Towards Fairness-Aware Adversarial Learning

Supplementary Material

A. Proof of theorem 1

Definition 1 (CDAW: Class-wise Distributionally Adversarial Weight)**.** *Given a class-wise objective loss $\ell'_c \in \mathbb{R}$ on the adversarial examples, for all classes $c \in C$, the optimal Class-wise Distributionally Adversarial Weight vector $\boldsymbol{w}^{\boldsymbol{cda}}_*$ belonging to the probability simplex Δ_C aims to maximize the overall loss:*

$$\begin{aligned} \mathcal{L}_{\mathrm{FAAL}} &:= \max \sum_{c=1}^{C} w_c^{cda} \ell'_c \\ \textit{s.t.} \quad & d(\mathcal{U}, \boldsymbol{w}^{\boldsymbol{cda}}) \leq \tau, \boldsymbol{w}^{\boldsymbol{cda}} \in \Delta_C \end{aligned} \tag{1}$$

$$\boldsymbol{w}^{\boldsymbol{cda}}_* := \arg\max \mathcal{L}_{\mathrm{FAAL}} \tag{2}$$

Theorem 1. *Given the loss $\mathcal{L}_{\mathrm{FAAL}}$ defined in Eq. (1) on the observed distribution, and suppose the regular loss $\mathcal{L} =: \frac{1}{C}\sum_{c=1}^{C} \ell'_c$ on the test distribution with unknown group distribution shift, then the following holds for all $\boldsymbol{w}^{\boldsymbol{cda}} \in \Delta_C$:*

$$\Pr(\mathcal{L}_{\mathrm{FAAL}} > \mathcal{L}) \geq 1 - e^{-\tau n + O(n)} \tag{3}$$

Where $\mathrm{KL}(\mathcal{U}, \boldsymbol{w}^{\boldsymbol{cda}}) \leq \tau$, $\mathcal{U}$ is the uniform distribution.

Lemma 1. *[5, 12] Let $\widehat{\mathcal{D}}_n$ be the empirical distribution of n independent samples with distribution $\mathcal{D}$, then:*

$$\Pr(\mathcal{D} \in \mathcal{D}' \quad \textit{s.t.} \quad \mathrm{KL}(\widehat{\mathcal{D}}, \mathcal{D}') \leq r) \geq 1 - e^{-rn + O(n)} \tag{4}$$

Proof. Lemma 1 tells that the probability of $\mathcal{D} \in \mathcal{D}'$ (out-of-sample disappointment) decays at a prescribed exponential rate r as the sample size n tends to infinity—irrespective of true data-generating distribution. Here the out-of-sample disappointment quantifies the probability that the actual expected loss of the model exceeds its predicted loss.

Hence when considering the associated loss and replacing r as our defined constraint parameter τ, $\mathcal{L}_{\mathrm{FAAL}}$ can be rewritten as:

$$\mathcal{L}_{\mathrm{FAAL}} := \max \mathbb{E}_{\mathcal{D}'}[\ell(\theta)] \quad \textit{s.t.} \quad \mathrm{KL}(\widehat{\mathcal{D}}, \mathcal{D}') \leq \tau \tag{5}$$

Similarly $\mathcal{L}$ can be written as $\mathcal{L} := \mathbb{E}_{\mathcal{D}}[\ell(\theta)]$ with $\mathcal{D}$ may having unknown distribution shift, by replacing $\widehat{\mathcal{D}}$ with uniform distribution $\mathcal{U}$ and replacing the $\mathcal{D}'$ with $\boldsymbol{w}^{\boldsymbol{cda}}$, then we can get:

$$\Pr(\mathcal{L}_{\mathrm{FAAL}} \geq \mathcal{L}, \forall \boldsymbol{w}^{\boldsymbol{cda}} \in \Delta_C) > 1 - e^{-rn + O(n)} \tag{6}$$

This indicates $\mathcal{L}_{\mathrm{FAAL}}$ remains as an upper bound on the mean loss $\mathcal{L}$ uniformly over $\boldsymbol{w}^{\boldsymbol{cda}} \in \Delta_C$ with high probability. □

Algorithm 1 Fairness-Aware Adversarial Learning

Input: Training set $\{X, Y\}$, total epochs T, adversarial radius ϵ, step size α, the number of adversarial iteration K, model f parameterized by θ, the number of mini-batches M, batch size B, distribution shift constraint τ

Output: A robust and fair model

```
1:  for $t = 1, \ldots, T$ do
2:      for $i = 1, \ldots, M$ do
3:          # Phase 1: Inner maximization
            $\delta = 0$
4:          for $j = 1, \ldots, K$ do
5:              $\delta = \delta + \alpha \cdot \mathrm{sign}(\nabla_\delta \ell_{\mathrm{CE}}(f_\theta(x_i + \delta), y_i))$
6:              $\delta = \max(\min(\delta, \epsilon), -\epsilon)$
7:          end for
8:          $x_i^{adv} = \mathrm{clip}(x_i + \delta, 0, 1)$
9:          # Phase 2: Intermediate maximization
            $\ell_i = \ell_{\mathrm{CE}}(f_\theta(x_i^{adv}), y_i, \mathrm{reduction} =$'none'$)$
            # Calculate the cross-entropy loss for each instance
10:         for $c = 1 \ldots C$ do
11:             $\ell'_c = \ell_{\mathrm{CW}}(f_\theta(x_i^{adv}, y_i)[y_i = c]$
                # Calculate the average margin for each class $c$
12:         end for
13:         $\boldsymbol{w}^{\boldsymbol{cda}}_* = \texttt{solve\_kl\_dro}(\boldsymbol{\ell'}, \tau)$
            # Solve the optimal class-wise weights for the current batch under the worst distribution via DRO
14:         $\mathcal{L}_{\mathrm{FAAL}} = \frac{1}{B}\sum_{i=1}^{B} \boldsymbol{w}^{\boldsymbol{cda}}_*[y] \cdot \ell_i \cdot C$
15:         # Phase 3: Outer Minimization
            $\theta = \theta - \nabla_\theta \mathcal{L}_{\mathrm{FAAL}}$
16:     end for
17: end for
18: return Robust model $f_\theta$ with high fairness
```

Our assumption is that the robust fairness issue in the conventional AT is due to the overfitting of the unknown group (class) distribution shift induced by adversarial perturbations. Therefore, by optimizing with the upper loss $\mathcal{L}_{\mathrm{FAAL}}$, we are able to better deal with some unknown distribution shift **(even if they have not been during training)**, as will be indicated in Sec. C.1.

B. Algorithm Details

Algorithm 1 demonstrates the whole pseudo-code for the proposed FAAL framework, here we further explain the detail of *Phase 2: Intermediate maximization*. Here we extend *Phase 2* into PGD-AT by default as shown in lines 9-14. It is noted that our framework is completely compatible with

Table 1. Comparison among different adversarial training methods using Preact-ResNet18 model regarding the Clean/AA accuracy on CIFAR-10 dataset and the corresponding corruption accuracy of different corruptions on CIFAR-10-C dataset. The best performance is highlighted in **Bold**.

PRN-18 Model	Average Acc (Worst-class Acc) (%)										
	CIFAR-10		CIFAR-10-C (Noise)			CIFAR-10-C (Weather)		CIFAR-10-C (Blur)		CIFAR-10-C (Digital)	
	Clean	AA	Gaussian	Shot	Impulse	Brightness	Snow	Zoom	Motion	JPEG	Pixelation
TRADES	**82.54** (66.10)	49.05 (20.70)	**78.82** (57.26)	**79.72** (59.14)	**75.59** (51.98)	**78.60** (65.86)	**76.77** (61.00)	**77.17** (**65.16**)	**74.53** (**60.30**)	**80.69** (64.56)	**80.77** (64.44)
$\text{CFA}_{\text{TRADES}}$	80.36 (66.20)	**50.10** (26.50)	75.57 (56.00)	76.71 (58.08)	72.83 (51.98)	75.61 (63.18)	74.27 (55.32)	72.88 (58.42)	70.35 (49.66)	78.01 (62.38)	77.72 (62.82)
$\text{WAT}_{\text{TRADES}}$	80.37 (66.00)	46.16 (30.70)	76.32 (59.20)	77.21 (60.08)	72.44 (56.90)	75.04 (64.68)	72.83 (60.42)	73.22 (58.06)	71.09 (52.00)	77.94 (62.74)	77.89 (61.76)
$\text{FAAL}_{\text{TRADES}}$	81.62 (68.90)	48.48 (33.60)	77.42 (62.36)	78.31 (63.34)	74.10 (59.08)	76.62 (65.56)	74.53 (60.86)	73.26 (60.48)	70.77 (52.34)	78.83 (65.60)	78.67 (64.88)
$\text{FAAL}_{\text{TRADES-AWP}}$	82.19 (**72.00**)	48.08 (**35.00**)	77.60 (**65.02**)	78.67 (**66.16**)	73.66 (**61.58**)	76.81 (**67.30**)	74.34 (**61.04**)	73.84 (63.50)	71.67 (55.16)	79.38 (**68.28**)	79.50 (**67.96**)

other min-max-based adversarial training approaches like TRADES [16] or MART [13]. To achieve this, one just needs to keep the original implementation for both inner maximization and outer minimization unchanged and add the intermediate maximization independently.

Therefore, after generating adversarial examples by PGD in *Phase 1*, we first calculate the cross-entropy loss for each adversarial instance, then we compute the average CW margin loss for each class $\boldsymbol{\ell'} = [\ell'_1, ..., \ell'_c]$ (because there may be multiple samples for one class in a batch). It is noted that normally we are expecting that the batch-size is large enough to include samples from all classes, however, when there this no sample for a particular class in batch, *e.g.* for a large number of classes dataset CIFAR-100, we manually set the margin to -1. We utilize these class margins to solve the Class-wise Distributionally Adversarial Weight $\boldsymbol{w^{cda}}$ defined in Eq. (2) with constraint τ. By solving this convex optimization in the function `solve_kl_dro` thought the *cvxpy* solver, then $\boldsymbol{w^{cda}}[y]$ will map the optimal weights to all samples according to their class labels. In this way, when $\tau = 0$, $\boldsymbol{w^{cda}}$ becomes a uniform distribution with all values equal to $\frac{1}{C}$, and the proposed method degrades to the conventional PGD-AT. Therefore, FAAL can be seen as an extension on the conventional adversarial training framework, where the whole optimization is from **min-max** to **min-max-max**. Although we perform this convex optimization in every batch, the resulting extra computation is still negligible as shown in the experiments.

It is noted that all the existing state-of-the-art related works addressing robust fairness do not require the mini-batch should be balanced including ours. This is because we are considering a class-balanced dataset, *i.e.* all classes have the same number of samples, thus the expected value of the mini-batch gradient estimator (like an ideal Neural Network) is equal to the true gradient [1], irrespective of the size. In other words, we expect that a deep neural neural has enough capacity to handle this situation during training. Dealing with the imbalanced dataset like applying sampling strategy [11] is out of the focus of this paper.

C. Experiments Details

All our experiments are conducted on a single NVIDIA 3090ti GPU with 24GB of graphics memory.

C.1. Enhanced Robustness against Unknown Distribution Shift

As our assumption is that the robust fairness issue in the conventional AT is due to the over-fitting of the unknown group (class) distribution shift induced by adversarial perturbations, we are expecting that the proposed method is able to handle other unknown distribution shift noises apart from the adversarial noise that we are focusing on in the paper. Therefore, we further validate our assumption by conducting the extra experiments on CIFAR-10-C dataset [7], as this dataset is unseen during training, it can be regarded as the natural unknown distribution shift. CIFAR-10-C is an extended dataset based on CIFAR-10 test dataset [8], which contains several common corruptions that may occur in real-world situations. We report the average/worst-class corruption accuracy against the following corruptions:

- Gaussian noise: can appear in low-lighting conditions.
- Shot noise: also called Poisson noise, is electronic noise caused by the discrete nature of light itself.
- Impulse noise: a color analog of salt-and-pepper noise and can be caused by bit errors.
- Brightness: varies with daylight intensity.
- Snow: is a visually obstructive form of precipitation.
- Zoom blur: occurs when a camera moves toward an object rapidly.
- Motion blur: appears when a camera is moving quickly.
- JPEG: is a lossy image compression format that introduces compression artifacts.
- Pixelation: when upsampling a low-resolution image

The results depicted in Tab. 1 highlight the robustness of various PRN-18 models against different types of corruptions, with each model having been adversarially trained using the methodologies outlined in the main manuscript. It is evident that the CFA approach struggles to uphold high worst-case accuracy across all categories of corruption (noise, weather, blur, and digital). While WAT demonstrates some degree of robust fairness against noise corruption, it, unfortunately, compromises both the average and worst-case accuracy in the remaining categories (weather, blur, and digital). In contrast, FAAL shows a commendable ability to reserve robust fairness in noise, weather, and digital corruptions, albeit with a minor reduction in average accuracy. This underscores its enhanced capability for handling various corruptions and unknown distribution shifts, aligning with our initial hypotheses. An interesting observation is the consistent adverse impact of adversarial training on the robustness against blur corruption across all models. This could be attributed to a significant divergence between the adversarially learned noise during training and the characteristics of blurring, and even our approach with small τ can not fully cover this type of distribution shift. Notably, in this context, FAAL still outperforms the other two advanced models, underscoring its effectiveness.

C.2. Objective Loss for Solving the Intermediate Maximization

Figure 1 plots the results of fine-tuning the Preact-ResNet18 model on CIFAR-10 dataset using different objective losses in the proposed intermediate maximization. It can be easily observed that by replacing the original cross-entropy loss with the CW margin loss, the performance of robust fairness can be promoted better. This is because the magnitude of cross-entropy loss does not reflect how well the class has been learned, as indicated in [2]. Conversely, the margin loss represents the marginal discrepancy between the true label and the most suspicious category, the smaller the margin we obtain, the more robust performance we can achieve. In addition, margin loss is bound by $[-1, 1]$, while cross-entropy loss is unbounded from 0 to ∞.

C.3. Adversarial Fine-tuning for Preact-ResNet18 Model on CIFAR-10 Dataset

We first compare our methods with the conventional adversarial training and FRL on CIFAR-10 dataset [8] using Preact-ResNet18 (PRN-18) model [6]. We adversarially trained the models without considering robust fairness by conventional PGD [10] and TRADES [16], respectively. Similarly, we apply the best versions of FRL [15]: FRL-RWRM with $\tau_1 = \tau_2 = 0.05$ and FRL-RWRM with $\tau_1 = \tau_2 = 0.07$, where τ_1 and τ_2 are the fairness constraint parameters for reweight and remargin of FRL, we name them FRL-RWRM$_{0.05}$ and FRL-RWRM$_{0.07}$ for short, and the target models are fine-tuned for 80 epochs and the best results are presented. As for FAAL, we set the value of τ in our method as 0.5 and fine-tune the original pre-trained models for 2 epochs. The learning rate is configured from 0.01 in the first epoch and drops to 0.001 in the second epoch. We report the average & worst-class accuracy under different adversarial attacks (Clean / PGD [10] / CW [3] / AutoAttack [4]) as the evaluation metrics. The perturbation budget is set to $\epsilon = 8/255$ on CIFAR-10 dataset.

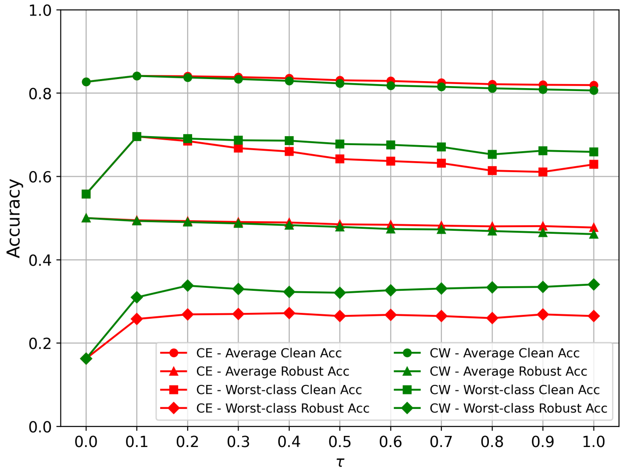


Figure 1. CE *vs*. CW: Accuracy for fine-tuning Preact-ResNet18 models using different objectives during the intermediate maximization on CIFAR-10

As shown in Tab. 2, given two adversarial trained Preact-ResNet18 models with different adversaries (PGD-AT and TRADES) on CIFAR-10 dataset, the proposed method FAAL is able to fine-tune both models to promote the robust fairness. It can be easily observed that FAAL$_{\mathrm{AT}}$ can obtain better results on the overall and worst-class robustness, even if the TRADES model is not adversarially trained by the PGD adversary. Most importantly, FRL requires many epochs (*80 epochs*) to obtain the final results, while our method, is able to achieve slightly better results within *only 2 epochs*. It is noted that the improvement is not as huge as that on the large Wide-ResNet model as discussed in the main manuscript, the full table of it can be seen in Tab. 3. This discrepancy may be attributed to the differences in model architecture, where the larger model, endowed with a greater number of parameters, is inherently better suited to executing more complex training paradigms.

C.4. Working with Other Adversaries?

Although we use the PGD-AT adversary by default for adversarial fine-tuning, our method can be easily integrated with other techniques like TRADES and TRADES-AWP as well. Tables 2 and 3 reports the results of the PRN/WRN models on CIFAR-10 dataset, where the different adversaries are applied to fine-tune the original unfair model. We can observe that fine-tuning the model with AT-based

Table 2. Evaluation of different fine-tuning methods on CIFAR-10 dataset using PRN-18 model. The best result is highlighted in **Bold**.

Adversarially Trained PRN-18 Model	Fine-Tuning Epochs	Average Accuracy (Worst Class Accuracy) (%)			
		Clean	PGD-20	CW-20	AA
PGD-AT	-	**82.72** (55.80)	50.92 (16.80)	**50.14** (16.50)	**47.38** (12.90)
+ Finetune with FRL-RWRM$_{0.05}$	80	80.92 (**72.20**)	48.18 (34.20)	46.13 (32.30)	44.54 (30.70)
+ Finetune with FRL-RWRM$_{0.07}$	80	82.47 (71.90)	49.11 (35.20)	47.20 (32.30)	45.57 (30.00)
+ Fine-tune with FAAL$_{\text{AT}}$	**2**	82.34 (67.80)	49.97 (34.80)	48.15 (32.30)	45.51 (**31.10**)
+ Fine-tune with FAAL$_{\text{AT-AWP}}$	**2**	82.08 (66.90)	**51.62** (**38.70**)	48.89 (**35.00**)	46.68 (30.10)
+ Fine-tune with FAAL$_{\text{TRADES}}$	**2**	77.96 (70.80)	48.57 (37.90)	45.32 (32.10)	44.37 (30.70)
+ Fine-tune with FAAL$_{\text{TRADES-AWP}}$	**2**	77.61 (66.30)	49.99 (40.30)	45.89 (34.20)	44.48 (30.20)
TRADES	-	82.54 (66.10)	52.29 (24.60)	**50.59** (22.50)	**49.05** (20.70)
+ Fine-tune with FRL-RWRM $_{0.05}$	80	80.71 (67.80)	48.22 (34.70)	46.56 (30.70)	44.64 (28.30)
+ Fine-tune with FRL-RWRM $_{0.07}$	80	82.12 (65.90)	47.62 (33.10)	46.23 (30.00)	44.60 (27.90)
+ Fine-tune with FAAL$_{\text{AT}}$	**2**	**83.67** (**71.80**)	48.93 (34.60)	47.51 (32.70)	45.10 (28.80)
+ Fine-tune with FAAL$_{\text{AT-AWP}}$	**2**	83.26 (69.40)	**52.42** (**39.40**)	49.83 (**33.50**)	47.62 (30.70)
+ Fine-tune with FAAL$_{\text{TRADES}}$	**2**	81.38 (69.40)	50.01 (34.00)	47.66 (30.50)	46.42 (28.30)
+ Fine-tune with FAAL$_{\text{TRADES-AWP}}$	**2**	80.43 (68.10)	51.45 (37.90)	48.04 (32.10)	47.09 (**31.30**)

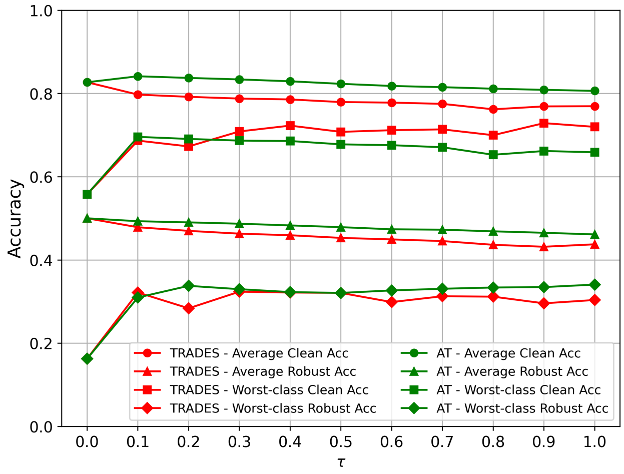


Figure 2. Fine-tuning a pre-trained PGD-AT model using FAAL$_{\text{AT}}$ and FAAL$_{\text{TRADES}}$ (different adversaries) on CIFAR-10

(PGD-AT) or TRADES-based adversaries both can boost the worst-class accuracy. And TRADES-based adversaries can perform slightly better than AT-based on the worst-class AutoAttack accuracy. However, when applying FAAL with a TRADES-based adversary, it results in a larger drop in the overall clean/robust accuracy, but AT-based FAAL is able to maintain the average clean/robust accuracy while boosting the worst-class accuracy. After considering the overall performance, we decided to use the proposed FAAL with PGD-AT adversary as our default setting. Figure 2 illustrates this phenomenon more clearly.

C.5. Adversarial Training for Preact-ResNet18 Model on CIFAR-10 Dataset

We also train the Preact-ResNet18 model *from scratch* using different methods, including PGD-AT [10], TRADES [16], CFA [14], WAT [9], FAAL with different adversaries. All models are trained for 200 epochs with the learning rate decaying by a factor of 0.1 at the 100-th and 150-th epochs, successively. In addition, we applied the default setting and the hyper-parameters including batch size 128; SGD momentum optimizer with the initial learning rate of 0.1; weight decay 5×10^{-4}; ReLU activation function and no label smoothing. The results are reported in Tab. 4. It can be clearly observed that our method FAAL consistently outperforms the other state-of-the-art approaches most of the time. Compared to the result of fine-tuning, training with more epochs indeed brings some benefits, not only from the average robustness but also from the worst-class robustness. A clear demonstration can be seen in Fig. 3, where our method is able to alleviate the issue of robust fairness by promoting the worst-case performance across all categories.

Table 3. Evaluation of different fine-tuning methods on CIFAR-10 dataset using WRN-34-10 model. The best result is highlighted in **Bold**.

Adversarially Trained WRN-34-10 Model	Fine-Tuning Epochs	Average Accuracy (Worst Class Accuracy) (%)			
		Clean	PGD-20	CW-20	AA
PGD-AT	-	86.07 (69.70)	55.90 (29.90)	54.29 (28.30)	52.46 (24.40)
+ Fine-tune with FRL-RWRM$_{\lambda=0.05}$	80	83.25 (**74.80**)	50.37 (38.10)	49.77 (36.60)	46.97 (33.10)
+ Fine-tune with FRL-RWRM$_{\lambda=0.07}$	80	85.12 (71.60)	52.56 (37.10)	51.92 (35.50)	49.60 (31.70)
+ Fine-tune with FAAL$_{\mathrm{AT}}$	**2**	**86.23** (69.70)	54.00 (37.60)	53.11 (36.90)	50.81 (35.70)
+ Fine-tune with FAAL$_{\mathrm{AT-AWP}}$	**2**	85.47 (69.40)	**56.46** (39.20)	**54.50** (**38.10**)	**52.47** (**36.90**)
+ Fine-tune with FAAL$_{\mathrm{TRADES}}$	**2**	83.02 (74.10)	53.14 (40.00)	51.49 (37.40)	50.31 (36.50)
+ Fine-tune with FAAL$_{\mathrm{TRADES-AWP}}$	**2**	80.99 (73.40)	53.38 (**41.30**)	49.66 (33.80)	48.94 (32.60)
TRADES	-	84.92 (67.00)	55.32 (27.10)	53.92 (24.80)	**52.51** (23.20)
+ Fine-tune with FRL-RWRM$_{\lambda=0.05}$	80	82.90 (72.70)	53.16 (40.60)	51.39 (36.30)	49.97 (35.40)
+ Fine-tune with FRL-RWRM$_{\lambda=0.07}$	80	85.19 (70.90)	53.76 (39.20)	52.92 (36.80)	51.30 (34.60)
+ Fine-tune with FAAL$_{\mathrm{AT}}$	**2**	**85.96** (**75.00**)	53.46 (39.80)	52.72 (38.20)	50.91 (35.30)
+ Fine-tune with FAAL$_{\mathrm{AT-AWP}}$	**2**	85.39 (72.90)	**56.07** (**43.30**)	**54.16** (**38.60**)	52.45 (35.40)
+ Fine-tune with FAAL$_{\mathrm{TRADES}}$	**2**	83.79 (73.60)	54.44 (39.00)	52.54 (37.30)	51.35 (36.10)
+ Fine-tune with FAAL$_{\mathrm{TRADES-AWP}}$	**2**	82.44 (70.50)	55.45 (39.60)	52.48 (37.90)	51.60 (**36.90**)
MART	-	83.62 (67.90)	**56.22** (32.50)	52.79 (25.70)	**50.95** (22.00)
+ Fine-tune with FRL-RWRM$_{\lambda=0.05}$	80	**83.72** (**71.80**)	52.16 (37.50)	50.73 (35.00)	49.19 (31.70)
+ Fine-tune with FRL-RWRM$_{\lambda=0.07}$	80	82.09 (**71.80**)	50.86 (36.00)	49.78 (33.00)	47.78 (30.30)
+ Fine-tune with FAAL$_{\mathrm{AT}}$	**2**	83.49 (68.00)	51.65 (37.80)	50.36 (37.10)	48.63 (34.00)
+ Fine-tune with FAAL$_{\mathrm{AT-AWP}}$	**2**	82.17 (64.00)	54.31 (39.50)	51.72 (**37.70**)	50.31 (36.40)
+ Fine-tune with FAAL$_{\mathrm{TRADES}}$	**2**	81.65 (70.60)	51.65 (37.80)	50.05 (36.90)	48.73 (35.30)
+ Fine-tune with FAAL$_{\mathrm{TRADES-AWP}}$	**2**	80.08 (68.90)	52.74 (**40.30**)	49.91 (37.40)	48.95 (**36.60**)
TRADES-AWP	-	85.35 (67.90)	59.20 (28.80)	**57.14** (26.50)	**56.18** (25.80)
+ Fine-tune with FRL-RWRM$_{\lambda=0.05}$	80	82.31 (65.90)	49.90 (31.70)	49.68 (34.00)	46.50 (27.70)
+ Fine-tune with FRL-RWRM$_{\lambda=0.07}$	80	84.24 (65.70)	48.63 (30.90)	49.77 (31.50)	46.53 (28.60)
+ Fine-tune with FAAL$_{\mathrm{AT}}$	**2**	87.02 (**76.30**)	52.54 (35.00)	51.70 (34.40)	49.87 (30.60)
+ Fine-tune with FAAL$_{\mathrm{AT-AWP}}$	**2**	**86.75** (74.80)	**57.14** (**43.40**)	55.34 (**40.10**)	53.93 (**37.00**)
+ Fine-tune with FAAL$_{\mathrm{TRADES}}$	**2**	84.62 (74.80)	54.03 (34.00)	52.69 (32.10)	51.37 (30.50)
+ Fine-tune with FAAL$_{\mathrm{TRADES-AWP}}$	**2**	84.25 (**76.10**)	57.35 (38.30)	54.54 (34.30)	53.66 (32.90)

C.6. More Comparisons and Ablation Study

Although CFA [14] and WAT [9] are not proposed for fine-tuning, we also managed to adapt them for fine-tuning. We followed their implementations, respectively and tried to apply them to fine-tune a WRN-34-10 model trained by the PGD adversary. As shown in Tab. 5, both WAT and CFA fail to provide high worst-class robustness within limited epochs, but our FAAL method is able to achieve this within 2 epochs. In addition, we also investigate the impact of the number of epochs in fine-tuning for our proposed FAAL. It can be seen that our method illustrates similar and stable performances for different numbers of epochs.

C.7. Adversarial Fine-tuning/Training for ResNet18 Model on CIFAR-100 Dataset

In this section, we explore the proposed FAAL into a more challenging dataset, *i.e.* CIFAR-100 [8] with 100 categories. It is noted that robust fairness in multi-class classification with a large number of classes can indeed be more challenging and severe compared to small-class problems. Similarly, the perturbation budget is set to $\epsilon = 8/255$ on CIFAR-100 dataset.

The value of τ in our method is set to 0.05 for adversarial fine-tuning for this dataset, and the learning rate is configured from 0.01 in the first epoch and drops to 0.001 in

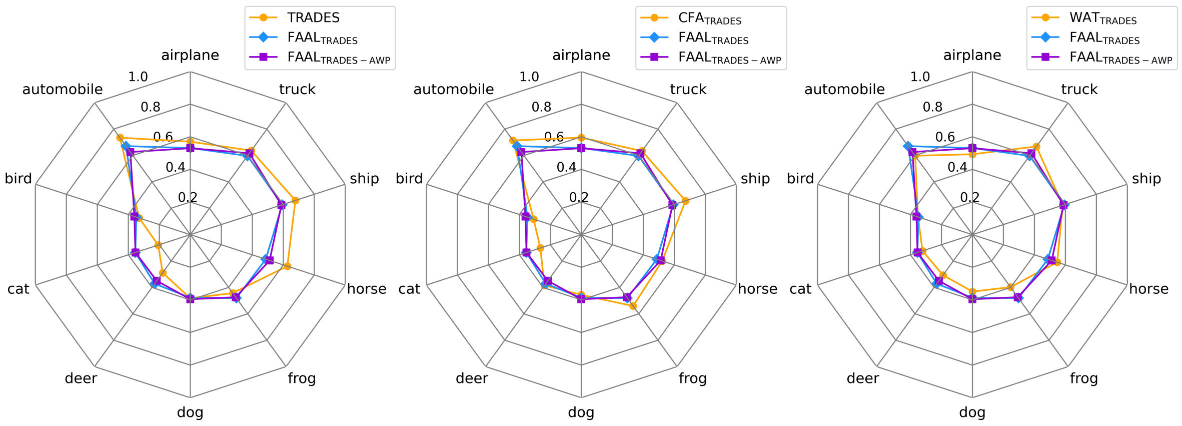


Figure 3. Class-wise robust Accuracy against AutoAttack after adversarially trained PRN-18 model on CIFAR-10 dataset

Table 4. Training from scratch with different methods on CIFAR-10 dataset using Preact-ResNet18 model.

Adversarially Trained PRN-18 Model	Average Acc (Worst-class Acc) (%)	
	Clean	AutoAttack
PGD-AT	82.72 (55.80)	47.38 (12.90)
TRADES	82.54 (66.10)	49.05 (20.70)
CFA$_{\text{AT}}$	80.82 (64.60)	**50.10** (24.40)
CFA$_{\text{TRADES}}$	80.36 (66.20)	**50.10** (26.50)
WAT$_{\text{TRADES}}$	80.37 (66.00)	46.16 (30.70)
FAAL$_{\text{AT}}$	**82.20** (62.90)	49.10 (33.70)
FAAL$_{\text{TRADES}}$	81.62 (68.90)	48.48 (33.60)
FAAL$_{\text{TRADES-AWP}}$	82.19 (**72.00**)	48.08 (**35.00**)

Table 5. More comparisons results for fine-tuning the WRN on CIFAR-10

Adversarially Trained WRN-34-10 Model	Fine-Tuning Epochs	Average Acc (Worst-class Acc) (%)	
		Clean	AutoAttack
PGD-AT	2	86.07 (69.70)	52.46 (24.40)
+ Fine-tune with CFA	2	86.29 (69.50)	50.57 (24.70)
+ Fine-tune with WAT	2	85.30 (68.70)	**53.15** (25.20)
+ Fine-tune with FAAL$_{\text{AT}}$	2	86.23 (69.70)	50.81 (35.70)
+ Fine-tune with FAAL$_{\text{AT}}$	4	86.56 (**75.00**)	50.57 (34.00)
+ Fine-tune with FAAL$_{\text{AT}}$	6	**86.64** (74.20)	50.63 (**35.80**)
+ Fine-tune with FAAL$_{\text{AT}}$	8	86.32 (73.20)	50.30 (34.90)

the second epoch. In terms of the regular adversarial training, we following the experimental settings in WAT [9], applied the hyper-parameters including batch size 128; SGD momentum optimizer with the initial learning rate of 0.1; weight decay 2×10^{-4}; ReLU activation function and no label smoothing. We train the ResNet18 model from scratch using different methods, including PGD-AT [10], TRADES [16], CFA [14], WAT [9], FAAL with different adversaries. All models are trained for 100 epochs with the learning rate decaying by a factor of 0.1 at the 75-th and 90-th epochs, successively. we also start to facilitate the proposed intermediate maximization (see Algorithm 1 lines 9-14) after the 75-th epoch with the only hyper-parameter τ from 0.025 and enlarge it to 0.05 after the 90-th epoch.

Similarly, we reported the results of the average/worst-class clean accuracy and AutoAttack accuracy in Tab. 6. For fine-tuning, we compare our proposed method FAAL with FRL-RWRM$_{0.05}$ [15], it can be seen that FAAL is able to achieve comparable to FRL-RWRM while reducing the amount of learning epoch up to 40 times less (2 epochs *vs*. 80 epochs). For full adversarial training, we compare the results of FAAL compared to three baselines, *i.e*. TRADES, CFA and WAT. It can be seen that FAAT$_{\text{TRADES}}$ achieves the highest worst-class robust accuracy (same as WAT), meanwhile, it remains comparable results on the average robustness without sacrificing the average/worst-class clean accuracy too much.

Table 6. Result comparison of different methods on CIFAR-100 dataset using ResNet18 model.

Adversarially Trained RN-18 Model	Average Acc (Worst-class Acc) (%)	
	Clean	AutoAttack
TRADES	54.57 (19.00)	23.57 (1.00)
+ Fine-tune with FRL-RWRM$_{0.05}$	52.55 (22.00)	21.11 (2.00)
+ Fine-tune with FAAL$_{\text{AT}}$	**58.50** (21.00)	21.91 (2.00)
+ Fine-tune with FAAL$_{\text{AT-AWP}}$	58.41 (19.00)	23.44 (2.00)
+ Fine-tune with FAAL$_{\text{TRADES}}$	54.96 (18.00)	22.71 (2.00)
+ Fine-tune with FAAL$_{\text{TRADES-AWP}}$	54.90 (18.00)	23.25 (2.00)
CFA$_{\text{TRADES}}$	55.57 (**23.00**)	**24.56** (2.00)
WAT$_{\text{TRADES}}$	53.99 (19.00)	22.89 (**3.00**)
FAAL$_{\text{AT}}$	56.84 (16.00)	21.85 (**3.00**)
FAAL$_{\text{TRADES}}$	55.87 (21.00)	23.57 (**3.00**)

C.8. Hyper-parameter Selection

In our method, τ is the only extra hyper-parameter introduced, it represents how much worse the distribution we want to consider around the uniform distribution. Figure 4 plots the average and the average/worst-class clean accuracy and robust accuracy (against CW-100 attack) with the increasing value of τ from $0.$ to 1, where FAAL_{AT} is used to fine-tune a pre-trained TRADES model on the CIFAR-10 dataset. We can see that with the increase of τ, there is a trade-off tendency between the worst-class robustness and average robustness, the improvement of worst-class robustness will scarify some average robustness. By considering all accuracy, we set up the value of 0.5 for τ in FAAL as the default setting for CIFAR-10 dataset. A similar evaluation is made on CIFAR-100 and we use $\tau = 0.05$ by default.

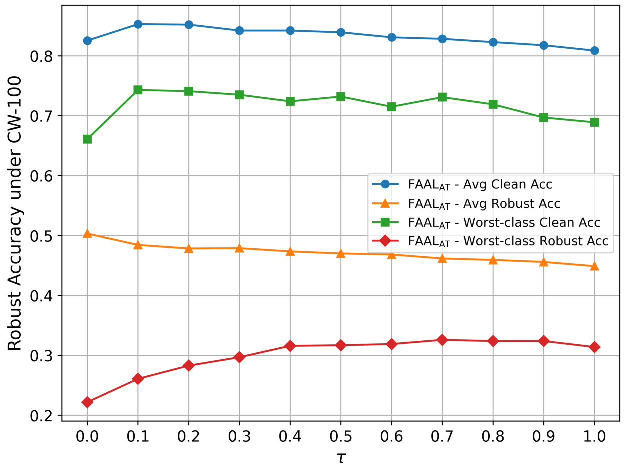


Figure 4. Apply FAAL_{AT} for fine-tuning a TRADES model on CIFAR-10 dataset

C.9. More Visualizations

Figure 5 visualizes the class-wise clean/robust accuracy before and after fine-tuning the WRN model, demonstrating that after applying the proposed FAAL, the robust fairness issue can be alleviated effectively. In Fig. 6, we plot the class weights that the algorithms automatically learned during training, we can see the weights vary significantly for different batches (colored in red). The proposed FAAL directly adapts to these changes in every batch. The average learned weights (last image) in Fig. 6 also show a good agreement with those unfair categories compared to the results in Fig. 5a. In addition, Figs. 7 to 10 gives more visualisation results for fine-tuning across different models.

References

[1] Christopher M Bishop and Nasser M Nasrabadi. *Pattern recognition and machine learning*. Springer, 2006. 2

[2] Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arechiga, and Tengyu Ma. Learning imbalanced datasets with label-distribution-aware margin loss. *Advances in neural information processing systems*, 32, 2019. 3

[3] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *2017 ieee symposium on security and privacy (sp)*, pages 39–57. Ieee, 2017. 3

[4] Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *International conference on machine learning*, pages 2206–2216. PMLR, 2020. 3

[5] Amir Dembo and Ofer Zeitouni. *Large deviations techniques and applications*. Springer Science & Business Media, 2009. 1

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 3

[7] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv preprint arXiv:1903.12261*, 2019. 2

[8] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 2, 3, 5

[9] Boqi Li and Weiwei Liu. Wat: improve the worst-class robustness in adversarial training. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 14982–14990, 2023. 4, 5, 6

[10] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017. 3, 4, 6

[11] Dunlu Peng, Tianfei Gu, Xue Hu, and Cong Liu. Addressing the multi-label imbalance for neural networks: An approach based on stratified mini-batches. *Neurocomputing*, 435:91–102, 2021. 2

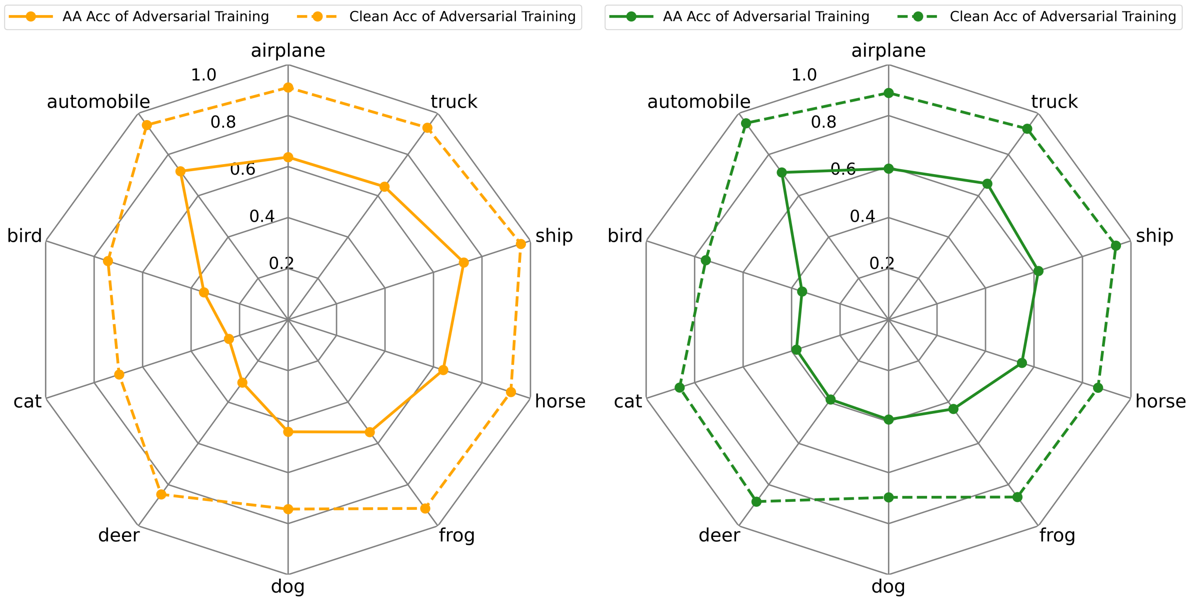
[12] Bart PG Van Parys, Peyman Mohajerin Esfahani, and Daniel Kuhn. From data to decisions: Distributionally robust optimization is optimal. *Management Science*, 67(6):3387–3402, 2021. 1

[13] Yisen Wang, Difan Zou, Jinfeng Yi, James Bailey, Xingjun Ma, and Quanquan Gu. Improving adversarial robustness requires revisiting misclassified examples. In *International conference on learning representations*, 2019. 2

[14] Zeming Wei, Yifei Wang, Yiwen Guo, and Yisen Wang. Cfa: Class-wise calibrated fair adversarial training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8193–8201, 2023. 4, 5, 6

[15] Han Xu, Xiaorui Liu, Yaxin Li, Anil Jain, and Jiliang Tang. To be robust or to be fair: Towards fairness in adversarial training. In *International Conference on Machine Learning*, pages 11492–11501. PMLR, 2021. 3, 6

[16] Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric Xing, Laurent El Ghaoui, and Michael Jordan. Theoretically principled trade-off between robustness and accuracy. In *International conference on machine learning*, pages 7472–7482. PMLR, 2019. 2, 3, 4, 6



(a) Original AT model trained with Uniform class weights

(b) Fine-tuned FAAL model trained with CDAW weights

Figure 5. From (a) to (b): Fine-tuning the WRN-34-10 model with FAAL on CIFAR-10 dataset

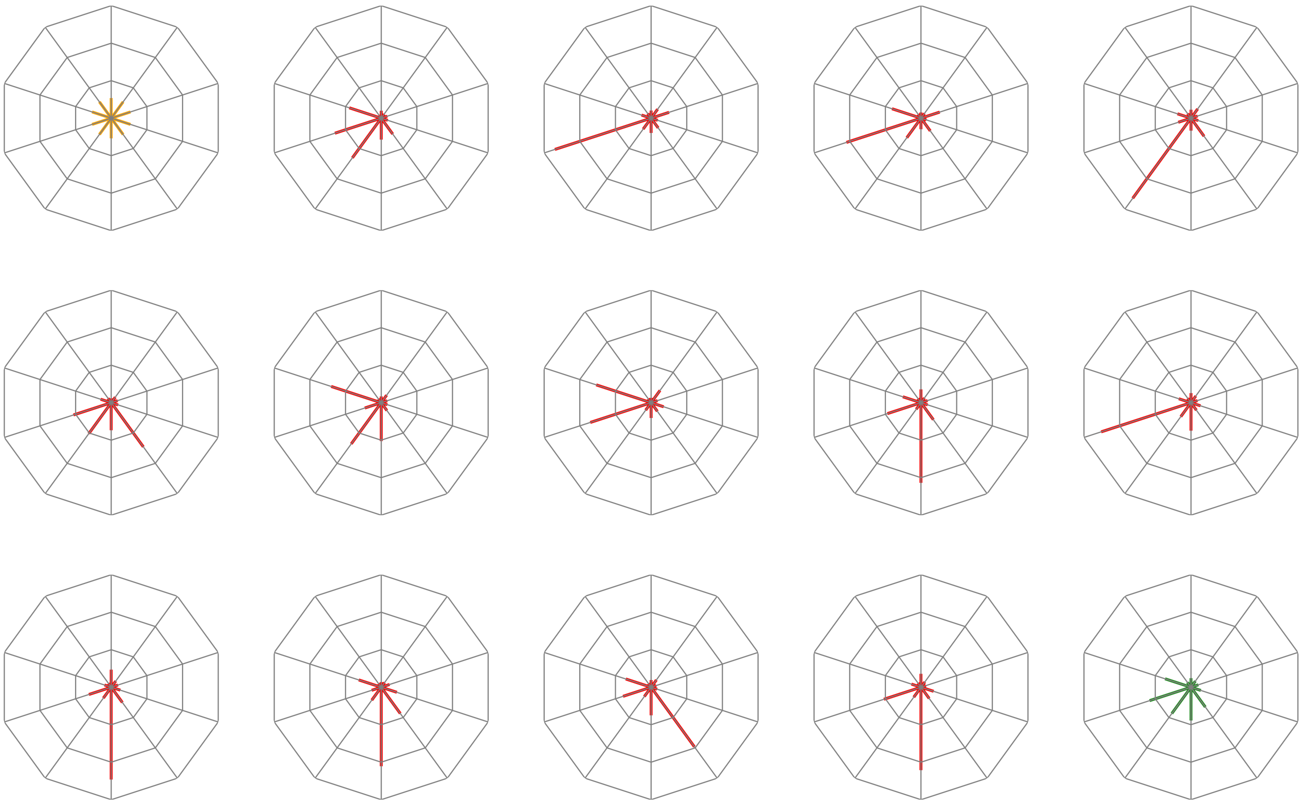


Figure 6. The weights learned by FAAL during training on CIFAR-10 dataset: the first image represents the uniform distribution (orange); the last one represents the averaged weights learned by FAAL during training (green); others represent weights learned for some specific batches (red)

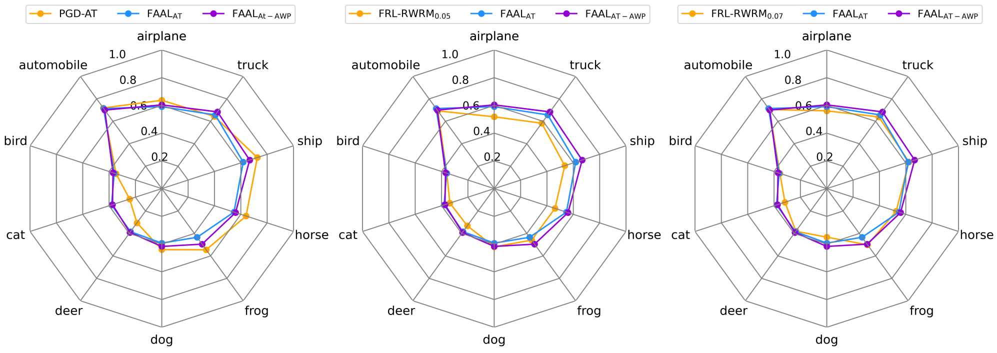


Figure 7. Class-wise robust Accuracy against AutoAttack after fine-tuning the PGD adversarially trained WRN model on CIFAR-10

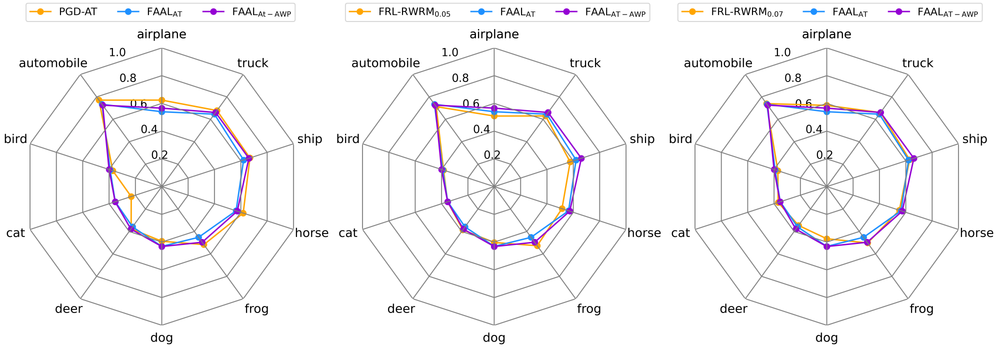


Figure 8. Class-wise robust Accuracy against AutoAttack after fine-tuning the TRADES adversarially trained WRN model on CIFAR-10

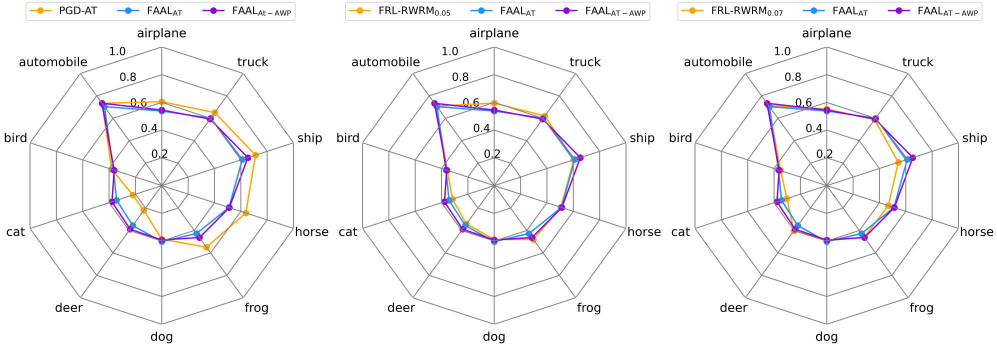


Figure 9. Class-wise robust Accuracy against AutoAttack after fine-tuning the MART adversarially trained WRN model on CIFAR-10

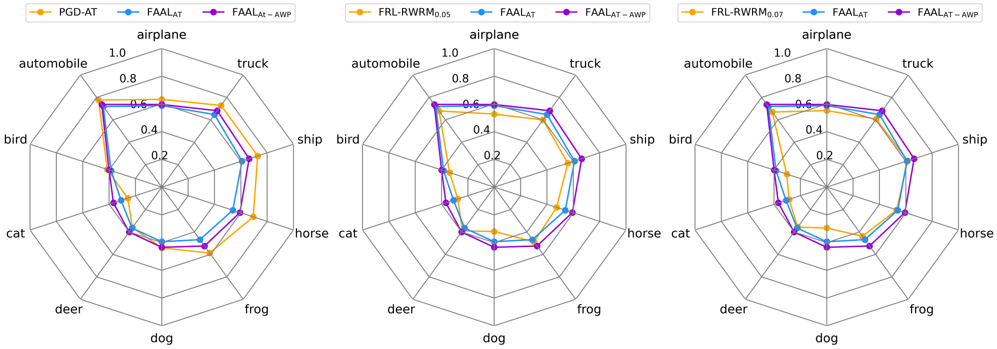


Figure 10. Class-wise robust Accuracy against AutoAttack after fine-tuning the AWP adversarially trained WRN model on CIFAR-10