabeled test set $\mathcal{D}_{test}$, we can estimate an approximate OOD degree of this test set. Here, we compute α as,

$$\alpha = \frac{\text{avgConf}(\mathcal{D}_{test})}{\text{avgConf}(\mathcal{D}_o)}, \tag{6}$$

where avgConf$(\cdot)$ calculates the average confidence score of a dataset. In the experiment, we will evaluate some fixed values of α, which are useful on some occasions but less so on others. Moreover, being fixed implies that it does not work for in-distribution data unless it is fixed to 1.

Compared with TransCal (Wang et al., 2020) which needs to be re-trained on every new test set, our ACE, once obtaining the two calibrators, adds negligible time cost when adapting to test sets. In fact, the confidence vectors of $\mathcal{D}_{test}$ can be stored while performing model testing, allowing Eq. 6 to be fast executed (average confidence of $\mathcal{D}_o$ can be pre-computed during training).

[2]By default, we set $d = 10$, which means 10 times more incorrectly classified samples than correct ones.

3.5 Discussion

Difficulty is a relative concept. Despite being formulated as $\frac{N_F}{N_T}$, difficulty also depends on the model or classifier. For stronger models, the difficulty level would be lower (even $N_F = 0$) and vice versa. In this paper, we assume fixed models and choose not to put the model as a subscript in the definition of difficulty for simplicity.

Domain gap *vs.* difficulty. Domain gap is used to describe the distribution difference between domains and certainly exists between an OOD test set and the calibration set. Therefore, a possible way to calibrate OOD data is to find a dataset with similar distribution to the OOD test set, which is essentially reflected in (Salvador et al., 2021; Tomani et al., 2021). Our paper points out a new way to craft the domain gap by modifying the difficulty of the calibration set. In fact, domain gap is a complex phenomenon and related to many factors aside from difficulty, so it would be interesting to investigate other factors which can help OOD calibration.

An alternative method. We emphasize the main contribution is to report that calibration set difficulty is influential on OOD calibration performance. The designed method, in comparison, is more from an intuitive perspective. There might be other alternatives. For example, we could use the average confidence of a dataset (we use it in Eq. 6 to calculate α instead) to estimate its difficulty and create a calibration set that has a closer difficulty level to the OOD test dataset. We show this alternative also gives improvement over some baselines. (Please refer to Appendix D.3 for more details.)

Potential limitation and further direction. Our weighting method (Eq. 5) assumes that an OOD test set sits between $\mathcal{D}_o$ and $\mathcal{D}_h$ in terms of difficulty. This assumption should be valid for most cases in practice because the difficulty of $\mathcal{D}_o$ is very low and that of $\mathcal{D}_h$ is very high (we use $d = 10$ by default, which translates to 9.1% top-1 accuracy). We empirically observe that $d = 10$ is effective, which translates to an accuracy of 9.09%. We believe a dataset with 9.09% accuracy is difficult enough to cover a wide range of test sets. In addition, distribution shift occurs in a variety of ways (Hendrycks et al., 2021a; Taori et al., 2020). There might exist scenarios (*e.g.,* adversarial attack) where the confidence score is less effective in describing the distribution shift. In such cases, our method might not be able to achieve significant improvement over existing algorithms. In fact, it would be interesting to explore other potential ways to characterize distribution discrepancy.

4 Experiment

4.1 Experimental Setup

Neural Networks. We consider both convolutional and non-convolutional networks. Specifically, we use ResNet-152 (He et al., 2016), ViT-Small-Patch32-224 (Dosovitskiy et al., 2020) and Deit-Small-Patch16-224 (Touvron et al., 2021). The three networks are either trained or fine-tuned on the ImageNet training set (Deng et al., 2009).

Calibration set and in-distribution test set. Following the protocol in (Gupta et al., 2020), we divide the validation set of ImageNet into two halves: one for the in-distribution test (namely ImageNet-Val), the other for learning calibration methods (namely calibration set $\mathcal{D}_o$).

Out-of-distribution test sets. In the experiment, we use the following *six real-world* out-of-distribution benchmarks. (i) ImageNet-V2 (Recht et al., 2019) is a new version of ImageNet test set. It contains three different sets resulting from different sampling strategies: Matched-Frequency (A), Threshold-0.7 (B), and Top-Images (C). Each version has $10,000$ images from 1000 classes; (ii) ImageNet-S(ketch) (Wang et al., 2019) shares the same 1000 classes as ImageNet but all the images are black and white sketches. It contains $50,000$ images; (iii) ImageNet-R(endition) (Hendrycks et al., 2021a) contains artificial renditions of ImageNet classes. It has $30,000$ images of 200 classes. Following (Hendrycks et al., 2021a), we sub-select the model logits for the 200 classes before computing calibration metrics. (iv) ImageNet-Adv(ersarial) (Hendrycks et al., 2021b) is adversarially selected to be hard for ResNet-50 trained on ImageNet. It has $7,500$ samples of 200 classes. As for ImageNet-R, we sub-select the logits for the 200 classes before computing the calibration metric. Moreover, we test on synthetic CIFAR-10-C(orruptions) and ImageNet-C(orruptions) (Hendrycks & Dietterich, 2019). Both these two datasets are modified with synthetic perturbations such as blur,

Table 1: OOD calibration performance of our method (ACE) integrated with three post-hoc methods: vector scaling, temperature scaling (Temp. Scaling), and Spline. ECE (25 bins, %) for top-1 predictions is reported. We use **ResNet-**152 on various image classification datasets with *various distribution shifts*. For each column, the lowest number is in **bold** and the second lowest <u>underlined</u>. Our method (ACE) effectively improves the post-hoc methods on 15 out of 18 occasions. ▲/▼ denotes ECE is lower / higher than the post-hoc method when being used alone, with statistical significance (p-value < 0.05) based on the two-sample t-test.

Methods	ImgNet-V2-A	ImgNet-V2-B	ImgNet-V2-C	ImgNet-S	ImgNet-R	ImgNet-Adv
uncalibrated	9.5016	6.2311	4.3117	24.6332	17.8621	50.8544
Vector Scaling	6.8068	4.2184	2.9258	20.3726	14.5037	44.7593
+ ACE	5.6291 ±0.0397 ▲	3.7742 ±0.0237 ▲	3.1141 ±0.0150 ▼	15.8747 ±0.0252 ▲	10.6343 ±0.0356 ▲	40.5773 ±0.0491 ▲
Temp. Scaling	4.4413	2.7309	1.6831	15.7879	10.4797	42.6302
+ ACE	<u>3.5615</u> ±0.0028 ▲	2.5692 ±0.0013 ▲	1.7021 ±0.0001 ▼	<u>10.3915</u> ±0.0092 ▲	<u>6.7458</u> ±0.0083 ▲	<u>38.0651</u> ±0.0114 ▲
Spline	4.5321	**1.8034**	<u>1.3357</u>	19.6392	13.1116	45.3623
+ ACE	**2.8201** ±0.0283 ▲	<u>2.0235</u> ±0.0154 ▼	**1.0550** ±0.0092 ▲	**6.9264** ±0.0864 ▲	**6.8533** ±0.0011 ▲	**31.0926** ±0.0422 ▲

pixelation, and compression artifacts at a range of severities. We use 80 different distortions (16 different types with 5 levels of intensity each) which are the same as those in (Ovadia et al., 2019).

Post-hoc calibration methods. In the experiment, we validate the effectiveness of ACE by integrating it with the existing calibration methods through which we obtain confidence scores $\mathbf{q}$ (Section 3.4). Specifically, we use vector scaling (Guo et al., 2017), temperature Scaling (Guo et al., 2017), and Spline (Gupta et al., 2020) as baseline calibrators, and compare with a recent method Perturbation (Tomani et al., 2021) which is specifically designed for OOD calibration. In addition, we also compare with more existing calibration methods, *i.e.,* Vanilla (Hendrycks & Gimpel, 2016), Ensemble (Lakshminarayanan et al., 2016), SVI (Wen et al., 2018), SVI-AvUC and SVI-AvUTS (Krishnan & Tickoo, 2020), to show our method competitive.

4.2 Calibration on Out-of-distribution Datasets

ACE improves calibration methods on out-of-distribution datasets. We evaluate our method combined with three post-hoc calibrators on six out-of-distribution test sets and compare it with those calibrators used alone. Table 1 shows ECE (using 25 bins) results of ResNet-152. Our ACE is shown to consistently improve the OOD calibration results of the three baseline calibrators in most of the test cases. For example, when calibrating ResNet-152, our method improves temperature scaling by 0.88%, 0.17%, 5.40%, 3.73% and 4.57% decrease in ECE, on ImageNet-V2-A/B, ImageNet-S/R/Adv, respectively. Under the same settings, the ECE of our method is slightly higher (0.019%) than the baseline on the ImageNet-V2-C dataset.

ACE works effectively under two other neural networks. To show the effectiveness of our method for different backbones, we adopt two transformer models (ViT-Small-Patch32-224 and Deit-Small-Patch16-224) as backbones, and experimental settings are the same as those in Table 1. Table 2 indicates that for backbone ViT-Small-Patch32-224 our method reduces ECE of the three baselines on five out of the six OOD test sets. For example, compared with Spline, ECE of our method is 1.82%, 0.28%, 11.13%, 6.16% and 14.53% lower on ImageNet-V2-A/C, ImageNet-S/R/Adv, respectively. On the other hand, Table 2 demonstrates that for the Deit-Small-Patch16-224 backbone, our method is beneficial on all the six OOD test sets. In addition, comparing the *uncalibrated* results of the three backbones, transformer models generally have a lower ECE under OOD test sets. Specifically, *ViT-Small-Patch*32-224 is shown to be superior to Deit-Small-Patch16-224 on four out of six test sets.

Comparison with the existing calibration methods. In Table 3, we compare our method with the state-of-the-art methods, *i.e.*, various variants of AvUC (Krishnan & Tickoo, 2020) and Ensemble (Lakshminarayanan et al., 2016), on CIFAR-10-C and ImageNet-C. Following the protocol in (Ovadia et al., 2019; Krishnan & Tickoo, 2020), we report the results at intensity 5. Our method improves Spline by reducing ECE by 11.18% and 6.70% on CIFAR-10-C and ImageNet-C, respectively.

Table 2: OOD calibration performance (ECE, %) of our method (ACE) and Perturbation (Tomani et al., 2021) applied on Spline (Gupta et al., 2020). We report results using three neural networks: ResNet-152 (He et al., 2016) (ResNet), ViT-Small-Patch32-224 (Dosovitskiy et al., 2020) (ViT), and Deit-Small-Patch16-224 (Touvron et al., 2021) (Deit). All the other notations and settings are the same with Table 1. Our method improves the calibrator baselines in 16 out of 18 scenarios, while Perturbation has mixed performance.

Models	Methods	ImgNet-V2-A	ImgNet-V2-B	ImgNet-V2-C	ImgNet-S	ImgNet-R	ImgNet-Adv
ResNet	Spline	4.5321	**1.8034**	1.3357	19.6392	13.1116	45.3623
	+ ACE	**2.8201** ▲	2.0235 ▼	**1.0550** ▲	**6.9264** ▲	6.8533 ▲	**31.0926** ▲
	+ Perturbation	5.4175 ▼	8.2109 ▼	9.3326 ▼	7.9805 ▲	**2.9171** ▲	32.3677 ▲
ViT	Spline	4.7572	**1.6859**	1.4683	15.9864	12.5494	38.0404
	+ ACE	**2.9329** ▲	2.0832 ▼	**1.1831** ▲	**4.8514** ▲	6.3699 ▲	23.5147 ▲
	+ Perturbation	5.0302 ▼	6.3854 ▼	7.8929 ▼	5.9254 ▲	**3.7302** ▲	**22.5118** ▲
Deit	Spline	5.0289	2.1261	1.3923	20.7714	9.6996	31.3674
	+ ACE	**2.4576** ▲	**1.6475** ▲	**1.3544** ▲	**5.6622** ▲	**3.6721** ▲	**15.7885** ▲
	+ Perturbation	3.3520 ▲	2.4547 ▼	2.9461 ▼	15.9003 ▲	8.1481 ▲	27.9474 ▲

Table 3: Method comparison on CIFAR-10-C and ImageNet-C with ResNet-20 and ResNet-50, respectively. Following the protocol in Ovadia et al. (2019), we report mean ECE (10 bins for CIFAR-10-C and 25 bins for ImageNet-C, %) across 16 different types of data shift at intensity 5 with lowest numbers in **bold** and the second lowest underlined. For each row, we compare ACE with the best of the competing ones (*i.e.,* SVI-AvUC) using the two-sample t-test.

Dataset	Uncalibrated	Ensemble	SVI	SVI-AvUTS	SVI-AvUC	Spline	Spline+ACE
CIFAR-10-C	0.1942	0.1611	0.2389	0.1585	0.1374	0.3382	**0.1264** ▲
ImageNet-C	0.3151	0.0880	0.1188	0.0800	0.0542	0.1147	**0.0477** ▲

Compared with these methods, our method is competitive on both ImageNet-C and CIFAR-10-C. For example, for CIFAR-10-C, our method achieves 3.21% and 1.10% lower calibration error than SVI-AvUTS and SVI-AvUC, respectively.

4.3 ACE Does Not Compromise In-distribution Calibration

We show ECE results on in-distribution test set (ImageNet-Val) using ResNet-152. We adopt the same three post-hoc calibration baselines and Perturbation (Tomani et al., 2021) for comparison. As shown in Fig. 2(a), we observe that the post-hoc calibration baselines themselves effectively reduce the ECE score compared with the uncalibrated system and that Spline generally performs the best. Perturbation is shown to deteriorate the calibration performance for all three baselines. Because Perturbation is not adaptive to different test sets, its effectiveness is not guaranteed when a test set is out of its optimal domain confined by the generated diverse set. In comparison, when our method is integrated with the baselines, the resulting calibration performance is very close to the baselines when being used alone. This is mainly because of the adaptive weighting scheme (see Section 3.4 for more explanations). Thus our method is *not* compromised on the in-distribution test set.

4.4 Component Analysis of ACE

Impact of the size of the two extreme calibration sets. ACE uses an "easy" calibration set $\mathcal{D}_o$ (the original calibration set) and a "hard" calibration set $\mathcal{D}_h$. The original $\mathcal{D}_o$ and $\mathcal{D}_h$ have $25,000$ and $5,885$ images, respectively. Here, we simultaneously reduce the size of $\mathcal{D}_o$ and $\mathcal{D}_h$ by a certain percentage and report calibration accuracy (ECE) in Fig. 2(b). From the results on the in-distribution dataset ImageNet-Val and out-of-distribution dataset ImageNet-V2-A, we observe that our method is relatively stable on both test sets when the size changes. Yet for best results, we recommend using possibly large calibration sets.

Impact of the difficulty of $\mathcal{D}_h$. To analyze the impact of hyperparameter d (Section 3.4), we create multiple $\mathcal{D}_h$ with various values of d. Results are shown in Fig. 3 (a). We observe that calibration

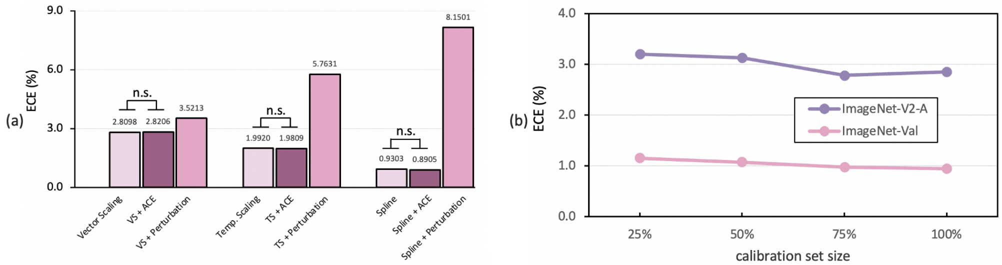

Figure 2: **(a) Evaluation of ACE on the ID test set ImageNet-Val.** We calibrate the ResNet-152 classifier and use ECE (%) for top-1 predictions as evaluation metric. "n.s." means the difference between results is not statistically significant (p-value > 0.05). **(b) Effect of the size of the two extreme calibration sets.** Starting from their original size ($25,000$ and $5,885$ images respectively for $\mathcal{D}_o$ and $\mathcal{D}_h$), we randomly select a certain percentage of calibration sets. We report ECE of ResNet-152 with Spline on ImageNet-Val and ImageNet-V2-A.

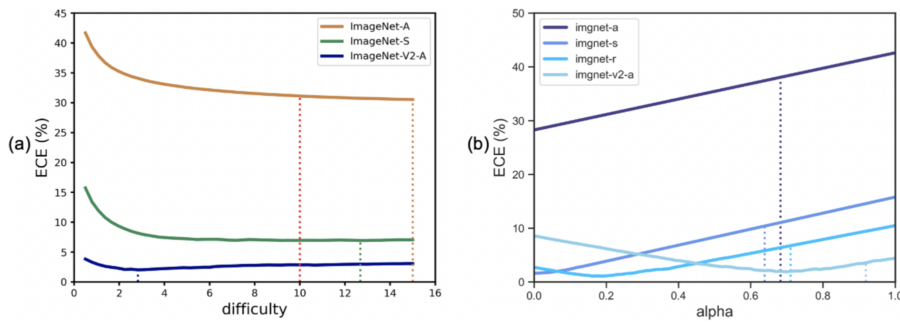

Figure 3: **(a) Impact of hyperparameter d on calibration accuracy.** We densely sample values of d (0.5 to 15) and report ECE (%) of ResNet-152 with Spline on ImageNet-V2-A, ImageNet-S and ImageNet-Adv. We also mark the results using our empirically selected value ($d = 10$) and the optimal values shown by the dotted vertical line. **(b) Densely sampled values of α (0 to 1) *vs.* our computed α (Eq. 5)** Comparing with the densely sampled values of α, computed α (shown by the dotted vertical line) is close to the optimal value with reasonable difficulty for each test set.

accuracy is slightly higher on ImageNet-Adv when the hard calibration set is more difficult, while the performance on the other two datasets drops at the same time. Moreover, we find the optimal difficulty is different for various test sets. That said, by setting $d = 10$, we generally have good performance, and it is important to note that this difficulty level is considerably high (equivalent to 9.09% classification accuracy) and thus covers most test scenarios.

Optimal adaptive α vs. our computed α (Eq. 5). We compare both values in Fig. 3(b). First, for datasets with normal difficulty (*e.g.,* ImageNet-V2-A), the value computed by our scheme is quite close to the optimal value. Second, for extremely difficult datasets such as ImageNet-S and ImageNet-A, α computed by our method is less optimal. That said, we emphasize that in practice it is infeasible to do a greedy search because the target is unlabeled, where our method is generally useful.

5 Conclusion

This paper studies how to calibrate a model on OOD datasets. Our important contribution is diagnosing why existing post-hoc algorithms fail on OOD test sets. Specifically, we report the difficulty of the calibration set influences the calibration function learning, and in other words, an OOD test set would witness poor calibration performance if the calibration set does not have an appropriate difficulty level. Realizing the importance of calibration set difficulty, we design a simple and effective method named adaptive calibrator ensemble (ACE) which combines the outputs of two calibrators trained on datasets with extreme difficulties. We show that ACE improves three commonly used calibration methods on various OOD test sets (*e.g.*, ImageNet-C) without degrading ID calibration performance. In future work, we would like to further study how the domain gap and calibration set difficulty interact with each other and thereby improve OOD calibration.

REFERENCES

Mari-Liis Allikivi and Meelis Kull. Non-parametric bayesian isotonic calibration: Fighting over-confidence in binary classification. In *ECML/PKDD (2)*, pp. 103–120, 2019.

Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine learning*, 79(1):151–175, 2010.

Glenn W Brier. Verification of forecasts expressed in terms of probability. *Monthly weather review*, 78(1):1–3, 1950.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Weijian Deng and Liang Zheng. Are labels necessary for classifier accuracy evaluation? In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2021.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

S. Garg, S. Balakrishnan, Z. C. Lipton, B. Neyshabur, and H. Sedghi. Leveraging unlabeled data to predict out-of-distribution performance. 2022.

Devin Guillory, Vaishaal Shankar, Sayna Ebrahimi, Trevor Darrell, and Ludwig Schmidt. Predicting with confidence on unseen distributions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1134–1144, 2021.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International Conference on Machine Learning*, pp. 1321–1330. PMLR, 2017.

Kartik Gupta, Amir Rahimi, Thalaiyasingam Ajanthan, Thomas Mensink, Cristian Sminchisescu, and Richard Hartley. Calibration of neural networks using splines. *arXiv preprint arXiv:2006.12800*, 2020.

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*, pp. 770–778, 2016.

Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *Proceedings of the International Conference on Learning Representations*, 2019.

Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *arXiv preprint arXiv:1610.02136*, 2016.

Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8340–8349, 2021a.

Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. *CVPR*, 2021b.

Tom Joy, Francesco Pinto, Ser-Nam Lim, Philip HS Torr, and Puneet K Dokania. Sample-dependent adaptive temperature scaling for improved calibration. *arXiv preprint arXiv:2207.06211*, 2022.

Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, et al. Wilds: A benchmark of in-the-wild distribution shifts. In *International Conference on Machine Learning*, pp. 5637–5664. PMLR, 2021.

Ranganath Krishnan and Omesh Tickoo. Improving model calibration with accuracy versus uncertainty optimization. *Advances in Neural Information Processing Systems*, 33:18237–18248, 2020.

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

Meelis Kull, Telmo Silva Filho, and Peter Flach. Beta calibration: a well-founded and easily implemented improvement on logistic calibration for binary classifiers. In *Artificial Intelligence and Statistics*, pp. 623–631. PMLR, 2017.

Meelis Kull, Miquel Perello-Nieto, Markus Kängsepp, Hao Song, Peter Flach, et al. Beyond temperature scaling: Obtaining well-calibrated multiclass probabilities with dirichlet calibration. *arXiv preprint arXiv:1910.12656*, 2019.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *arXiv preprint arXiv:1612.01474*, 2016.

Jeremiah Liu, Zi Lin, Shreyas Padhy, Dustin Tran, Tania Bedrax Weiss, and Balaji Lakshminarayanan. Simple and principled uncertainty estimation with deterministic deep learning via distance awareness. *Advances in Neural Information Processing Systems*, 33:7498–7512, 2020.

Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh. Domain adaptation: Learning bounds and algorithms. In *Proc. COLT*, 2009.

Mahdi Pakdaman Naeini and Gregory F Cooper. Binary classifier calibration using an ensemble of near isotonic regression models. In *2016 IEEE 16th International Conference on Data Mining (ICDM)*, pp. 360–369. IEEE, 2016.

Jeremy Nixon, Michael W Dusenberry, Linchuan Zhang, Ghassen Jerfel, and Dustin Tran. Measuring calibration in deep learning. In *CVPR Workshops*, volume 2, 2019.

Yaniv Ovadia, Emily Fertig, Jie Ren, Zachary Nado, David Sculley, Sebastian Nowozin, Joshua V Dillon, Balaji Lakshminarayanan, and Jasper Snoek. Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, 2019.

Amir Rahimi, Amirreza Shaban, Ching-An Cheng, Richard Hartley, and Byron Boots. Intra order-preserving functions for calibration of multi-class neural networks. *Advances in Neural Information Processing Systems*, 33:13456–13467, 2020.

Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In *International Conference on Machine Learning*, pp. 5389–5400. PMLR, 2019.

Tiago Salvador, Vikram Voleti, Alexander Iannantuono, and Adam Oberman. Improved predictive uncertainty using corruption-based calibration. *arXiv preprint arXiv:2106.03762*, 2021.

Rohan Taori, Achal Dave, Vaishaal Shankar, Nicholas Carlini, Benjamin Recht, and Ludwig Schmidt. Measuring robustness to natural distribution shifts in image classification. *Advances in Neural Information Processing Systems*, 33:18583–18599, 2020.

Junjiao Tian, Dylan Yung, Yen-Chang Hsu, and Zsolt Kira. A geometric perspective towards neural calibration via sensitivity decomposition. *Advances in Neural Information Processing Systems*, 34, 2021.

Christian Tomani, Sebastian Gruber, Muhammed Ebrar Erdem, Daniel Cremers, and Florian Buettner. Post-hoc uncertainty calibration for domain drift scenarios. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10124–10132, 2021.

Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Herve Jegou. Training data-efficient image transformers and distillation through attention. In *International Conference on Machine Learning*, volume 139, pp. 10347–10357, July 2021.

Joost Van Amersfoort, Lewis Smith, Yee Whye Teh, and Yarin Gal. Uncertainty estimation using a single deep deterministic neural network. In *International conference on machine learning*, pp. 9690–9700. PMLR, 2020.

Haohan Wang, Songwei Ge, Zachary Lipton, and Eric P Xing. Learning robust global representations by penalizing local predictive power. In *Advances in Neural Information Processing Systems*, pp. 10506–10518, 2019.

Ximei Wang, Mingsheng Long, Jianmin Wang, and Michael I Jordan. Transferable calibration with lower bias and variance in domain adaptation. *arXiv preprint arXiv:2007.08259*, 2020.

Yeming Wen, Paul Vicol, Jimmy Ba, Dustin Tran, and Roger Grosse. Flipout: Efficient pseudo-independent weight perturbations on mini-batches. *arXiv preprint arXiv:1803.04386*, 2018.

Jonathan Wenger, Hedvig Kjellström, and Rudolph Triebel. Non-parametric calibration for classification. In *International Conference on Artificial Intelligence and Statistics*, pp. 178–190. PMLR, 2020.

Ross Wightman. Pytorch image models. `https://github.com/rwightman/pytorch-image-models`, 2019.

Bianca Zadrozny and Charles Elkan. Obtaining calibrated probability estimates from decision trees and naive bayesian classifiers. In *Icml*, volume 1, pp. 609–616. Citeseer, 2001.

Bianca Zadrozny and Charles Elkan. Transforming classifier scores into accurate multiclass probability estimates. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 694–699, 2002.

APPENDIX

We first introduce the experimental setup including training details, dataset split, and computation resources. We also report more metrics (*i.e.,* KSE (Gupta et al., 2020) and BS (Brier, 1950)) in Table A1 and detailed statistical test results of Table 1. Then, we provide more comparative results with Perturbation (Tomani et al., 2021) in Table A5, and we report full results on CIFAR-10-C and ImageNet-C in Table A6 and Table A7, respectively. Lastly, we give more component analysis of the proposed ACE method in Section D.

A EXPERIMENTAL SETUP

A.1 CIFAR-10 SETUP

Follow the protocol in (Guo et al., 2017; Krishnan & Tickoo, 2020), we use $5,000$ images from the training set of CIFAR-10 as the calibration set. We use ResNet-20 designed for CIFAR-10 and train it using publicly available codes in (Krishnan & Tickoo, 2020).

A.2 IMAGENET SETUP

Follow the protocol in (Guo et al., 2017), we divide the validation set of ImageNet into two halves: one for in-distribution test; the other for learning calibration methods. We use ResNet-50, ResNet-152, Vit-Small-Patch32-224 and Deit-Small-Patch16-224. Their weights are publicly provided by PyTorch Image Models (timm-0.5.4) (Wightman, 2019).

A.3 BASELINE METHODS

Our proposed ACE method is used for improving post-hoc methods (*i.e.*, Vector Scaling, Temperature Scaling, and Spline) on OOD test sets. For each baseline, we use the publicly available codes to train the calibration model. We follow the code and use the same training settings (such as regularization, training scheduler, and training hyper-parameters). The codes we used are:

Vector Scaling: https://github.com/saurabhgarg1996/calibration

Temperature Scaling: https://github.com/gpleiss/temperature_scaling

Spline: https://github.com/kartikgupta-at-anu/spline-calibration

A.4 MORE METRICS FOR TABLE 1

We report the ECE (%) result in Table 1. To better prove the effectiveness of our method, we report another two metrics: KSE (%) (Gupta et al., 2020) and Brier Score (%) in Table A1. Table A1 shows that our method is also effective with these two metrics.

A.5 THE STATISTICAL SIGNIFICANCE TEST IN TABLE 1

We adopt the two-sample t-test, which tells whether the performance of the baseline and baseline + ACE has a significant difference. All methods are run for 5 times based on 5 random seeds $(1, 2, 3, 4, 5)$.

Given a random seed, we use it to randomly downsample the hard calibration set from the original validation set. For all random seeds, the samples for the baseline are indeed the same. However, when training a calibrator, every mini-batch is randomly sampled and shuffled, thus resulting in randomness. As reported in Table A2 of the main paper (mean and standard deviation of ECE), the impact of different random seeds is slight.

A.6 COMPUTATION RESOURCE

We use the Pytorch-1.9.1 framework and run all the experiment on one GPU (GeForce RTX 2080 Ti). The CPU is 24 Intel(R) Core(TM) i9-10920X CPU @ $3.50G$Hz.

Table A1: We used two other metrics, Brier Score (%) and KS-Error (%) (Gupta et al., 2020). We evaluate two calibrators (Temperature Scaling and Spline). All other settings remain the same with Table 1

Metric	Methods	ImgNet-V2-A	ImgNet-V2-B	ImgNet-V2-C	ImgNet-S	ImgNet-R	ImgNet-Adv
	UnCal	5.2260	9.5910	4.0399	24.6331	17.8626	50.8544
	Temp.Scaling	4.0937	1.1129	0.8773	15.7880	10.4752	42.6302
KSE	+ACE	3.0661	0.7809	0.8406	**1.0386**	6.7335	38.0691
	Spline	4.4217	1.0765	0.8813	19.6394	13.0808	45.3623
	+ACE	**1.2029**	**0.7239**	**0.3483**	5.8538	**3.5370**	**31.1308**
	UnCal	15.7902	13.0527	11.1197	21.6672	18.0285	39.1104
	Temp.Scaling	14.8083	12.6830	10.9561	17.2627	15.2080	30.3974
BS	+ACE	14.7192	12.6815	10.9532	15.3793	**14.3487**	26.2166
	Spline	14.8779	**12.5798**	10.8702	18.9953	16.1986	32.0494
	+ACE	**14.7086**	12.5804	**10.8640**	**14.9486**	14.6938	**18.8537**

Table A2: The *t-statistic* and p *values* of the two-sample t-test method in Table 1 of main paper. We report the resulting statistics and p values here, which are one-on-one corresponded to the numbers in Table 1. We regard $p < 0.05$ as statistically significant.

Methods		ImgNet-V2-A	ImgNet-V2-B	ImgNet-V2-C	ImgNet-S	ImgNet-R	ImgNet-Adv
Vector Scaling	t-statistic	59.25	37.39	-25.14	355.60	170.03	217.22
	p	$7.31e^{-12}$	$2.87e^{-10}$	$6.70e^{-9}$	$4.37e^{-18}$	$1.60e^{-15}$	$2.25e^{-16}$
Temp. Scaling	t-statistic	615.89	249.42	-195.10	1164.86	800.82	898.46
	p	$5.40e^{-20}$	$7.47e^{-17}$	$5.33e^{-16}$	$3.30e^{-22}$	$6.62e^{-21}$	$2.63e^{-21}$
Spline	t-statistic	120.74	-28.46	60.99	294.01	675.16	109.61
	p	$2.47e^{-14}$	$2.50e^{-9}$	$5.80e^{-12}$	$2.00e^{-17}$	$2.59e^{-20}$	$5.36e^{-14}$

Table A3: The *t-statistic* and p *values* of the Welch's t-test in Table 1 of main paper. We report the resulting statistics and p values here, which are one-on-one corresponded to the numbers in Table 1. We regard $p < 0.05$ as statistically significant.

Methods		ImgNet-V2-A	ImgNet-V2-B	ImgNet-V2-C	ImgNet-S	ImgNet-R	ImgNet-Adv
Vector Scaling	t-statistic	59.25	37.39	-25.14	355.60	170.03	217.22
	p	$4.85e^{-7}$	$3.05e^{-6}$	$1.48e^{-5}$	$3.75e^{-10}$	$7.17e^{-9}$	$2.68e^{-9}$
Temp. Scaling	t-statistic	615.89	249.42	-195.10	1164.86	800.82	898.46
	p	$4.16e^{-11}$	$1.55e^{-9}$	$4.14e^{-9}$	$3.25e^{-12}$	$1.45e^{-11}$	$9.20e^{-12}$
Spline	t-statistic	120.74	-28.46	60.99	294.01	675.16	109.61
	p	$2.82e^{-8}$	$9.06e^{-6}$	$4.32e^{-7}$	$8.02e^{-10}$	$2.88e^{-11}$	$4.15e^{-8}$

A.7 DATASETS

We list the links of the used datasets and check carefully their licenses for our usage.

ImageNet-Validation (Deng et al., 2009) (https://www.image-net.org);
ImageNet-V2-A/B/C (Recht et al., 2019) (https://github.com/modestyachts/ImageNetV2);
ImageNet-Corruption (Hendrycks & Dietterich, 2019) (https://github.com/hendrycks/robustness);
ImageNet-Sketch (Wang et al., 2019) (https://github.com/HaohanWang/ImageNet-Sketch);
ImageNet-Adversarial (Hendrycks et al., 2021b) (https://github.com/hendrycks/natural-adv-examples);
ImageNet-Rendition (Hendrycks et al., 2021a) (https://github.com/hendrycks/imagenet-r);
CIFAR-10 (Krizhevsky et al., 2009)(https://www.cs.toronto.edu/ kriz/cifar.html);

CIFAR-10**-C** (Hendrycks & Dietterich, 2019)(https://github.com/hendrycks/robustness);

B More Comparison

B.1 Comparison with Perturbation

In Table A4, we compare our method with a recent OOD calibration method Perturbation (Tomani et al., 2021). In Table A4, we observe that Perturbation improves the baselines on Level 5 of ImageNet- C. In fact, these test sets contain data that are seriously out of distribution. However, for datasets that lean towards being in-distribution, *e.g.*, Level 1 in ImageNet-C, Perturbation worsens the baselines. A probable reason is that the diverse calibration set where Perturbation is trained is closer to heavily OOD data (Level-5). In comparison, our method adapts to various test sets through the weighting scheme and yields improvement with statistical significance in most test cases.

Table A4: Method comparison on **ImageNet-C** datasets Hendrycks & Dietterich (2019). We report ECE (%) for top-1 predictions (in %) of the ResNet-152 model. For each level of corruption (column), we report the average ECE using 25 bins with lowest numbers in **bold** and second lowest underlined. ACE improves calibration performance of two post-hoc calibration methods on all datasets.

	Corruption Intensity				
Method	Level 1	Level 2	Level 3	Level 4	Level 5
Uncalibrated	6.0684	7.8617	9.7938	12.3911	15.5049
Temperature Scaling (TS)	2.4880	**2.7976**	3.7996	5.1836	7.7213
Temperature + Perturbation	9.3084	8.6574	7.6707	5.7594	4.3672
Temperature + ACE	2.9733	3.1130	**3.1306**	**3.1494**	4.3034
Spline	**1.8049**	3.1690	5.2388	7.8672	11.0547
Spline + Perturbation	9.6207	8.1570	6.7643	5.1064	5.2777
Spline + ACE	3.6982	4.2046	4.2944	3.7231	**3.9707**

B.2 Comparison with TransCal

In Table A5, we compare our method with a recent OOD calibration method TranCal (Wang et al., 2020). In Table A5, we observe that TransCal is inferior to our method on the ImageNet-S dataset with ResNet-50.

Table A5: Method comparison on **ImageNet-V2-A, ImageNet-V2-B, ImageNet-V2-C, and ImageNet-S** datasets. Following the protocol in (Wang et al., 2020), we report ECE (%) for top-1 predictions (in %) of the ResNet-50 model.

Method	ImageNet-V2-A	ImageNet-V2-B	ImageNet-V2-C	ImageNet-S
Uncalibrated	9.50	6.23	4.31	22.32
Temperature Scaling	4.44	2.73	1.68	16.27
TransCal	12.26	4.43	1.86	8.10
Ours	3.56	2.56	1.70	7.53

C Full Results on ImageNet-C and CIFAR-10-C

In the Table 3 of the main paper, we report the mean ECE (%) across 16 different types of data shift at intensity 5. In addition, we report the complete ECE results on CIFAR-10-C and ImageNet-C at intensity 5 in Table A6 and Table A7. We observe that our method effectively improves the baselines

(Spline) and gives state-of-the-art calibration accuracy under 2 out of 3 quartiles and mean value on both CIFAR-10-C and ImageNet-C.

Table A6: Full results on CIFAR-10-C datasets (Hendrycks & Dietterich, 2019). We report the lower quartile (25-th percentile), median (50-th percentile), mean and upper quartile (75-th percentile) of ECE computed across 16 different types of data shift at intensity 5 with lowest numbers in **bold** and second lowest underlined.

Metric		Method								
		Vanilla	Temp Scaling	Ensemble	SVI	LL SVI	SVI -AvUTS	SVI -AvUC	Spline	Spline +Ours
ECE	lower quartile	0.2121	0.0997	0.0549	0.0925	0.2027	<u>0.0466</u>	**0.0398**	0.2045	0.0846
	median quartile	0.3022	0.1834	0.1054	0.2146	0.3077	0.1516	<u>0.1107</u>	0.3007	**0.1078**
	mean	0.3151	0.1993	0.1611	0.2389	0.3267	0.1585	<u>0.1374</u>	0.3382	**0.1264**
	upper quartile	0.4148	0.2915	0.2551	0.3636	0.4246	0.2345	<u>0.2303</u>	0.4376	**0.1519**

Table A7: Full results on ImageNet-C datasets (Hendrycks & Dietterich, 2019). We report the lower quartile(25-th percentile), median (50-th percentile), mean and upper quartile (75-th percentile) of ECE computed across 16 different types of datashift at intensity 5 with lowest numbers in **bold** and second lowest underlined.

Metric		Method								
		Vanilla	Temp Scaling	Ensemble	SVI	LL SVI	SVI -AvUTS	SVI -AvUC	Spline	Spline +Ours
ECE	lower quartile	0.1244	0.0959	0.0503	0.0722	0.1212	0.0420	<u>0.0319</u>	0.0575	**0.0233**
	median quartile	0.1737	0.1392	0.0900	0.1144	0.1684	0.0807	**0.0447**	0.1143	<u>0.0452</u>
	mean	0.1942	0.1600	0.0880	0.1188	0.1868	0.0800	<u>0.0542</u>	0.1147	**0.0477**
	upper quartile	0.2744	0.2364	0.1264	0.1723	0.2676	0.1275	<u>0.0696</u>	0.1363	**0.0606**

D More Component Analysis

D.1 Comparing Fixed Weighting Schemes With the Adaptive Weight

In this section, we compare the adaptive weight ($\alpha = \frac{\text{avgConf}(\mathcal{D}_{test})}{\text{avgConf}(\mathcal{D}_o)}$) with setting α to fixed values 0.2, 0.5, and 0.9. The difficulty level for the out-of-distribution situation is 10. We evaluate the three calibration baselines on the six out-of-distribution test sets and one in-distribution test set using ResNet-152 as backbone and use the mean ECE value over all the seven test sets (six OOD datasets and one ID test set) as evaluation metric.

Fig. 4 (left) indicates that for the Spline method, setting α to 0.2 and 0.5 deteriorates the Spline baseline, while $\alpha = 0.9$ improves it. Because they are agnostic about test sets, setting fixed values might work in some proper cases and be less useful in others. Our design (α) is shown to improve the baselines on various OOD test sets and seems to be superior to fixed values (under Spline).

In Table A8, we also report the value of the adaptive α used in Table 1 and Table 2.

D.2 Test Data Are Given in Batch

In real-world scenarios, test data may not all be accessible. Here we study how the calibration performance changes when test data are given in batches of various sizes. In Fig. 4 (right), α is calculated from test batches of various sizes. We observe our method still achieves improvement over the temperature scaling baseline and has similar ECE with the method computed on the full test set under reasonably large batch sizes (≥ 64).

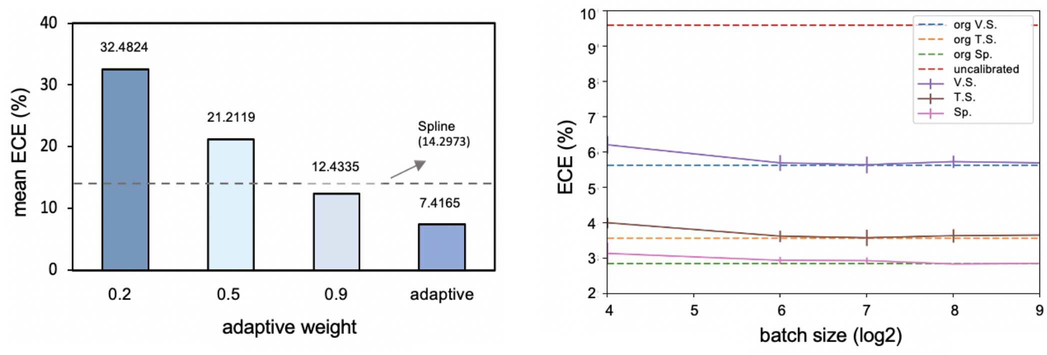

Figure 4: **Left**: Comparison of different weighting schemes for ACE. We report the mean ECE (%) on six OOD datasets (ImageNet-V2-A/B/C, ImageNet-S, ImageNet-Adv, ImageNet-R) and one ID test set (ImageNet-Val). Spline is used as the calibration baseline. The ResNet-152 model is used. **Right**: Effectiveness of α computed in test batches of different sizes (16, 64, 128, 256 and 512). Comparing with calculating α on the full set, using test batches yields similar ECE (%) especially when the batch size is at least 64. Vector Scaling (V.S.), Temperature scaling (T.S.), Spline (Sp.) are used as the baseline calibrator for our ACE. We also include the original baseline results in the figure (*e.g.*, org T.S.).

Table A8: The adaptitve α that we adopt in Table 1 and Table 2.

Model	ImgNet-Val	ImgNet-V2-A	ImgNet-V2-B	ImgNet-V2-c	ImgNet-S	ImgNet-R	ImgNet-Adv
ResNet	0.994080	0.918328	0.972311	0.989697	0.63765	0.709984	0.682187
Vit	0.998655	0.896980	0.969018	0.98561	0.538366	0.674307	0.637850
Deit	0.998741	0.912270	0.967555	0.999048	0.612748	0.648445	0.618136

D.3 An Alternative Method

In L210-216 of the main paper, we mentioned that a possible way to calibrate OOD data is to estimate its difficulty and create a calibration set that has a closer difficulty level with the OOD test dataset. Moreover, according to Sec. 3.5 of the main paper, the average confidence score could serve as an unsupervised indicator to the degree of how out-of-distribution a test set is (Guillory et al., 2021). Here, we propose another post-hoc calibration method for OOD calibration. Specifically, we first estimate the error rate of a test set (Garg et al., 2022):

$$error_{\mathcal{D}_{test}} = (1 - \text{Acc}(\mathcal{D}_o)) + (\text{avgConf}(\mathcal{D}_o) - \text{avgConf}(\mathcal{D}_{test})). \tag{7}$$

Thus, we can compute $d_{\mathcal{D}_{test}}$ as:

$$d_{\mathcal{D}_{test}} = \frac{error_{\mathcal{D}_{test}}}{1 - error_{\mathcal{D}_{test}}}. \tag{8}$$

According to Table A9, our estimation method is also shown to be effective. Specifically, it has the second lowest ECE on CIFAR-10-C and is only 0.0034 higher than SVI-AvUC on ImageNet-C.

D.4 $\mathcal{D}_o$ and $\mathcal{D}_h$ have the same number of samples for tuning the function.

The size of $\mathcal{D}_h$ in our submission is $5,884$. We randomly sample the easy calibration set $\mathcal{D}_o$ into the same size ($5,884$), the difficulty of which remains the same due to random sampling. We report performance calibration (ECE, %) of Temperature Scaling and our improved version on all the seven test sets below. The ResNet-152 classifier is used. The results in Table A10 show that our method remains beneficial, i.e., achieving lower ECE when combined with Temperature Scaling, when the easy and the hard calibration sets have the same size. The results show that our method remains beneficial, *i.e.,* achieving lower ECE when combined with Temperature Scaling, when the easy and the hard calibration sets have the same size.

Table A9: Method comparison on CIFAR-10-C and ImageNet-C datasets with ResNet-20 and ResNet-50, respectively. Following the protocol in (Ovadia et al., 2019), we report mean ECE (%) across 16 different types of data shift at intensity 5 with lowest numbers in **bold** and second lowest underlined.

Dataset	Vanilla	SVI	SVI -AvUC	Spline	Spline +ACE	Spline +Estimation
CIFAR-10-C	0.1942	0.2389	0.1374	0.3382	**0.1264**	<u>0.1298</u>
ImageNet-C	0.3151	0.1188	<u>0.0542</u>	0.1147	**0.0477**	0.0576

Table A10: Calibration performance of our method integrated with Temperature Scaling on one in-distribution test set and six OOD test sets. ECE (25bins, %) for top-1 predictions. Here we $\mathcal{D}_o$ with the sample size of $\mathcal{D}_h$ $(5,884)$.

Method	ImgNet-Val	ImgNet-V2-A	ImgNet-V2-B	ImgNet-V2-c	ImgNet-S	ImgNet-R	ImgNet-Adv
Temp.Scaling	1.9670	4.3571	2.7234	1.7880	15.6735	10.3832	42.5225
+ACE	1.9623	3.4842	2.5458	1.6764	10.3131	6.6726	37.9957

D.5 The original calibration set $\mathcal{D}_o$ is not easy.

In Sec. 3.5, we mentioned that difficulty is a relative concept and depends on the classifier. Note that for a weaker classifier, a certain dataset will be harder. With this in mind, we experimented with two weaker classifiers, (i.e., harder $\mathcal{D}_o$) and observed that our method is still effective. Specifically, we adopt LCNet-050 and TinyNet-E, which have 60.094% and 59.856% top-1 accuracy, respectively on the ImageNet-Val dataset. We apply Temperature Scaling with the proposed method to the two classifiers and report calibration performance (ECE, %) below. These results show that our method consistently improves Temperature Scaling when the "easy calibration set" has high difficulty (*i.e.,* is not easy).

Table A11: Calibration performance of our method integrated with Temperature Scaling on one in-distribution test set and six OOD test sets. ECE (25 bins, %) for top-1 predictions. We use LCNet-050 and TinyNet-E, which have 60.094% and 59.856% top-1 accuracy, respectively on the validation set of ImageNet dataset. (Note IN is short for ImageNet)

Model	Method	IN-Val	IN-V2-A	IN-V2-B	IN-V2-C	IN-S	IN-R	IN-Adv
LCENet-050	Temp.Scaling	1.8293	6.6047	2.9681	1.6949	20.3415	18.9839	43.1683
	+ACE	1.8238	4.8591	2.2639	1.7516	14.0055	15.3397	39.2584
TinyNet-E	Temp.Scaling	1.3888	6.8949	2.7991	1.7194	22.4438	20.7810	41.3513
	+ACE	1.3857	5.4262	2.4606	1.8311	17.1741	17.7259	38.0800

D.6 MORE TYPES OF OOD TEST SETS.

We further provide the calibration results (ECE, %) on another challenging and diverse dataset iWildCam-WILDS (Koh et al., 2021) with the ResNet-50 classifier. iWildCam-WILDS is an animal species classification dataset, where the distribution shift arises due to changes in camera angle, lighting, and background. Tabel A12 shows that our method can also improve the calibration performance on iWildCam-WILDS, especially, improves temperature scaling by 0.9% decrease in ECE on the OOD test set.

Table A12: Calibration performance of our method integrated with Temperature Scaling and Spline on the in-distribution and OOD iWildCam-WILDS dataset. ECE (25bins, %) for top-1 predictions and ResNet-50 classifier is used.

Dataset	Uncal.	Temp.Scaling	Temp.Scaling+Ours	Spline	Spline+Ours
iWildCam-WILDS-ID	14.2701	2.6786	2.5833	3.8142	3.6965
iWildCam-WILDS-OOD	13.5552	4.8231	3.9738	4.9902	4.8425