The Pitfalls and Promise of Conformal Inference Under Adversarial Attacks (*Preliminary Version*)

Ziquan Liu [1] Yufei Cui [2] Yan Yan [3] Yi Xu [4] Xiangyang Ji [5] Xue Liu [2] Antoni B. Chan [6]

Abstract

In safety-critical applications such as medical imaging and autonomous driving, where decisions have profound implications for patient health and road safety, it is imperative to maintain both high adversarial robustness to protect against potential adversarial attacks and reliable uncertainty quantification in decision-making. With extensive research focused on enhancing adversarial robustness through various forms of adversarial training (AT), a notable knowledge gap remains concerning the uncertainty inherent in adversarially trained models. To address this gap, this study investigates the uncertainty of deep learning models by examining the performance of conformal prediction (CP) in the context of standard adversarial attacks within the adversarial defense community. It is first unveiled that existing CP methods do not produce informative prediction sets under the commonly used l_∞-norm bounded attack if the model is not adversarially trained, which underpins the importance of adversarial training for CP. Our paper next demonstrates that the prediction set size (PSS) of CP using adversarially trained models with AT variants is often worse than using standard AT, inspiring us to research into CP-efficient AT for improved PSS. We propose to optimize a Beta-weighting loss with an entropy minimization regularizer during AT to improve CP-efficiency, where the Beta-weighting loss is shown to be an upper bound of PSS at the population level by our theoretical analysis. Moreover, our empirical study on four image classification across three popular AT baselines datasets validates the effectiveness of the proposed Uncertainty-Reducing AT (AT-UR).

1. Introduction

The research into adversarial defense has been focused on improving adversarial training with various strategies, such as logit-level supervision (Zhang et al., 2019; Cui et al., 2021a) and loss re-weighting (Wang et al., 2019; Liu et al., 2021a). However, the predictive uncertainty of an adversarially trained model is a crucial dimension of the model in safety-critic applications such as healthcare (Razzak et al., 2018), and is not sufficiently understood. Existing works focus on calibration uncertainty (Stutz et al., 2020; Qin et al., 2021; Kireev et al., 2022), without investigating a practical uncertainty quantification of a model, e.g., a prediction set in image classification (Shafer & Vovk, 2008; Angelopoulos et al., 2020; Romano et al., 2020).

On the other hand, the research into conformal prediction (CP) has been extended to non-i.i.d. (identically independently distributed) settings, including distribution shifts (Gibbs & Candes, 2021) and toy adversarial noise (Ghosh et al., 2023; Gendler et al., 2021). However, there is little research work on the performance of CP under standard adversarial attacks in the adversarial defense community, such as PGD-based attacks (Madry et al., 2018; Croce & Hein, 2020) with l_∞-norm bounded perturbations. For example, (Gendler et al., 2021) and (Ghosh et al., 2023) only consider l_2-norm bounded adversarial perturbations with a small attack budget, e.g., $\epsilon = 0.125$ for the CIFAR dataset (Krizhevsky et al., 2009). In contrast, the common l_2-norm bounded attack budget in adversarial defense community reaches $\epsilon = 0.5$ on CIFAR (Croce & Hein, 2020). In other words, existing research on adversarially robust conformal prediction is not practical enough to be used under standard adversarial attacks.

In this context, our paper is among the first research papers to explore uncertainty of deep learning models within the framework of CP in the presence of a *standard* adversary. We first present an empirical result that shows the failure of three popular CP methods on non-robust models under a standard adversarial attack, indicating the necessity of using adversarial training (AT) during the training stage. Next, we show the CP performance of three popular AT methods and find that advanced AT methods like TRADES (Zhang et al., 2019) and MART (Wang et al., 2019) substantially increase

[1] Queen Mary University of London [2] McGill University, Mila [3] Washington State University [4] Dalian University of Technology [5] Tsinghua University [6] City University of Hong Kong. Correspondence to: Ziquan Liu <ziquan.liu@qmul.ac.uk>.

Proceedings of the 41^{st} International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

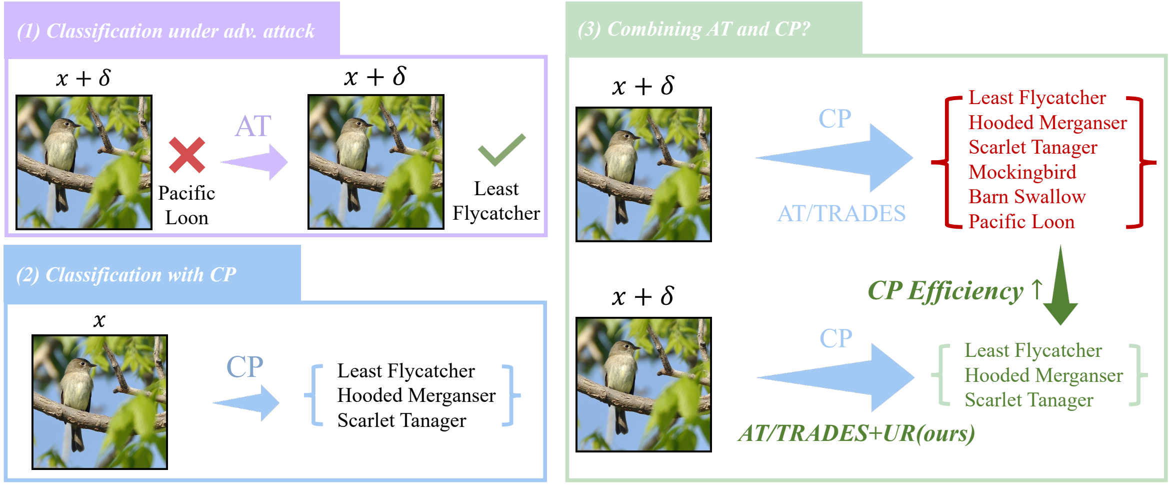

Figure 1. The proposed uncertainty-reducing adversarial training (AT-UR) improves the CP-efficiency of existing adversarial training methods like AT, FAT and TRADES. (1) AT improves the Top-1 robust accuracy of a standard model; (2) CP generates a prediction set with a pre-specified coverage guarantee for an input image, but for models not adversarially trained, CP fails to generate informative prediction sets, as the PSS is almost the same as the class number, when models being attacked (Fig. 2); (3) When using CP in an adversarially trained model, the prediction set size is generally large, leading to inefficient CP. Our AT-UR substantially improves the CP-efficiency of existing AT methods.

the PSS in CP even though they improve the Top-1 robust accuracy. This key observation inspires us to develop the uncertainty-reducing AT (AT-UR) to learn an adversarially robust model with improved *CP-efficiency* (Angelopoulos et al., 2020), meaning that CP uses a smaller PSS to satisfy the coverage. The proposed AT-UR consists of two training techniques, Beta weighting and entropy minimization, based on our observation about the two major factors that affect PSS: True Class Probability Ranking (TCPR) and prediction entropy, both defined in Sec. 5. Our theoretical analysis on the Beta-weighting loss reveals that the proposed weighted loss is an upper bound for the PSS at the population level. The proposed AT-UR is demonstrated to be effective at reducing the PSS of models on multiple image classification datasets. In summary, there are four major contributions of this paper.

1. We test several CP methods under commonly used adversarial attacks in the adversarial defense community. It turns out that for models not adversarially trained, CP cannot to generate informative prediction sets. Thus, adversarial training is necessary for CP to work under adversarial attacks.
2. We test the performance of adversarially trained models with CP and demonstrate that improved AT often learns a more uncertain model and leads to less efficient CP with increased PSS.
3. We propose uncertainty-reducing AT (AT-UR) to learn a CP-efficient and adversarially robust model by minimizing the entropy of predictive distributions and a weighted loss where the weight is a Beta density function of TCPR.
4. Our main theorem shows that at the population-level, the Beta-weighting loss is an upper bound for the targeted PSS, so minimizing the weighted loss leads to reduced PSS in theory. This theoretical result corroborates our hypothesis that optimizing the *promising* samples with high weights leads to reduced PSS.
5. Our empirical study demonstrates that the proposed AT-UR learns adversarially robust models with substantially improved CP-efficiency on four image classification datasets across three AT methods, validating our major theoretical result.

The paper has a structure as follows. Section 2 discusses related works and Section 3 introduces mathematical notations and two key concepts in this paper. Section 4 shows the pitfalls of three CP methods under standard attacks when the model is not robustly trained and the low CP-efficiency of two improved AT methods and motivates us to develop the AT-UR introduced in Section 5. Our major empirical results are shown in Section 6 and we conclude the paper in Section 7.

2. Related Works

Adversarial Robustness. The most effective approach to defending against adversarial attacks is adversarial training (AT) (Madry et al., 2018). There is a sequence of works following the vanilla version of AT based on projected gradient descent (PGD), including regularization (Qin et al., 2019; Liu & Chan, 2022; Liu et al., 2021b), logit-level supervision (Zhang et al., 2019; Cui et al., 2021a) and loss re-weighting (Wang et al., 2019; Liu et al., 2021a). Existing methods on regularization focus on improving Top-1 robust accuracy by training the model with certain properties like linearization (Qin et al., 2019) and large margins (Liu & Chan, 2022). In contrast, our work focuses on the PSS, i.e., the efficiency of CP, in adversarially trained models by regularizing the model to have low prediction entropy. The entropy minimization regularization also entails logit-level supervision as in (Zhang et al., 2019). In comparison, our proposed approach, AT-EM, enhances CP efficiency, whereas TRADES (Zhang et al., 2019) impedes CP-efficiency. The most related work is (Gendler et al., 2021) which also studies CP under adversarial attacks. However, there are two fundamental differences: 1) (Gendler et al., 2021) only considers a small attack budget under l_2-norm bounded attacks, while our work investigates CP under common adversarial attacks in adversarial defense literature with l_∞-norm bounded attacks; 2) Our paper shows that AT is essential for CP to work under strong adversarial attacks and proposes novel AT methods to learn a CP-efficient and adversarially-robust model, while (Gendler et al., 2021) only considers the post-training stage. Our experiment validates that (Gendler et al., 2021) fails when there are strong adversarial attacks (Fig. 2).

Uncertainty Quantification. Uncertainty quantification aims to provide an uncertainty measure for a machine learning system's decisions. Within this domain, Bayesian methods stand out as a principled approach, treating model parameters as random variables with distinct probability distributions. This is exemplified in Bayesian Neural Networks (BNNs), which place priors on network weights and biases, updating these with posterior distributions as data is observed (Gal & Ghahramani, 2016; Kendall & Gal, 2017; Cui et al., 2020). However, the large scale of modern neural networks introduces challenges for Bayesian methods, making prior and posterior selection, and approximate inference daunting tasks (Kingma et al., 2015; Cui et al., 2021b; 2023; CUI et al., 2023). This can sometimes compromise the optimal uncertainty quantification in BNNs. In contrast, the frequentist approach offers a more direct route to uncertainty estimation. It views model parameters as fixed yet unknown, deriving uncertainty through methods like conformal prediction (Vovk et al., 1999; Ghosh et al., 2023; Gendler et al., 2021). While Bayesian methods integrate prior beliefs with data, their computational demands in large networks can be overwhelming, positioning the straightforward frequentist methods as a viable alternative for efficient uncertainty quantification. Thus, our paper investigates the uncertainty of adversarially trained models via CP. Note that our work is fundamentally different from existing research on uncertainty calibration for AT (Stutz et al., 2020; Qin et al., 2021; Kireev et al., 2022), as our focus is to produce a valid prediction set while uncertainty calibration aims to align accuracy and uncertainty. Finally, (Einbinder et al., 2022) proposes to train a model with uniform conformity scores on a calibration set in standard training, while our work proposes CP-aware adversarial training to reduce PSS.

3. Preliminary

Before diving into the details of our analysis and the proposed method, we first introduce our mathematical notations, adversarial training and conformal prediction.

Notations. Denote a training set with m samples by $\mathcal{D}_{\text{tr}} = \{(x_i, y_i)\}_{i=1}^m$. Suppose each data sample $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$ is drawn from an underlying distribution $\mathcal{P}$ defined on the space $\mathcal{X} \times \mathcal{Y}$, where x_i and y_i are the feature and label, respectively. Particularly, we consider the classification problem and assume that there are K classes, i.e., $\mathcal{Y} = \{1, ..., K\}$ (we denote $[K] = \{1, ..., K\}$ for simplicity). Let $f_\theta : \mathcal{X} \to \Delta_p^K$ denote a predictive model from a hypothesis class $\mathcal{F}$ that generates a K-dimensional probability simplex: $\Delta_p^K = \{v \in [0,1]^K : \sum_{k=1}^K v_k = 1\}$. θ is the model parameter we optimize during training. A loss function $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ is used to measure the difference between the prediction made by $f_\theta(x)$ and the ground-truth label y.

To measure the performance of f_θ in the sense of population over $\mathcal{P}$, the *true risk* is typically defined as $R(f_\theta) = \mathbb{P}_{(x,y)\sim\mathcal{P}}[f_\theta(x) \neq y]$. Unfortunately, $R(f)$ cannot be realized in practice, since the underlying $\mathcal{P}$ is unreachable. Instead, the *empirical risk* $\widehat{R}(f_\theta) = \frac{1}{m}\sum_{i=1}^m \mathbb{I}[f_\theta(x_i) \neq y_i]$ is usually used to estimate $R(f_\theta)$, where $\mathbb{I}[\cdot]$ is the indicator function. The estimation error of $\widehat{R}(f_\theta)$ to $R(f_\theta)$ is usually referred to as generalization error bound and can be bounded by a standard rate $O(1/\sqrt{m})$. To enable the minimization of empirical risk, a loss function ℓ is used as the surrogate of $\mathbb{I}[\cdot]$, leading to the classical learning paradigm empirical risk minimization (ERM): $\min_{f_\theta \in \mathcal{F}} \widehat{L}(f_\theta) = \frac{1}{m}\sum_{i=1}^m \ell(f_\theta(x_i), y_i)$. In this work, we use the standard cross-entropy loss as the loss function where j is the index for a jth element in a vector,

$$\ell(f_\theta(x_i), y_i) = -\sum_{j=1}^{K} y_{ij} \log(f_\theta(x_i)_j). \tag{1}$$

Adversarial training. Write the loss for sample (x_i, y_i) in adversarial training as $l(\tilde{x}_i, y_i)$, where $\tilde{x}_i = x_i + \delta_i$ and δ_i is generated from an adversarial attack, e.g., PGD attack (Madry et al., 2018). The vanilla adversarial training minimizes the loss with uniform weights for a mini-batch with B samples, i.e.,

$$\nabla f_\theta = \nabla \frac{1}{B} \sum_{i=1}^{B} l(f_\theta(\tilde{x}_i)_j, y_i), \tag{2}$$

where ∇f_θ is the gradient in this mini-batch step optimization with respect to θ.

Conformal prediction (CP). CP is a distribution-free uncertainty quantification method and can be used in a wide range of tasks including both regression and classification (Vovk et al., 1999; 2005). This paper focuses on the image classification task, where CP outputs a prediction set instead of the Top-1 predicted class as in a standard image classification

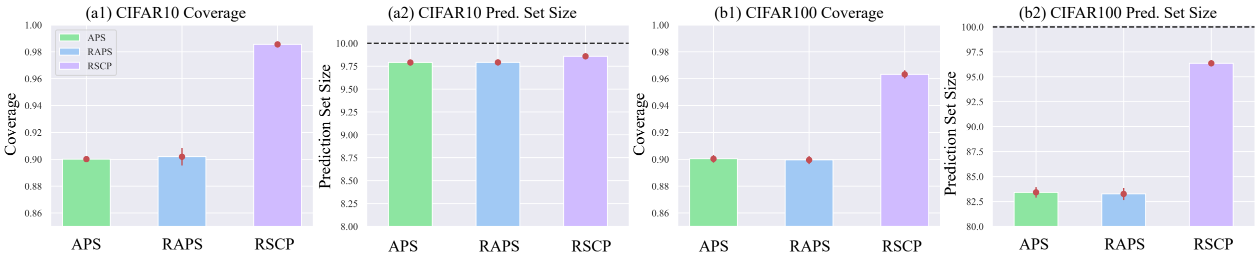

Figure 2. The performance of three representative CP methods using non-robust models under standard adversarial attacks in the adversarial defense community. The red line denotes means standard deviation of the metric. For comparison, the average PSS for normal images is 1.03 and 2.39 for CIFAR10 and CIFAR 100. See Sec. 6.1 for details of the experiment.

	Dataset	CIFAR10	CIFAR100	Caltech256	CUB200
AT	Rob. Coverage	90.55(0.51)	90.45(0.59)	91.35(0.85)	90.33(0.89)
	Rob. Set Size	**3.10(0.07)**	**23.79(0.80)**	**43.20(2.11)**	**37.37(2.11)**
	Clean Acc.	89.76(0.15)	68.92(0.38)	75.28(0.51)	65.36(0.27)
	Rob. Acc.	50.17(0.91)	28.49(1.14)	47.53(0.67)	26.29(0.44)
TRADES	Rob. Coverage	90.72(0.62)	90.35(0.57)	90.82(0.81)	90.38(0.76)
	Rob. Set Size	3.31(0.09)	27.60(0.97)	44.80(3.42)	52.18(2.60)
	Clean Acc.	87.31(0.27)	62.83(0.33)	69.57(0.25)	58.16(0.38)
	Rob. Acc.	53.07(0.23)	32.07(0.20)	47.07(0.37)	27.82(0.23)
MART	Rob. Coverage	91.60(0.48)	90.67(0.83)	91.92(0.84)	90.22(0.53)
	Rob. Set Size	3.81(0.07)	28.37(1.29)	46.79(2.73)	45.31(1.78)
	Clean Acc.	85.43(0.24)	59.66(0.26)	69.68(0.31)	58.72(0.18)
	Rob. Acc.	**54.48(0.29)**	**34.04(0.46)**	**49.82(0.32)**	**28.99(0.25)**

Table 1. CP and Top-1 accuracy of three popular adversarial defense methods under PGD100 adversarial attack. Bold numbers are the best PSS and robust accuracy.

model, and satisfies a coverage guarantee. Mathematically, CP maps an input sample x to a prediction set $\mathcal{C}(x)$, which is subset of $[K] = \{1, \cdots, K\}$, with the following coverage guarantee,

$$P(y \in \mathcal{C}(x)) \geq 1 - \alpha, \tag{3}$$

where $1 - \alpha$ is a pre-defined confidence level such as 90%, meaning that the prediction set will contain the ground-truth label with 90% confidence for future data. This paper mainly considers the *split conformal prediction*, an efficient CP approach applicable to any pre-trained black-box classifier (Papadopoulos et al., 2002; Lei et al., 2018) as it does not need to re-train the classifier with different train-calibration-test splits.

The prediction set of CP is produced by the calibration-then-test procedure. In the context of a classification task, we define a prediction set function $\mathcal{S}(x, u; \pi, \tau)$, where u is a random variable sampled from a uniform distribution Uniform$[0, 1]$ independent of all other variables, π is shorthand for the predictive distribution $f_\theta(x)$, and τ is a threshold parameter that controls the size of the prediction set. An increase in the value of τ leads to an expansion in the size of the prediction set within $\mathcal{S}(x, u; \pi, \tau)$. We give one example (Romano et al., 2020) of the function $\mathcal{S}$ in Appendix A. The calibration process computes the smallest threshold parameter $\hat{\tau}_{\text{cal}}$ to achieve an empirical coverage of $(1 - \alpha)(n_c + 1)/n_c$ on the calibrations set with n_c samples. For a test sample x^*, a prediction set is the output of the function $\mathcal{S}(x^*, u; \pi^*, \hat{\tau}_{\text{cal}})$.

4. Necessitate AT for robust and efficient coverage.

The pitfalls of CP under strong adversarial attacks. We test the performance of three conformal prediction methods, i.e., APS (Adaptive Prediction Sets) (Romano et al., 2020), RAPS (Regularized Adaptive Prediction Sets) (Angelopoulos et al., 2020) and RSCP (Randomly Smoothed Conformal Prediction) (Gendler et al., 2021), under standard adversarial attacks. Specifically, for APS and RAPS, we use PGD100 adversarial attacks with l_∞-norm bound and attack budget $\epsilon = 8/255 = 0.0314$. For RSCP, we adopt PGD20 with an l_2-norm bound, in accordance with the original paper's settings, but with a larger attack budget of $\epsilon = 0.5$ as in RobustBench (Croce & Hein, 2020). If not specified otherwise, we use adversarial attack PGD100 with l_∞ norm and $\epsilon = 8/255 = 0.0314$ to generate adversarial examples throughout this paper.

Fig. 2 shows the coverage and PSS of three CP methods on CIFAR10 and CIFAR100 when models are trained in a standard way, i.e., without adversarial training. Although all CP methods have good coverages, their PSS' are close to the number of classes in both datasets as the classifier is completely broken under the strong adversarial attacks. In contrast, when the same models are applied to standard images, the PSS are 1.03 and 2.39 for CIFAR10/CIFAR100. This result reveals that adversarial training is indispensable if one wants to use CP to get reasonable uncertainty quantification for their model in an adversarial environment. Therefore, in next section, we test AT and two improved AT methods to investigate the performance of CP for adversarially trained models.

Improved AT Compromises Conformal Prediction's Efficiency. We test three popular adversarial training methods, i.e., AT (Madry et al., 2018), TRADES (Zhang et al., 2019) and MART (Wang et al., 2019), using APS as the conformal prediction method under a commonly used adversarial attack, PGD100 with l_∞-norm and $\epsilon = 8/255 = 0.0314$. See more detailed experimental settings in Sec. 6. Tab. 1 shows their coverage and PSS, as well as clean and robust accuracy on four datasets. The results demonstrate that while the two enhanced adversarial training methods, TRADES

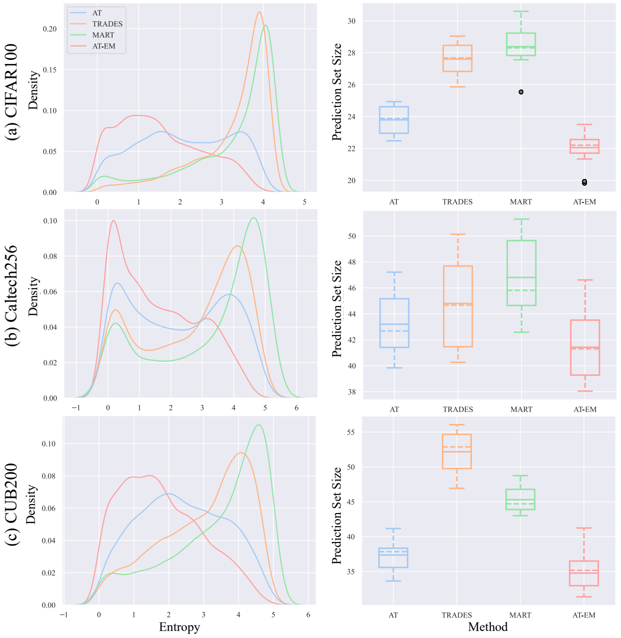

Figure 3. **(Left)**: The kernel density estimation for predictive distribution's entropy on adversarial test sets. **(Right)**: Box plot of PSS of three AT baselines and AT-EM. AT-EM effectively controls prediction entropy and improves CP-efficiency. See Tab. 1 and Tab. 3 for their coverages.

and MART, effectively improve the Top-1 accuracy in the presence of adversarial attacks, they lead to an increase in the size of the prediction set, consequently yielding a less CP-efficient model. In other words, the improvement in Top-1 accuracy does not necessarily lead to less uncertainty. Therefore, to design a new AT method that learns an adversarially robust model with efficient CP, a deep investigation into the PSS is necessary. In the following section, we identify two major factors that play an important role in controlling the PSS through our empirical study.

5. Uncertainty-Reducing Adversarial Training

This section investigates two factors highly correlated with PSS and introduces our uncertainty-reducing adversarial training method.

5.1. Entropy Minimization for CP-Efficiency

The PSS is closely related to the entropy of prediction distribution, as both quantities reflect the prediction uncertainty of a model. A more uniform categorical distribution has higher uncertainty, which is reflected in its higher entropy. Fig. 3 visualizes the kernel density estimation (KDE) (Rosenblatt, 1956; Parzen, 1962) of entropy values calculated with adversarial test samples on three datasets. It is evident that TRADES and MART learn models with predictive distributions that have higher entropy values than AT, thus increasing the PSS comparatively.

To decrease the PSS of AT, we add an entropy minimization term to the loss function,

$$\ell_{\text{EM}}(f_\theta(x_i), y_i) = -\sum_{j}^{K} y_{ij} \log(f_\theta(x_i)_j) + \lambda_{\text{EM}} H(f_\theta(x_i)), \tag{4}$$

where the regularization is the entropy function $H(f_\theta(x_i)) = -\sum_j^K f_\theta(x_i)_j \log(f_\theta(x_i)_j)$. We set λ_{EM}=0.3 in all of our experiment based on a hyper-parameter search experiment on CIFAR100, where $\lambda_{\text{EM}} \in \{0.1, 0.3, 1.0, 3.0\}$. The AT scheme with entropy minimization (EM) is denoted as AT-EM. This entropy term is the same as the entropy minimization in semi-supervised learning (Grandvalet & Bengio, 2004). However, note that our work is the first to use entropy minimization in adversarial training for improving CP-efficiency. Fig. 3 also shows the KDE of entropy values on adversarial test sets using AT-EM. The reduction in predictive entropy effectively leads to a substantial decrease in the PSS of AT-EM.

The second factor that affects PSS is the distribution of True Class Probability Ranking (TCPR) on the test dataset. The TCPR is defined as the ranking of a sample x's ground-truth class probability among the whole predictive probability. In equation, we sort π with the descending order into $\hat{\pi}$,

$$\hat{\pi} = \{\pi_{(1)}, \cdots, \pi_{(K)}\}, \tag{5}$$

where $\pi_{(j)} \geq \pi_{(j+1)}, \forall j = 1, \cdots, K-1$, and (j) is the sorted index. TCPR is the index j in $\hat{\pi}$ corresponding to the ground-truth label y, i.e., $Sort(y) = j$.

The TCPR matters to the PSS as we observe that a model with higher robust accuracy does not necessarily have a smaller PSS as shown in Tab. 1. This discovery indicates that improving Top-1 accuracy, i.e., the percentage of samples with TCPR=1, is not enough to learn a CP-efficient model. In particular, the model capacity might be not strong enough to fit all the adversarial training data or achieve 100% adversarial training accuracy as a result of a strong adversary and high task complexity, e.g., a large number of classes. For instance, on CIFAR100, the robust accuracy on training data of a pre-trained ResNet50 is only around 45% after 60 epochs of fine-tuning.

Motivated by this observation, we propose to use a Beta distribution density function (Fig. 4) to weight the loss samples so that the TCPR distribution shifts towards the lower TCPR region. This design embodies our intuition that the training should focus on samples with *promising* TCPR's, whose TCPR's are neither 1 nor too large, because TCPR=1 means the sample is correctly classified and a large TCPR means the sample is an outlier and probably hopeless to learn. Those samples with promising TCPR's are important to control PSS as they are the *majority* of the dataset and thus largely affect the averaged PSS, see Fig. 8 for the percentage of promising samples throughout AT training on CIFAR100.

With the previous intuition, we propose an importance weighting scheme based on Beta distribution density function of TCPR to learn a CP-efficient model. Let the TCPR of sample $\tilde{x}_i$ be $r_i \in [K]$ and the normalized TCPR be $\hat{r}_i \in (0, 1]$. Note that in our implementation we use the index starting from 0 instead of 1, so $\hat{r}_i \in [0, 1)$ in practice. We use the Beta distribution density function, e.g., Fig. 4, to give an importance weight to sample $\tilde{x}_i$. We use the Beta distribution density up-shifted by 1

$$\begin{aligned}\tilde{p}_{\text{Beta}}(z; a, b) &= 1 + p_{\text{Beta}}(z; a, b)\\ &= 1 + \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} \cdot (z)^{a-1} \cdot (1-z)^{b-1}, \quad (6)\end{aligned}$$

where $\Gamma(a)$ is the Gamma function. We use the add-1 Beta function $\tilde{p}_{\text{Beta}}$ for stable optimization and better performance based on our pilot study. To enforce the model to focus on samples with promising TCPR's, we use the Beta distribution with $a = 1.1$ and $b \in \{3.0, 4.0, 5.0\}$. When $a = 1.1$ and $b = 5.0$, we have the Beta weighting function shown in Fig. 4. The objective function of Beta-weighting AT is

$$\ell_{\text{Beta}}(f_\theta(x_i), y_i) = -\tilde{p}_{\text{Beta}}(\hat{r}_i; a, b) \cdot \sum_{j}^{K} y_{ij} \log(f_\theta(x_i)_j) \quad (7)$$

We name this Beta distribution based importance weighting scheme in AT as AT-Beta.

In summary, the proposed AT-UR consists of two methods, AT-Beta and AT-EM. It also contains the combination of the two methods, i.e.,

$$\begin{aligned}\ell_{\text{Beta-EM}}(f_\theta(x_i), y_i) = -\tilde{p}_{\text{Beta}}(\hat{r}_i; a, b) \cdot \sum_{j}^{K} y_{ij} \log(f_\theta(x_i)_j)\\ + \lambda_{\text{EM}} H(f_\theta(x_i)), \quad (8)\end{aligned}$$

denoted as AT-Beta-EM. We test the three variants of AT-UR in our experiment and observe that different image classification tasks need different versions of AT-UR.

5.2. Theoretical Analysis on Beta Weighting

The previous subsection introduces the intuition behind the proposed AT-UR. This section gives the theoretical analysis on the Beta weighting, which shows a theoretical connection between Beta weighting and the PSS. We drop the subscript θ for f_θ to lighten the notation. Note that we leave the full proof in Appendix D.

Importance Weighting (IW) Algorithm. IW assigns importance weight $\omega(x, y)$ to each sample $(x, y) \in \mathcal{D}_{\text{tr}}$ such that $\omega(x, y)$ is directly determined by TCPR $\hat{r}$. Analogous to the empirical risk $\widehat{R}(f)$, we define the *IW empirical risk* with weights $\omega(x, y)$ for f as follows

$$\widehat{R}_\omega(f) = \frac{1}{m} \sum_{i=1}^{m} \omega(x_i, y_i) \cdot \ell(f(x_i), y_i). \quad (9)$$

It is worth noting that restricting $\omega(x_i, y_i) = 1$ as a special case for all data samples reduces $\widehat{R}_\omega(f)$ to $\widehat{R}(f)$.

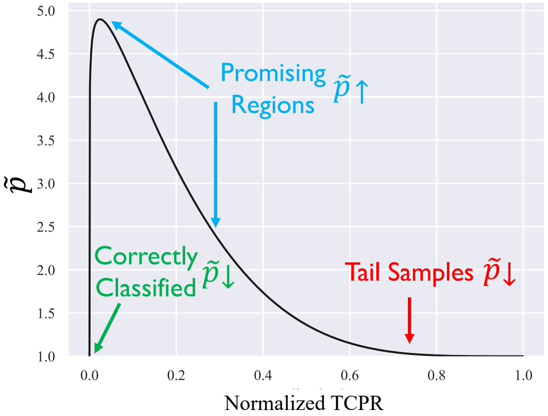

Figure 4. The Beta distribution density function $\tilde{p}_{\text{Beta}}$ used in our experiment. This weighting scheme increases the importance of samples in the promising region.

5.3. Beta Weighting for CP-Efficiency

We strategically design a group-wise IW approach that groups data into K disjoint subsets according their TCPR's, and assign the same weight to a group of data. For a sample (x, y), the importance weight is $\omega(x, y) = \tilde{p}_{\text{Beta}}(\hat{r}(x, y); a, b)$. The following theorem proves that the expectation of ℓ_{Beta} is an upper bound for the expectation of PSS, which indicates that optimizing ℓ_{Beta} is theoretically beneficial to reducing PSS and CP-efficiency.

Theorem 5.1. *(Connection between rank minimization and Beta-weighted minimization)* $L_{Beta}(f) := \sum_{k=1}^{K} \sigma_k \cdot \mathbb{E}[\ell(f(X), Y) | r_f(X, Y) = k]$, *where* $\sigma_k \sim p_{Beta}(k/(K+1); a, b)$.

$$\mathbb{E}_X[|\mathcal{C}_f(X)|] \le L_{Beta}(f),$$

where $|\mathcal{C}_f(X)|$ *is the cardinality of the prediction set* $\mathcal{C}_f(X)$ *for a classifier* f *with input* X *and* $r_f(X, Y)$ *is TCPR of* (X, Y) *in the classifier* f.

Remark. This theorem collaborates our intuition in the previous subsection that optimizing samples with moderate PSS with high importance may lead to the improvement of CP-efficiency. As far as we know, the main theorem is one of the first to build a connection between importance weighting and PSS in conformal prediction. The next section presents our empirical result on various datasets, which further confirm the effectiveness of the propsoed AT-UR. See Appendix D for the full proof.

6. Experiment

We first give the details of our experimental setting and then present the main empirical result.

6.1. Experimental Setting

Model. We use the adversarially pre-trained ResNet50 (He et al., 2016; Salman et al., 2020) with l_∞ norm and an attack budget $\epsilon_{pt} = 4/255$ in all experiment of our paper.

Dataset	CIFAR10		CIFAR100		Caltech256		CUB200	
Metric	Cvg	PSS	Cvg	PSS	Cvg	PSS	Cvg	PSS
AT	93.25(0.45)	2.54(0.04)	91.99 (0.61)	14.29(0.59)	94.35(0.81)	23.73(1.68)	91.87(0.90)	17.75(0.71)
AT-EM*	92.36(0.53)	**2.45(0.04)**	91.87(0.61)	13.29(0.49)	93.41(0.58)	21.19(1.46)	91.26(0.57)	**16.47(0.61)**
AT-Beta*	91.96(0.39)	2.50(0.04)	91.24(0.69)	**11.61(0.40)**	93.52(0.73)	**18.54(1.32)**	91.37(0.75)	16.56(0.76)
AT-Beta-EM*	92.06(0.44)	2.50(0.04)	91.13(0.63)	11.78(0.47)	93.50(0.69)	18.56(1.32)	91.93(0.68)	16.67(0.58)
FAT	93.01(0.53)	2.55(0.05)	92.04(0.60)	13.60(0.37)	93.88(0.45)	22.85(0.97)	91.37(0.80)	17.21(0.81)
FAT-EM*	92.71(0.66)	2.49(0.05)	91.63(0.91)	14.43(2.12)	93.52(0.67)	21.81(1.28)	91.58(1.25)	16.60(1.05)
FAT-Beta*	92.08(0.54)	2.55(0.04)	90.82(0.49)	11.10(0.25)	93.49(0.82)	**18.07(1.30)**	91.10(0.61)	**16.14(0.54)**
FAT-Beta-EM*	92.28(0.30)	**2.43(0.02)**	91.18(0.55)	**11.09(0.25)**	93.55(0.56)	18.16(1.04)	91.40(0.62)	16.39(0.49)
TRADES	93.01(0.46)	2.49(0.03)	91.75(0.72)	12.22(0.50)	94.33(0.49)	22.82(1.37)	92.03(0.59)	22.29(0.96)
TRADES-EM*	92.22(0.19)	**2.33(0.01)**	91.43(0.77)	12.39(0.54)	93.49(0.65)	**15.03(1.32)**	91.52(0.82)	**17.80(2.37)**
TRADES-Beta*	92.43(0.47)	2.44(0.04)	91.20(0.72)	**10.76(0.44)**	93.55(0.48)	17.02(1.04)	90.91(0.89)	18.19(0.99)
TRADES-Beta-EM*	92.12(0.36)	2.42(0.02)	91.22(0.62)	11.09(0.38)	93.13(0.33)	17.27(0.72)	90.96(0.93)	18.28(1.22)

Table 2. Comparison of AT baselines and the proposed AT-UR variants denoted with *, under the AutoAttack (Croce & Hein, 2020). The average coverage (Cvg) and Prediction Set Size (PSS) are presented, along with the standard deviation in parentheses. The most CP-efficient method is highlighted in bold. The result of using PGD100 attacks is in Tab. 3.

The reason is that, besides testing on CIFAR10/100, we also test on more challenging datasets such as Caltech256 and CUB200, on which an adversarially pre-trained model is shown to be much more robust than random initialized weights (Liu et al., 2023).

Dataset. Four datasets are used to evaluate our method, i.e., CIFAR10, CIFAR100 (Krizhevsky et al., 2009), Caltech-256 (Griffin et al., 2007) and Caltech-UCSD Birds-200-2011 (CUB200) (Wah et al., 2011). CIFAR10 and CIFAR100 contain low-resolution images of 10 and 100 classes, where the training and validation sets have 50,000 and 10,000 images respectively. Caltech-256 has 30,607 high-resolution images and 257 classes, which is split into training and validation set using a 9:1 ratio. CUB200 also contains high-resolution bird images for fine-grained image classification, with 200 classes, 5,994 training images and 5,794 validation images.

Training and Adversarial Attack. In all adversarial training of this paper, we generate adversarial perturbations using PGD attack. The PGD attack has 10 steps, with stepsize $\lambda = 2/255$ and attack budget $\epsilon = 8/255$. The batch size is set as 128 and the training epoch is 60. We divide the learning rate by 0.1 at the 30th and 50th epoch. We use the strong AutoAttack (Croce & Hein, 2020) with $\epsilon = 8/255$ in Tab. 2. We use PGD attack with 100 steps for all other results in this paper. The stepsize and attack budget in PGD100 is the same as in adversarial training, i.e., $\lambda = 2/255$ and $\epsilon = 8/255$. See more training details in Appendix B.

Conformal Prediction Setting. We fix the training set in our experiment and randomly split the original test set into calibration and test set with a ratio of 1:4 for conformal prediction. For each AT method, we repeat the training for three trials with three different seeds and repeat the calibration-test splits five times, which produces 15 trials for our evaluation. The mean and standard deviation of coverage and PSS of 15 trials are reported. If not specified, we use APS (Romano et al., 2020) as the CP method in our paper as the performance of APS is more stable than RAPS, as shown in Fig. 2. The target coverage is set as 90% following existing literature in CP (Romano et al., 2020; Angelopoulos et al., 2020; Ghosh et al., 2023). We use the same adversarial attack setting as in (Gendler et al., 2021), i.e., both calibration and test samples are attacked with the same adversary. We discuss the limitation of this setting in the conclusion section.

Baselines. We use AT (Madry et al., 2018), Fair-AT (FAT) (Xu et al., 2021) and TRADES (Zhang et al., 2019) as the baseline and test the performance of the two proposed uncertainty-reducing methods with the three baselines. AT and TRADES are the most popular adversarial training methods and FAT reduces the robustness variance among classes, which could reduce the PSS, which is validated by our experiment. Note that we only reports the performance of CP, i.e., coverage and PSS, in the main paper as the main target of our paper is to improve CP efficiency.

6.2. Experimental Results

Efficacy of AT-UR in reducing PSS. The coverage and PSS of all tested methods under the AutoAttack are shown in Tab. 2. The proposed AT-UR methods effectively reduce the PSS when combined with the three AT baselines on four datasets, validating our intuition on the connection between the two factors, i.e., predictive entropy and TCPR, and PSS. More importantly, the result is also consistent with our finding in Theorem D.1. There are two phenomena worth noting. First, the Beta weighting generally works better than EM when using AT and FAT, with Beta+EM potentially improving the CP-efficiency in some cases. Second, when using TRADES, EM is more promising than Beta weighting (e.g., EM is better than other two on three out of four datasets). Thus, we recommend that for AT and FAT, using Beta or Beta-EM is the first choice if one needs to train an adversari-

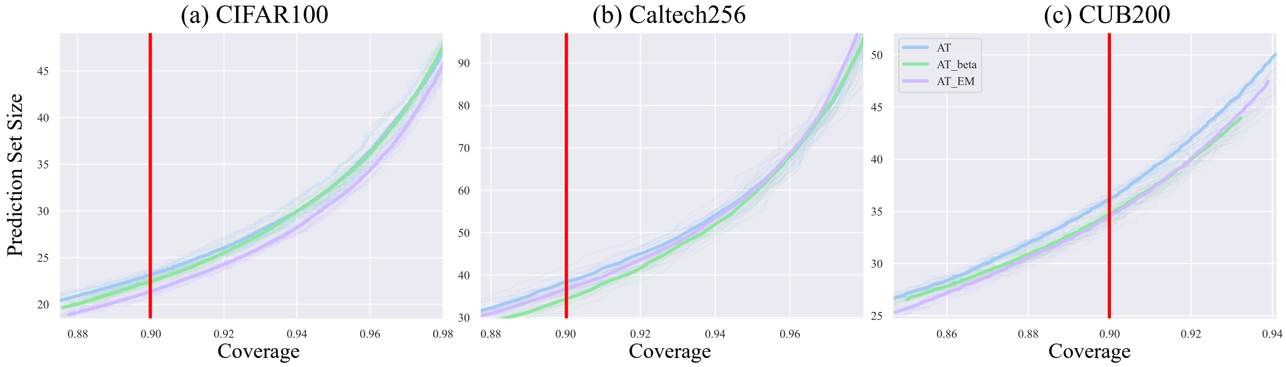

Figure 5. The CP curve of coverage versus PSS. Each point on the curve is obtained by adjusting the threshold $\hat{\tau}_{\text{cal}}$. We plot 15 CP curves (opaque line) and their average (solid line) for each method. The red vertical line indicates the operating point for 90% coverage.

ally robust and also CP-efficient model, while for TRADES, it is more reasonable to first try EM. Note that although the Top-1 accuracy of our method (Appendix C) is decreased compared to baselines, the main target of our method is to improve CP efficiency as we use the conformal prediction instead of the Top-1 prediction. Tab. 4 shows the normalized PSS result to mitigate the influence of different K's on the comparison.

Coverage-PSS curve visualization. To visualize the effect of AT-UR more comprehensively, we plot the CP curve by adjusting the threshold $\hat{\tau}_{\text{cal}}$ to get different points on the curve of coverage versus PSS. Fig. 5 shows the CP curve of AT, AT-Beta and AT-EM on three datasets. It demonstrates that AT-UR achieves a reduced PSS compared to the AT baseline, not only at 90% coverage , but also over a wide range of coverage values.

6.3. Detailed Empirical Analysis

(a) Sensitivity to hyperparameters and performance under different attack budgets. We use different Beta-weighting hyperparameters on Caltech256. The performance is stable within a range of b=(3.0, 4.0, 5.0) as shown in Tab. 6. In addition to the ϵ=8.0, we test different attack budgets ϵ=4.0, 12.0, 16.0 and report the result on Caltech256 in Tab. 7. The result shows that across the attack budgets, our method is consistently better than the AT baseline.

(b) Compare with Uncertainty-Aware training (Einbinder et al., 2022). We train three models with vanilla AT, Conformal AT in (Einbinder et al., 2022) and our AT-Beta on CIFAR100 respectively. The experiment follows the setting in the Conformal Training (See more details in Appendix B). AT, Conformal AT and AT-Beta have the averaged coverage and PSS of (89.82, 33.43), (90.36, 35.32) and (89.78, **30.18**) respectively, demonstrating the effectiveness of our Beta-weighting scheme over Conformal AT in the adversarial environment.

(c) Does Focal loss improve CP-efficiency? We consider using a power function $\hat{r}_i^\eta$ as in focal loss (Lin et al., 2017) to generate loss weights and test the CP performance of AT-Focal. We set $\eta = 0.5$ based on a hyperparameter search from $\{0.1, 0.5, 1.0, 2.0\}$. AT-Focal forces the model to focus on hard samples, contrary to our AT-Beta which focuses on promising samples. The averaged coverage and PSS of AT-Focal on CIFAR100 and Caltech256 are (90.50, 27.24) and (91.38, 48.35) respectively, which is far worse than the AT baseline of (90.45, 23.79) and (91.35, 43.20). This result corroborates that promising samples are crucial for improving CP-efficiency instead of hard samples.

(d) What is the difference between label smoothing and AT-EM? The formulation of AT-EM is similar to the formulation of label smoothing (Müller et al., 2019), if we combine the log term in (4). However, label smoothing and AT-EM train the model into two different directions: the former increases the prediction entropy (by smoothing the label probabilities to be more uniform), while the latter decreases the prediction entropy. We validate this argument on Caltech256 and find that label smoothing makes the CP-efficiency much worse than the AT baseline, with an averaged coverage and PSS of (90.22, 46.39), compared to (91.35, 43.20) of AT.

7. Conclusion

This paper first studies the pitfalls of CP under adversarial attacks and thus underscores the importance of AT when using CP in an adversarial environment. Then we unveil the compromised CP-efficiency of popular AT methods and propose to design uncertainty-reducing AT for CP-efficiency based on our empirical observation on two factors affecting the PSS. Our theoretical results establish the connection between PSS and Beta weighting. Our experiment validates the effectiveness of the proposed AT-UR on four datasets when combined with three AT baselines. A common limitation shared by this study and (Gendler et al., 2021) is the assumption that the adversarial attack is known, enabling the calibration set to be targeted by the same adversary as the test set. In future research, we will alleviate this constraint by exploring CP within an adversary-agnostic context.

Impact Statements

This paper investigating and improving CP-efficiency for deep learning models under adversarial attacks makes important contribution to the reliability and safety of artificial intelligence (AI) systems. By exposing the risk caused by adversarial attacks in the conformal prediction setting, the research not only bolsters the robustness of deep learning models but also provides a more nuanced understanding of conformal prediction. The theoretical and empirical results in this paper hold immense societal implications, particularly in high-stakes applications such as self-driving cars and medical diagnosis, advancing the positive impact of AI on society by promoting secure and reliable AI-driven advancements.

References

Angelopoulos, A. N., Bates, S., Jordan, M., and Malik, J. Uncertainty sets for image classifiers using conformal prediction. In *International Conference on Learning Representations*, 2020.

Croce, F. and Hein, M. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *International conference on machine learning*, pp. 2206–2216. PMLR, 2020.

Cui, J., Liu, S., Wang, L., and Jia, J. Learnable boundary guided adversarial training. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 15721–15730, 2021a.

Cui, Y., Yao, W., Li, Q., Chan, A. B., and Xue, C. J. Accelerating monte carlo bayesian prediction via approximating predictive uncertainty over the simplex. *IEEE transactions on neural networks and learning systems*, 33(4): 1492–1506, 2020.

Cui, Y., Liu, Z., Li, Q., Chan, A. B., and Xue, C. J. Bayesian nested neural networks for uncertainty calibration and adaptive compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2392–2401, 2021b.

CUI, Y., Liu, Z., Liu, X., Liu, X., Wang, C., Kuo, T.-W., Xue, C. J., and Chan, A. B. Bayes-MIL: A new probabilistic perspective on attention-based multiple instance learning for whole slide images. In *The Eleventh International Conference on Learning Representations*, 2023.

Cui, Y., Mao, Y., Liu, Z., Li, Q., Chan, A. B., Liu, X., Kuo, T.-W., and Xue, C. J. Variational nested dropout. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

Einbinder, B.-S., Romano, Y., Sesia, M., and Zhou, Y. Training uncertainty-aware classifiers with conformalized deep learning. *Advances in Neural Information Processing Systems*, 35:22380–22395, 2022.

Gal, Y. and Ghahramani, Z. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pp. 1050–1059. PMLR, 2016.

Gendler, A., Weng, T.-W., Daniel, L., and Romano, Y. Adversarially robust conformal prediction. In *International Conference on Learning Representations*, 2021.

Ghosh, S., Shi, Y., Belkhouja, T., Yan, Y., Doppa, J., and Jones, B. Probabilistically robust conformal prediction. In *Uncertainty in Artificial Intelligence*, pp. 681–690. PMLR, 2023.

Gibbs, I. and Candes, E. Adaptive conformal inference under distribution shift. *Advances in Neural Information Processing Systems*, 34:1660–1672, 2021.

Grandvalet, Y. and Bengio, Y. Semi-supervised learning by entropy minimization. *Advances in neural information processing systems*, 17, 2004.

Griffin, G., Holub, A., and Perona, P. Caltech-256 object category dataset. 2007.

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.

Harris, C. R., Millman, K. J., Van Der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., et al. Array programming with numpy. *Nature*, 585(7825):357–362, 2020.

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.

Kendall, A. and Gal, Y. What uncertainties do we need in bayesian deep learning for computer vision? *Advances in neural information processing systems*, 30, 2017.

Kingma, D. P., Salimans, T., and Welling, M. Variational dropout and the local reparameterization trick. *Advances in neural information processing systems*, 28, 2015.

Kireev, K., Andriushchenko, M., and Flammarion, N. On the effectiveness of adversarial training against common corruptions. In *Uncertainty in Artificial Intelligence*, pp. 1012–1021. PMLR, 2022.

Krizhevsky, A., Hinton, G., et al. Learning multiple layers of features from tiny images. 2009.

Lei, J., G'Sell, M., Rinaldo, A., Tibshirani, R. J., and Wasserman, L. Distribution-free predictive inference for regression. *Journal of the American Statistical Association*, 113(523):1094–1111, 2018.

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.

Liu, F., Han, B., Liu, T., Gong, C., Niu, G., Zhou, M., Sugiyama, M., et al. Probabilistic margins for instance reweighting in adversarial training. *Advances in Neural Information Processing Systems*, 34:23258–23269, 2021a.

Liu, Z. and Chan, A. B. Boosting adversarial robustness from the perspective of effective margin regularization. In *British Machine Vision Conference (BMVC)*, 2022.

Liu, Z., Yufei, C., and Chan, A. B. Improve generalization and robustness of neural networks via weight scale shifting invariant regularizations. In *ICML 2021 Workshop on Adversarial Machine Learning*, 2021b.

Liu, Z., Xu, Y., Ji, X., and Chan, A. B. Twins: A fine-tuning framework for improved transferability of adversarial robustness and generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16436–16446, 2023.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018.

Müller, R., Kornblith, S., and Hinton, G. E. When does label smoothing help? *Advances in neural information processing systems*, 32, 2019.

Papadopoulos, H., Proedrou, K., Vovk, V., and Gammerman, A. Inductive confidence machines for regression. In *Machine Learning: ECML 2002: 13th European Conference on Machine Learning Helsinki, Finland, August 19–23, 2002 Proceedings 13*, pp. 345–356. Springer, 2002.

Parzen, E. On estimation of a probability density function and mode. *The annals of mathematical statistics*, 33(3): 1065–1076, 1962.

Platt, J. et al. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3):61–74, 1999.

Qin, C., Martens, J., Gowal, S., Krishnan, D., Dvijotham, K., Fawzi, A., De, S., Stanforth, R., and Kohli, P. Adversarial robustness through local linearization. *Advances in Neural Information Processing Systems*, 32, 2019.

Qin, Y., Wang, X., Beutel, A., and Chi, E. Improving calibration through the relationship with adversarial robustness. *Advances in Neural Information Processing Systems*, 34: 14358–14369, 2021.

Razzak, M. I., Naz, S., and Zaib, A. Deep learning for medical image processing: Overview, challenges and the future. *Classification in BioApps: Automation of Decision Making*, pp. 323–350, 2018.

Romano, Y., Sesia, M., and Candes, E. Classification with valid and adaptive coverage. *Advances in Neural Information Processing Systems*, 33:3581–3591, 2020.

Rosenblatt, M. Remarks on some nonparametric estimates of a density function. *The annals of mathematical statistics*, pp. 832–837, 1956.

Salman, H., Ilyas, A., Engstrom, L., Kapoor, A., and Madry, A. Do adversarially robust imagenet models transfer better? *Advances in Neural Information Processing Systems*, 33:3533–3545, 2020.

Shafer, G. and Vovk, V. A tutorial on conformal prediction. *Journal of Machine Learning Research*, 9(3), 2008.

Stutz, D., Hein, M., and Schiele, B. Confidence-calibrated adversarial training: Generalizing to unseen attacks. In *International Conference on Machine Learning*, pp. 9155–9166. PMLR, 2020.

Vovk, V., Gammerman, A., and Saunders, C. Machine-learning applications of algorithmic randomness. 1999.

Vovk, V., Gammerman, A., and Shafer, G. *Algorithmic learning in a random world*, volume 29. Springer, 2005.

Wah, C., Branson, S., Welinder, P., Perona, P., and Belongie, S. The caltech-ucsd birds-200-2011 dataset. 2011.

Wang, Y., Zou, D., Yi, J., Bailey, J., Ma, X., and Gu, Q. Improving adversarial robustness requires revisiting misclassified examples. In *International conference on learning representations*, 2019.

Xu, H., Liu, X., Li, Y., Jain, A., and Tang, J. To be robust or to be fair: Towards fairness in adversarial training. In *International conference on machine learning*, pp. 11492–11501. PMLR, 2021.

Zhang, H., Yu, Y., Jiao, J., Xing, E., El Ghaoui, L., and Jordan, M. Theoretically principled trade-off between robustness and accuracy. In *International conference on machine learning*, pp. 7472–7482. PMLR, 2019.

A. Adaptive Prediction Sets (Romano et al., 2020)

We introduce one example of prediction set function, i.e., APS conformal prediction used in our experiment. Assume we have the prediction distribution $\pi(x) = f_\theta(x)$ and order this probability vector with the descending order $\pi_{(1)}(x) \geq \pi_{(2)}(x) \geq \ldots \geq \pi_{(K)}(x)$. We first define the following generalized conditional quantile function,

$$Q(x; \pi, \tau) = \min\{k \in \{1, \ldots, K\} \,:\, \pi_{(1)}(x) + \pi_{(2)}(x) + \ldots + \pi_{(k)}(x) \geq \tau\}, \tag{10}$$

which returns the class index with the generalized quantile $\tau \in [0, 1]$. The function $\mathcal{S}$ can be defined as

$$\mathcal{S}(x, u; \pi, \tau) = \begin{cases} \text{`}y\text{' indices of the } Q(x; \pi, \tau) - 1 \text{ largest } \pi_y(x), & \text{if } u \leq U(x; \pi, \tau), \\ \text{`}y\text{' indices of the } Q(x; \pi, \tau) \text{ largest } \pi_y(x), & \text{otherwise,} \end{cases} \tag{11}$$

where

$$U(x; \pi, \tau) = \frac{1}{\pi_{(Q(x;\pi,\tau))}(x)} \left[\sum_{k=1}^{Q(x;\pi,\tau)} \pi_{(k)}(x) - \tau \right].$$

It has input x, $u \in [0, 1]$, π, and τ and can be seen as a generalized inverse of Equation 10.

On the calibration set, we compute a generalized inverse quantile conformity score with the following function,

$$E(x, y, u; \pi) = \min\left\{\tau \in [0, 1] : y \in \mathcal{S}(x, u; \pi, \tau)\right\}, \tag{12}$$

which is the smallest quantile to ensure that the ground-truth class is contained in the prediction set $\mathcal{S}(x, u; \pi, \tau)$. With the conformity scores on calibration set $\{E_i\}_{i=1}^{n_c}$, we compute the $\lceil (1 - \alpha)(1 + n_c) \rceil$th largest value in the score set as $\hat{\tau}_{\text{cal}}$. During inference, the prediction set is generated with $\mathcal{S}(x^*, u; \pi^*, \hat{\tau}_{\text{cal}})$ for a novel test sample x^*.

B. More Experimental Details

APS Setting. We use the default setting of APS specified in the official code of (Angelopoulos et al., 2020), i.e., first use temperature scaling (Platt et al., 1999; Guo et al., 2017) to calibrate the prediction distribution then compute the generalized inverse quantile conformity score to perform the calibration and conformal prediction.

Hyperparameter and Baseline Setting. As mentioned in the main paper, we use $a = 1.1$ and search b from the discrete set $\{2.0, 3.0, 4.0, 5.0\}$ in Beta distribution since the parameter combinations perform well in our pilot study and satisfy the goal of focusing on promising samples. The learning rate and weight decay of AT, FAT and TRADES are determined by grid search from {1e-4,3e-4,1e-3,3e-3,1e-2} and {1e-3,1e-4,1e-5} respectively. We compute the class weight for FAT using the output of a softmax function with error rate of each class as input. The temperature in the softmax function is set as 1.0. For TRADES, we follow the default setting $\beta = 6.0$ for the KL divergence term (Zhang et al., 2019). Our AT-UR method also determines the learning rate and weight decay using the grid search with the same mentioned grid. For TRADES, we weight both the cross-entropy loss and KL divergence loss with the Beta density function based on TCPR.

CP Curve. The CP curve in Fig. 5 is obtained by using different threshold values, for instance, using the linspace function in numpy (Harris et al., 2020) with `np.linspace(0.9,1.1,200)` $\times\hat{\tau}_{\text{cal}}$ generates 200 different (coverage, PSS) points.

Compare with Conformal AT (Einbinder et al., 2022). We use the experimental setting in the original paper, where they train a randomly initialized ResNet50 using SGDM with batch size=128, learning rate=0.1, weight decay=0.0005, for 120 epochs, where the learning rate is divided by 10 at 100th epoch. 45000 original training samples are used for training and the remaining 5000 samples are used as a held-out set for computing the conformal loss. We use the same attack parameters in this setting as in other experiments during both training and inference. In the conformal inference based on APS scores, we split the original test set with a ratio of 1:1 into a calibration and a test set and only test the final-epoch model. We run three trials for each approach and report the average coverage and PSS in the main paper. The experiment result that the effectiveness of our AT-Beta is generalizable to the randomly intialized model training.

Dataset	CIFAR10		CIFAR100		Caltech256		CUB200	
Metric	Cvg	PSS	Cvg	PSS	Cvg	PSS	Cvg	PSS
AT	90.55(0.51)	3.10(0.07)	90.45(0.59)	23.79(0.80)	91.35(0.85)	43.20(2.11)	90.33(0.89)	37.37(2.11)
AT-EM*	90.39(0.48)	**3.05(0.05)**	90.35(0.82)	**22.05(1.02)**	91.09(0.79)	41.42(2.52)	90.08(1.10)	34.77(2.60)
AT-Beta*	90.46(0.51)	3.11(0.07)	90.10(0.51)	22.64(0.65)	90.20(0.84)	**35.39(2.66)**	90.17(1.06)	35.25(2.15)
AT-Beta-EM*	90.65(0.62)	3.10(0.08)	90.40(0.60)	22.53(0.91)	90.81(1.00)	36.17(3.73)	90.31(0.84)	**33.10(1.74)**
FAT	90.69(0.61)	3.16(0.07)	90.41(0.67)	23.54(0.81)	90.70(0.77)	41.52(2.43)	90.50(1.17)	39.43(2.88)
FAT-EM*	90.54(0.68)	3.06(0.06)	90.00(0.82)	23.47(2.71)	90.55(0.79)	39.72(2.49)	89.89(0.92)	35.51(2.06)
FAT-Beta*	90.47(0.51)	3.16(0.08)	90.22(0.47)	23.15(0.71)	89.90(0.70)	34.72(2.25)	89.92(0.84)	35.46(1.71)
FAT-Beta-EM*	90.71(0.61)	**3.04(0.07)**	90.36(0.50)	**22.28(0.63)**	90.41(0.61)	**33.59(2.75)**	89.88(0.91)	**34.35(1.68)**
TRADES	90.72(0.62)	3.31(0.09)	90.35(0.57)	27.60(0.97)	90.82(0.81)	44.80(3.42)	90.38(0.76)	52.18(2.60)
TRADES-EM*	90.54(0.40)	**3.16(0.05)**	90.36(0.71)	**26.76(1.00)**	90.68(0.87)	**38.83(3.78)**	90.05(0.76)	**44.96(2.75)**
TRADES-Beta*	90.41(0.56)	3.30(0.09)	90.14(0.85)	27.22(1.42)	90.48(0.70)	38.94(2.74)	89.83(0.84)	49.63(2.59)
TRADES-Beta-EM*	90.01(0.40)	3.21(0.06)	90.54(0.24)	26.55(0.35)	90.52(0.74)	39.83(3.02)	90.16(1.12)	48.54(2.58)

Table 3. Comparison of AT baselines and the proposed AT-UR variants denoted with *, under the PGD100 attack.

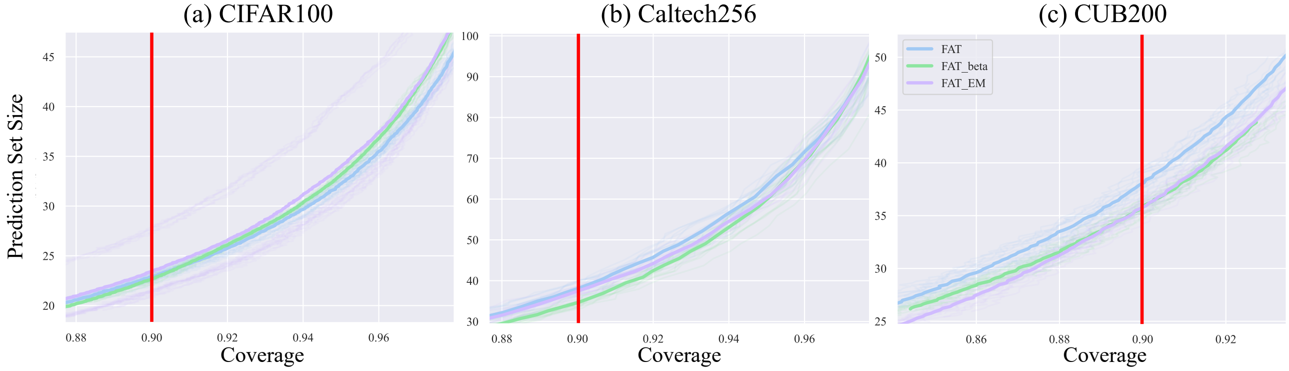
(a) CIFAR100 (b) Caltech256 (c) CUB200

Figure 6. The CP curve of coverage versus prediction set size using FAT and PGD100 attack.

C. More Experimental Results

Note that this paper uses CP as the inference method to achieve a coverage guarantee, which is orthogonal to the Top-1 inference method. Thus, Top-1 accuracy is not a relevant metric in the context of CP inference. Nevertheless, we show the Top-1 accuracy of tested methods in Tab. 5. Using AT-UR generally worsens the Top-1 accuracy, especially for TRADES. However, note that using TRADES-Beta-EM can improve the Top-1 robust accuracy of TRADES-Beta on CIFAR10 and TRADES-EM on Caltech256. This result again confirms the observation that Top-1 accuracy is not necessarily correlated with CP-efficiency. When we compare the result of using PGD100 and AA, the robust accuracy under AA drops while the prediction set size reduced (CP efficiency is improved), indicating that a stronger attack can lead to reduced PSS. To reduce the effect of number of classes (K) on the PSS, Tab. 4 shows the normalized PSS using K when using the PGD100 attack.

Fig. 6 and Fig. 7 shows the CP curve of FAT and TRADES when they are combined with EM and Beta on three datasets. It demonstrates that the CP-efficiency is also improved when using FAT and TRADES as in the experiment using AT. In most cases (5 out of 6), AT-UR (either EM or Beta) has a lower PSS than the corresponding baseline within a large range of coverage.

Fig. 8 shows the percentage of samples TCPR=1, 1<TCPR<20 and TCPR≥20 during AT on CIFAR100, demonstrating that the promising samples are the majority in most time training, especially for the early 30 epochs.

D. Proof of Theorem

Theorem D.1. *(Equivalence between rank minimization and Beta-weighted minimization)* $L_{Beta}(f) := \sum_{k=1}^{K} \sigma_k \cdot \mathbb{E}[\ell(f(X), Y) | r_f(X, Y) = k]$*, where* $\sigma_k \sim p_{Beta}(k/(K+1); a, b)$*.*

$$\mathbb{E}_X [|\mathcal{C}_f(X)|] \leq L_{Beta}(f).$$

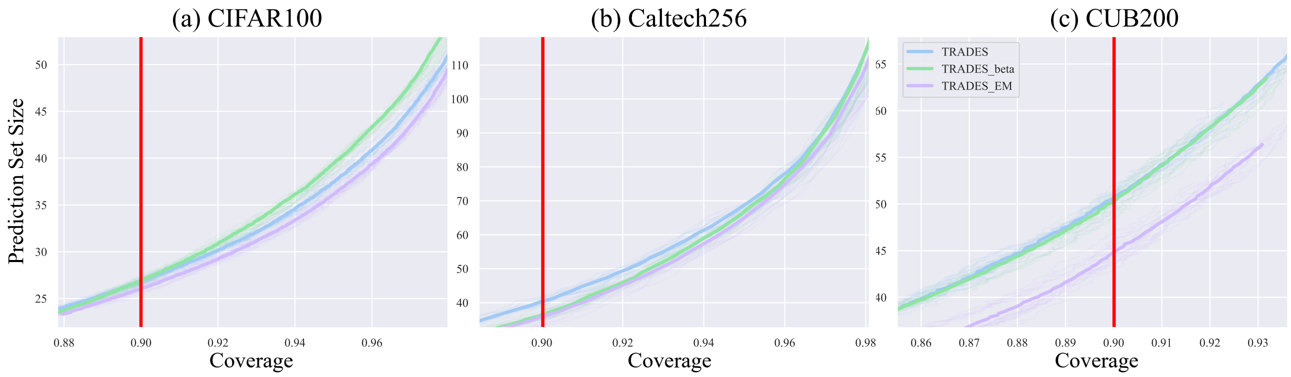

Figure 7. The CP curve of coverage versus prediction set size using TRADES and PGD100 attack.

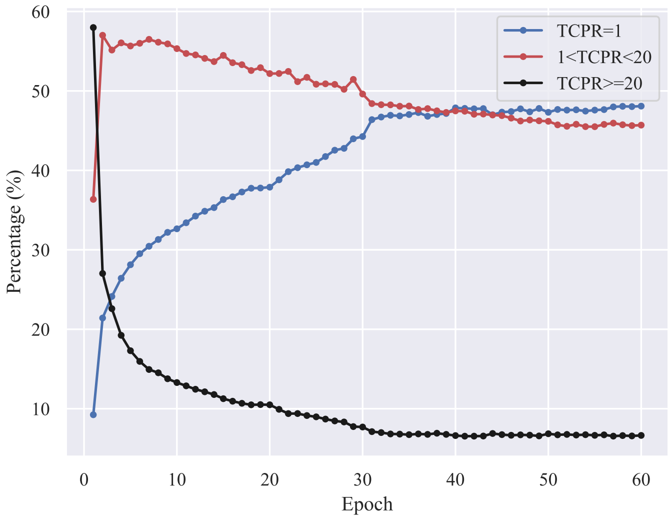

Figure 8. The percentage of samples with different TCPR's in CIFAR100 training.

Proof. (of Theorem D.1)

Before the proof, we first present two key lemmas (Lemma D.2 and Lemma D.4) below. Note that our theoretical analysis only uses the original Beta function p_{Beta} instead of the up-shifted version, which does not affect the conclusion since the orignal Beta-weighting can be regarded as a regularization term. As we noted in the main paper, we use the ranking starting from 0 in our implementation while the theoretical analysis assumes the ranking starts from 1. Nevertheless, we can shift the index from 1 to 0 to get the same theoretical result.

Lemma D.2. *(CP PSS upper bounded by partial average rank) Let* $K^* = \max\{k \in [K] : \mathbb{P}_{XY}[\sum_{l=1}^{k} f(X)_{(l)} \leq \tau_{1-\alpha} | r_f(X,Y) \geq k] \geq \alpha\}$.

$$\mathbb{E}_X[|\mathcal{C}_f(X)|] \leq \sum_{k=1}^{K^*} k \cdot \mathbb{P}[r_f(X,Y) = k] \tag{13}$$

Proof. (of Lemma D.2) We first introduce the notations. $f(X)_{(l)}$ is the lth sorted predictive probability with the descending order, $V(X,y)$ is the cumulative summation of $f(X)_{(l)}$, i.e., $V(X,y) = \sum_{l=1}^{y} f(X)_{(l)}$. $r_f(X,Y)$ is the TCPR of input

Dataset	CIFAR10		CIFAR100		Caltech256		CUB200	
Metric	Cvg	NPSS	Cvg	NPSS	Cvg	NPSS	Cvg	NPSS
AT	90.55(0.51)	31.0(0.7)	90.45(0.59)	23.79(0.80)	91.35(0.85)	16.8(0.8)	90.33(0.89)	18.7(1.1)
AT-EM*	90.39(0.48)	**30.5(0.5)**	90.35(0.82)	**22.05(1.02)**	91.09(0.79)	16.1(1.0)	90.08(1.10)	17.4(1.3)
AT-Beta*	90.46(0.51)	31.1(0.7)	90.10(0.51)	22.64(0.65)	90.20(0.84)	**13.8(1.0)**	90.17 (1.06)	17.6(1.1)
AT-Beta-EM*	90.65(0.62)	31.0(0.8)	90.40(0.60)	22.53(0.91)	90.81(1.00)	14.1(1.4)	90.31(0.84)	**16.5(0.9)**
FAT	90.69(0.61)	31.6(0.7)	90.41(0.67)	23.54(0.81)	90.70(0.77)	16.2(0.9)	90.50(1.17)	19.7(1.4)
FAT-EM*	90.54(0.68)	30.6(0.6)	90.00(0.82)	23.47(2.71)	90.55(0.79)	15.5(1.0)	89.89(0.919)	17.8(1.0)
FAT-Beta*	90.47(0.51)	31.6(0.8)	90.22(0.47)	23.15(0.71)	89.90(0.70)	13.5(0.9)	89.92(0.84)	17.7(0.9)
FAT-Beta-EM*	90.71(0.61)	**30.4(0.7)**	90.36(0.50)	**22.28(0.63)**	90.41(0.61)	**13.1(1.1)**	89.88(0.91)	**17.2(0.8)**
TRADES	90.72(0.62)	33.1(0.9)	90.35(0.57)	27.60(0.97)	90.82(0.81)	17.4(1.3)	90.38(0.76)	26.1(1.3)
TRADES-EM*	90.54(0.40)	**31.6(0.5)**	90.36(0.71)	**26.76(1.00)**	90.68(0.87)	**15.1(1.5)**	90.05(0.76)	**22.5(1.4)**
TRADES-Beta*	90.41(0.56)	33.0(0.9)	90.14(0.85)	27.22(1.42)	90.48(0.70)	15.1(1.1)	89.83(0.84)	24.8(1.3)
TRADES-Beta-EM*	90.01(0.40)	32.1(0.6)	90.54(0.24)	26.55(0.35)	90.52(0.74)	15.5(1.2)	90.16(1.12)	24.3(1.3)

Table 4. Comparison of AT baselines and the proposed AT-UR variants denoted with *, under the PGD100 attack. The average coverage (Cvg) and prediction set size normalized by the class number (NPSS, %) are presented.

Dataset	CIFAR10		CIFAR100		Caltech256		CUB200	
Metric	Std. Acc.	Rob. Acc.	Std. Acc.	Rob. Acc.	Std. Acc.	Rob. Acc.	Std. Acc.	Rob. Acc.
AT	89.76(0.15)	50.17(0.91)	68.92(0.38)	28.49(1.14)	75.28(0.51)	47.53(0.67)	65.36(0.27)	26.29(0.45)
AT-EM*	90.02(0.10)	48.92(0.39)	68.39(0.51)	28.33(0.73)	74.62(0.22)	46.23(0.44)	64.75(0.33)	25.60(0.25)
AT-Beta*	89.81(0.22)	47.50(0.78)	68.50(0.28)	28.04(0.57)	74.66(0.54)	45.40(0.59)	64.62(0.17)	25.57(0.41)
AT-Beta-EM*	90.00(0.06)	46.69(0.71)	68.45(0.35)	27.20(1.08)	74.71(0.36)	44.88(0.60)	64.44(0.22)	25.32(0.38)
FAT	89.96(0.25)	49.12(0.70)	68.80(0.38)	28.97(0.53)	75.20(0.33)	47.09(0.70)	65.01(0.19)	25.21(0.57)
FAT-EM*	90.19(0.07)	48.31(0.80)	68.76(0.39)	25.84(3.54)	74.59(0.20)	45.53(0.70)	65.13(0.25)	24.92(0.27)
FAT-Beta*	90.07(0.16)	47.09(0.50)	68.58(0.33)	27.90(0.56)	74.00(0.39)	45.12(0.90)	64.38(0.25)	24.76(0.26)
FAT-Beta-EM*	89.86(0.06)	48.61(0.13)	67.95(0.24)	28.06(0.24)	74.78(0.10)	45.75(0.48)	64.32(0.21)	23.57(0.14)
TRADES	87.31(0.27)	53.07(0.23)	62.83(0.33)	32.07(0.20)	69.57(0.25)	47.07(0.37)	58.16(0.38)	27.82(0.23)
TRADES-EM*	86.68(0.06)	52.71(0.26)	57.03(0.31)	30.29(0.25)	57.17(0.39)	39.56(0.59)	45.50(5.81)	22.26(2.26)
TRADES-Beta*	89.81(0.22)	47.50(0.78)	62.61(0.36)	30.20(0.31)	70.96(0.25)	46.74(0.23)	57.72(0.24)	23.49(0.21)
TRADES-Beta-EM*	86.99(0.10)	51.85(0.23)	62.13(0.34)	30.52(0.20)	69.44(0.25)	46.24(0.39)	56.03(0.15)	22.90(0.32)

Table 5. Top-1 clean and robust accuracy comparison of AT baselines and the proposed AT-UR variants under the PGD100 attack.

(X, Y) when using the classifier f. $\tau_{1-\alpha}$ is the $1-\alpha$ quantile of the conformity score at the population level and α is the confidence level for conformal prediction.

$$
\begin{aligned}
&\mathbb{E}_X[|\mathcal{C}_f(X)|] = \mathbb{E}_X[\sum_{y=1}^{K} \mathbb{1}[V(X,y) \leq \tau_{1-\alpha}]] \\
&= \sum_{y=1}^{K} \mathbb{E}_X[\mathbb{1}[V(X,y) \leq \tau_{1-\alpha}] \cdot \mathbb{E}_Y[\mathbb{1}[r_f(X,Y) < r_f(X,y)] + \mathbb{1}[r_f(X,Y) \geq r_f(X,y)]]] \\
&= \sum_{y=1}^{K} \mathbb{E}_X[\mathbb{1}[V(X,y) \leq \tau_{1-\alpha}] \cdot \mathbb{E}_Y[\mathbb{1}[r_f(X,Y) < r_f(X,y)]]] \\
&\quad + \sum_{y=1}^{K} \mathbb{E}_X[\mathbb{1}[V(X,y) \leq \tau_{1-\alpha}] \cdot \mathbb{E}_Y[\mathbb{1}[r_f(X,Y) \geq r_f(X,y)]]] \\
&= \underbrace{\sum_{y=1}^{K} \mathbb{E}_{XY}[\mathbb{1}[V(X,y) \leq \tau_{1-\alpha}] \cdot \mathbb{1}[r_f(X,Y) < r_f(X,y)]]}_{=A} \\
&\quad + \underbrace{\sum_{y=1}^{K} \mathbb{E}_{XY}[\mathbb{1}[V(X,y) \leq \tau_{1-\alpha}] \cdot \mathbb{1}[r_f(X,Y) \geq r_f(X,y)]]}_{=B}
\end{aligned}
$$

Method	(a,b)=(1.1, 2.0)	(a,b)=(1.1, 3.0)	(a,b)=(1.1, 4.0)	(a,b)=(1.1, 5.0)	(a,b)=(1.1, 6.0)	AT, (a,b)=(1.0, 1.0)
Cvg	90.59(0.56)	90.18(0.85)	90.25(0.76)	90.20(0.84)	90.97(1.08)	91.35(0.85)
PSS	37.09(2.24)	35.22(2.76)	35.55(2.28)	35.39(2.66)	38.54(3.47)	43.20(2.11)

Table 6. Comparison of using different (a,b) in AT-Beta on Caltech256.

Attack Budget	ϵ=4/255		ϵ=8/255		ϵ=12/255		ϵ=16/255	
Metric	Cvg	PSS	Cvg	PSS	Cvg	PSS	Cvg	PSS
AT	92.73(0.81)	21.91(1.59)	91.35(0.85)	43.20(2.11)	89.68(0.89)	78.80(5.43)	90.22(1.10)	136.86(7.20)
AT-EM	92.93(0.67)	21.64(1.89)	91.09(0.79)	41.42(2.52)	89.48(0.75)	76.39(4.21)	89.97(0.67)	132.15(4.08)
AT-Beta	91.85(1.06)	**17.17(2.05)**	90.20(0.84)	35.39(2.66)	89.78(1.22)	79.79(7.09)	90.12(0.87)	142.17(6.89)
AT-EM-Beta	91.83(0.76)	17.59(1.21)	90.81(1.00)	36.17(3.73)	89.97(0.92)	74.38(4.51)	90.05(0.98)	131.64(5.59)

Table 7. Comparison of using different attack budgets in PGD100 with Caltech256.

$$
\begin{aligned}
A = &\sum_{y=1}^{K} \mathbb{P}_{XY}[V(X,y) \leq \tau_{1-\alpha}, r_f(X,Y) < r_f(X,y)] \\
= &\sum_{y=1}^{K} \mathbb{P}_{XY}[r_f(X,Y) < r_f(X,y)] \cdot \mathbb{P}_{XY}[V(X,y) \leq \tau_{1-\alpha} | r_f(X,Y) < r_f(X,y)] \\
\overset{(a)}{<} &\sum_{y=1}^{K} \mathbb{P}_{XY}[r_f(X,Y) < r_f(X,y)] \cdot \alpha \\
&+ \sum_{y=1}^{K} \mathbb{P}_{XY}[r_f(X,Y) \geq r_f(X,y)] \cdot \alpha - \sum_{y=1}^{K} \mathbb{P}_{XY}[r_f(X,Y) \geq r_f(X,y)] \cdot \alpha \\
= &\sum_{y=1}^{K} \mathbb{P}_{XY}([r_f(X,Y) < r_f(X,y)] + [r_f(X,Y) \geq r_f(X,y)]) \cdot \alpha \\
&- \sum_{y=1}^{K} \mathbb{P}_{XY}[r_f(X,Y) \geq r_f(X,y)] \cdot \alpha \\
= &K\alpha - \sum_{y=1}^{K} \mathbb{P}_{XY}[r_f(X,Y) \geq r_f(X,y)] \cdot \alpha,
\end{aligned}
$$

where the above inequality (a) is due to Lemma D.3.

Lemma D.3. *For $0 \leq l_0 \leq K - r_f(X,Y)$, according to the definition of conformal prediction, we have:*

$$
\mathbb{P}_{XY}[\sum_{l=1}^{r_f(X,Y)+l_0} f(X)_{(l)} > \tau_{1-\alpha}] = \mathbb{E}_{XY}[\mathbb{1}[\sum_{l=1}^{r_f(X,Y)+l_0} f(X)_{(l)} > \tau_{1-\alpha}]] < \alpha.
$$

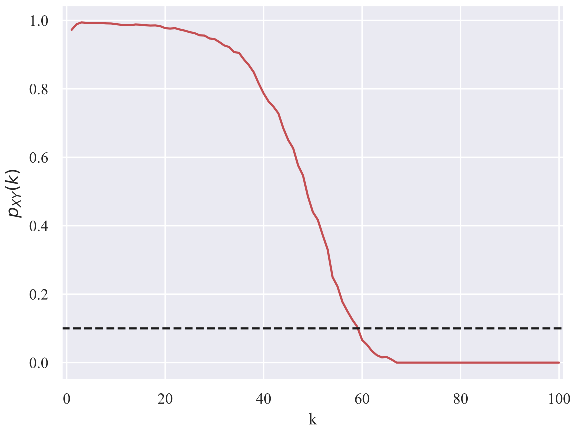

Figure 9. The empirical estimation of $\mathbb{P}_{XY}[\sum_{l=1}^{k} f(X)_{(l)} \leq \tau_{1-\alpha} | r_f(X,Y) \geq k]$ with the test set of CIFAR100 using an AT-trained model. The black dashed line is the confidence level α=0.1.

$$
\begin{aligned}
& B - \sum_{y=1}^{K} \mathbb{P}_{XY}[r_f(X,Y) \geq r_f(X,y)] \cdot \alpha \\
= & \sum_{y=1}^{K} \mathbb{E}_{XY}[\mathbb{1}[V(X,y) \leq \tau_{1-\alpha}] \cdot \mathbb{1}[r_f(X,Y) \geq r_f(X,y)]] - \sum_{y=1}^{K} \mathbb{P}_{XY}[r_f(X,Y) \geq r_f(X,y)] \cdot \alpha \\
= & \sum_{y=1}^{K} \mathbb{P}_{XY}[V(X,y) \leq \tau_{1-\alpha}, r_f(X,Y) \geq r_f(X,y)] - \sum_{y=1}^{K} \mathbb{P}_{XY}[r_f(X,Y) \geq r_f(X,y)] \cdot \alpha \\
= & \sum_{y=1}^{K} \mathbb{P}_{XY}[r_f(X,Y) \geq r_f(X,y)] \cdot \mathbb{P}_{XY}[V(X,y) \leq \tau_{1-\alpha} | r_f(X,Y) \geq r_f(X,y)] \\
& - \sum_{y=1}^{K} \mathbb{P}_{XY}[r_f(X,Y) \geq r_f(X,y)] \cdot \alpha \\
= & \sum_{y=1}^{K} \mathbb{P}_{XY}[r_f(X,Y) \geq r_f(X,y)] \cdot (\mathbb{P}_{XY}[V(X,y) \leq \tau_{1-\alpha} | r_f(X,Y) \geq r_f(X,y)] - \alpha) \\
= & \sum_{y=1}^{K} \mathbb{P}_{XY}[r_f(X,Y) \geq r_f(X,y)] \cdot (\mathbb{P}_{XY}[\sum_{l=1}^{r_f(X,y)} f(X)_{(l)} \leq \tau_{1-\alpha} | r_f(X,Y) \geq r_f(X,y)] - \alpha) \\
\overset{(a)}{=} & \sum_{k=1}^{K} \mathbb{P}_{XY}[r_f(X,Y) \geq k] \cdot (\mathbb{P}_{XY}[\sum_{l=1}^{k} f(X)_{(l)} \leq \tau_{1-\alpha} | r_f(X,Y) \geq k] - \alpha) \\
= & \sum_{k=1}^{K^*} \mathbb{P}_{XY}[r_f(X,Y) \geq k] \cdot (\mathbb{P}_{XY}[\sum_{l=1}^{k} f(X)_{(l)} \leq \tau_{1-\alpha} | r_f(X,Y) \geq k] - \alpha) \\
& + \sum_{k=K^*+1}^{K} \mathbb{P}_{XY}[r_f(X,Y) \geq k] \cdot (\mathbb{P}_{XY}[\sum_{l=1}^{k} f(X)_{(l)} \leq \tau_{1-\alpha} | r_f(X,Y) \geq k] - \alpha) \\
\overset{(b)}{\leq} & \sum_{k=1}^{K^*} \mathbb{P}_{XY}[r_f(X,Y) \geq k] \cdot (\mathbb{P}_{XY}[\sum_{l=1}^{k} f(X)_{(l)} \leq \tau_{1-\alpha} | r_f(X,Y) \geq k] - \alpha),
\end{aligned}
$$

where the above equality (a) is due to $k = r_f(X, y)$, the inequality (b) is due to the definition of $K^* = \max\{k \in [K] : \mathbb{P}_{XY}[\sum_{l=1}^{k} f(X)_{(l)} \leq \tau_{1-\alpha} | r_f(X,Y) \geq k] \geq \alpha\}$ and the assumption of the monotonically decreasing function of $\mathbb{P}_{XY}[\sum_{l=1}^{k} f(X)_{(l)} \leq \tau_{1-\alpha} | r_f(X,Y) \geq k]$ in k. We plot the empirical estimation of $\mathbb{P}_{XY}[\sum_{l=1}^{k} f(X)_{(l)} \leq \tau_{1-\alpha} | r_f(X,Y) \geq k]$ with the test set (adversarially attacked by PGD100) of CIFAR100 using an AT-trained model in Fig. 9, which validates our assumption on the monotonically decreasing property of this function.

Combining the above two inequalities together, we have

$$\begin{aligned}
&\mathbb{E}_X[|\mathcal{C}_f(X)|] \\
\leq & K\alpha + \sum_{k=1}^{K^*} \mathbb{P}_{XY}[r_f(X,Y) \geq k] \cdot (\mathbb{P}_{XY}[\sum_{l=1}^{k} f(X)_{(l)} \leq \tau_{1-\alpha} | r_f(X,Y) \geq k] - \alpha) \\
\leq & K\alpha + \sum_{k=1}^{K^*} \mathbb{P}_{XY}[r_f(X,Y) \geq k] \\
= & K\alpha + \sum_{k=1}^{K^*} k \cdot \mathbb{P}_{XY}[r_f(X,Y) = k].
\end{aligned}$$

After dropping the constant (since the training does not optimize the constant), this completes the proof for Lemma D.2. □

Lemma D.4. *(Partial average rank upper bounded by L_{Beta})*

$$\sum_{k=1}^{K^*} k \cdot \mathbb{P}[r_f(X,Y) = k] \leq \sum_{k=1}^{K} \sigma_k \cdot \mathbb{E}[\ell(f(X), Y) | r_f(X,Y) = k], \tag{14}$$

where $\sigma_k = 8\gamma \cdot \xi \cdot p_{Beta}(k/(K+1); a, b)$, γ is a positive constant satisfying $\bar{l}_k \geq k/\gamma, \forall k \in [K^*]$ and ξ is a positive constant satisfying $p_k \leq \xi \cdot (1 - \frac{k}{K+1})^{b-1}, \forall k \in [K^*]$.

Proof. (of Lemma D.4)
We use p_k to denote $\mathbb{P}[r_f(X,Y) = k]$ and $\bar{l}_k$ to denote $\mathbb{E}[\ell(f(X), Y) | r_f(X,Y) = k]$.

$$\begin{aligned}
&\frac{k \cdot p_k}{\bar{l}_k} \overset{(a)}{\leq} \frac{k \cdot p_k}{k/\gamma} \overset{(b)}{\leq} \gamma \cdot \xi \cdot (1 - \frac{k}{K+1})^{b-1} \overset{(c)}{\leq} \gamma \cdot \xi \cdot \frac{4}{(K+1)^{a-1}} \cdot (1 - \frac{k}{K+1})^{b-1} \\
&\overset{(d)}{\leq} 4\gamma \cdot \xi \cdot \frac{k^{a-1}}{(K+1)^{a-1}} \cdot (1 - \frac{k}{K+1})^{b-1} \cdot \frac{\Gamma(a)\Gamma(b)}{\Gamma(a+b)} \cdot \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} \\
&= 4\gamma \cdot \xi \cdot p_{\text{Beta}}(k/(K+1); a, b) \cdot \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} \\
&\overset{(e)}{\leq} 8\gamma \cdot \xi \cdot p_{\text{Beta}}(k/(K+1); a, b) = \sigma_k,
\end{aligned}$$

where the above inequality (a) is due to the assumption $\bar{l}_k \geq k/\gamma$, the inequality (b) is due to the assumption $p_k \leq \xi \cdot (1 - \frac{k}{K+1})^{b-1}$, the inequality (c) is due to the assumption $K \leq 4^5$ and $a - 1 \leq 1/5$, the inequality (d) is due to $1 \leq k$, the inequality (e) is due to Lemma D.5. We plot the curve of $\bar{l}_k$ versus k/γ ($\gamma = 10$) and p_k versus $\xi(1 - \frac{k}{K+1})^{b-1}$ ($\xi = 0.5$, $b = 5$) with an adversarially trained model on CIFAR100 in Fig. 10, indicating that the two assumptions are valid in practice. This proves the inequality

$$\sum_{k=1}^{K^*} k \cdot \mathbb{P}[r_f(X,Y) = k] \leq \sum_{k=1}^{K^*} \sigma_k \cdot \mathbb{E}[\ell(f(X), Y) | r_f(X,Y) = k],$$

and this completes the proof of Lemma D.4

Lemma D.5. *(Upper bound of Gamma functions)*

$$\frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} \leq 2.$$

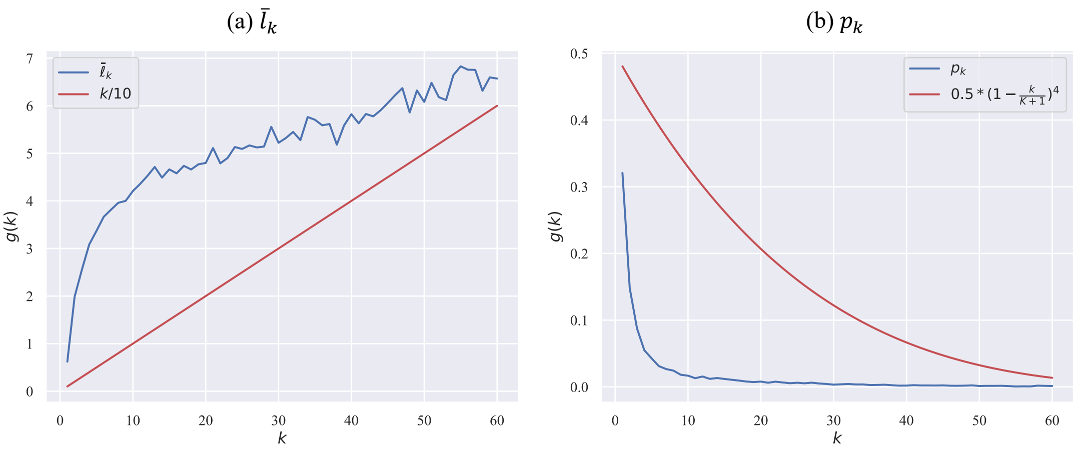

Figure 10. The empirical estimation of **(a)** $\bar{l}_k$ and **(b)** p_k on the test set of CIFAR100, with an adversarially trained model on CIFAR100. We only show k ≤60 as Fig. 9 suggests $K^* \approx 60$ in this experiment.

Proof. (of Lemma D.5)

We use Weierstrass's definition for Gamma function:

$$\Gamma(z) = \frac{\exp(-\gamma z)}{z} \prod_{i=1}^{\infty} (1 + z/i)^{-1} \cdot \exp(z/i),$$

where γ_0 is the Euler–Mascheroni constant. Now we start the proof:

$$\begin{aligned}
\frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} =& \frac{\exp(-\gamma_0 a)}{a} \cdot \frac{\exp(-\gamma_0 b)}{b} \cdot \frac{a+b}{\exp(-\gamma_0(a+b))} \cdot \\
& \prod_{i=1}^{\infty} (1+a/i)^{-1} \cdot (1+b/i)^{-1} \cdot (1+(a+b)/i) \cdot \exp(a/i + b/i - (a+b)/i) \\
=& \frac{a+b}{ab} \cdot \prod_{i=1}^{\infty} \frac{1+(a+b)/i}{(1+a/i)(1+b/i)} \overset{(a)}{<} 2 \cdot \prod_{i=1}^{\infty} \frac{(i+a+b)i}{(i+a)(i+b)} \\
=& 2 \cdot \prod_{i=1}^{\infty} (1 - \frac{ab}{i^2 + (a+b)i + ab}) \overset{(b)}{\leq} 2 \cdot \prod_{i=1}^{\infty} \exp(-\frac{ab}{i^2 + (a+b)i + ab}) \\
=& 2 \cdot \exp(-ab \cdot \sum_{i=1}^{\infty} \frac{1}{i^2 + (a+b)i + ab}) \overset{(c)}{\leq} 2 \cdot \exp(-ab \cdot \sum_{i=1}^{\infty} \frac{1}{i^2 + 11.2 \cdot i + 12}) \\
\overset{(d)}{\leq}& 2 \cdot \exp(-ab/5) \overset{(e)}{<} 2 \cdot \exp(-1/5) \leq 2,
\end{aligned}$$

where the inequality (a) is due to $1/a < 1, 1/b < 1$, the inequality (b) is due to $1 + x \leq \exp(x)$, the inequality (c) is due to $a \leq 1.2, b \leq 10$, the inequality (d) is due to

$$\sum_{i=1}^{n} \frac{1}{i^2 + 11.2 \cdot i + 12} > \frac{1}{5},$$

when $n > 1000$, and the inequality (e) is due to $a > 1, b > 1$. This completes the proof of Lemma D.5. □

This completes the proof of Lemma D.4. □

Now we can start proving Theorem D.1. By inequality (13) from Lemma D.2 and (14) from Lemma D.4, we have

$$\mathbb{E}_X[|\mathcal{C}_f(X)|] \overset{(13)}{\leq} \sum_{k=1}^{K^*} k \cdot \mathbb{P}[r_f(X,Y)=k] \overset{(14)}{\leq} \sum_{k=1}^{K} \sigma_k \cdot \mathbb{E}[\ell(f(X),Y)|r_f(X,Y)=k] = L_{\text{Beta}}(f).$$

This completes the proof of Theorem D.1 □