Sample-wise Adaptive Weighting for Transfer Consistency in Adversarial Distillation

Anonymous authors
Paper under double-blind review

Abstract

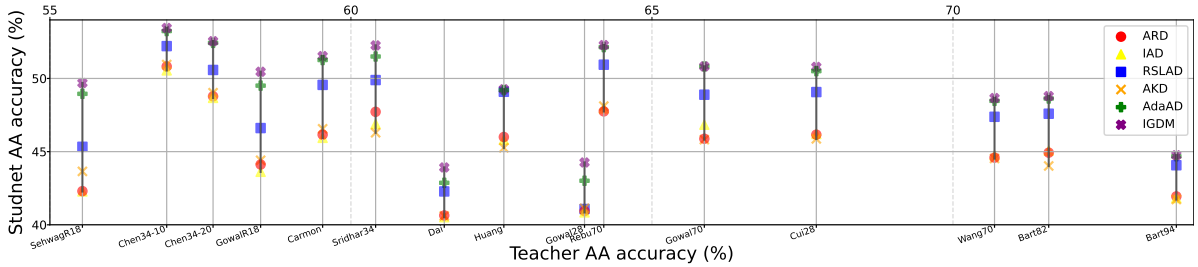
Adversarial distillation in the standard min–max adversarial training framework aims to transfer adversarial robustness from a large, robust teacher network to a compact student. However, existing work often neglects to incorporate state-of-the-art robust teachers. Through extensive analysis, we find that stronger teachers do not necessarily yield more robust students–a phenomenon known as robust saturation. While typically attributed to capacity gaps, we show that such explanations are incomplete. Instead, we identify adversarial transferability–the fraction of student-crafted adversarial examples that remain effective against the teacher–as a key factor in successful robustness transfer. Based on this insight, we propose Sample-wise Adaptive Adversarial Distillation (SAAD), which reweights training examples by their measured transferability without incurring additional computational cost. Experiments on CIFAR-10, CIFAR-100, and Tiny-ImageNet show that SAAD consistently improves AutoAttack robustness over prior methods.

1 Introduction

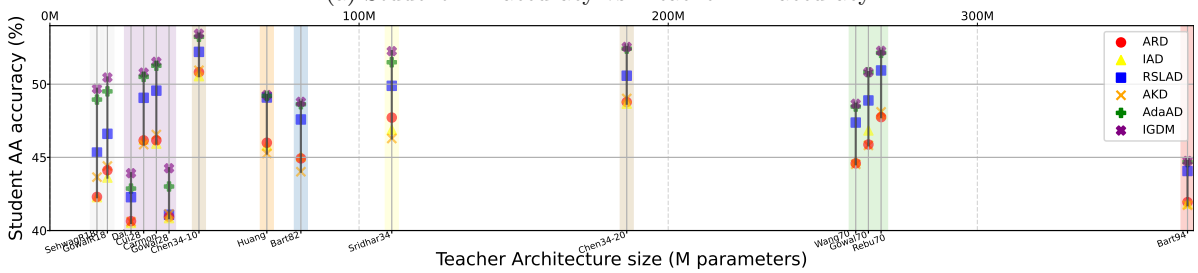
Deep neural networks have achieved remarkable success across diverse domains, yet they remain highly susceptible to adversarial perturbations (Goodfellow et al., 2014; Carlini & Wagner, 2017; Madry et al., 2017; Athalye et al., 2018), posing significant risks in safety-critical applications (Grigorescu et al., 2020; Ma et al., 2021; Wang et al., 2023a). In response, a variety of defense strategies have been proposed (Das et al., 2017; Cohen et al., 2019; Carmon et al., 2019; Xie et al., 2019; Zhang et al., 2022; Jin et al., 2023), among which adversarial training (AT) (Goodfellow et al., 2014; Madry et al., 2017) has emerged as a leading method. Despite its effectiveness, AT typically requires large-scale models, resulting in a substantial performance gap for lightweight architectures commonly deployed in resource-constrained settings (Madry et al., 2017). To bridge this gap, AT-based *adversarial distillation (AD)* methods (Goldblum et al., 2020; Zhu et al., 2021; Zi et al., 2021; Maroto et al., 2022; Huang et al., 2023; Jung et al., 2024; Park & Min, 2024; Lee et al., 2025) have been proposed as a promising approach for transferring the robustness of large teacher models to compact student models.

Despite the promise of AD, many existing studies often neglect to incorporate state-of-the-art robust teachers from standardized benchmarks such as RobustBench (Croce et al., 2021). A natural expectation is that a more robust teacher would yield a correspondingly robust student. However, as illustrated in Figure 1a, our experiments reveal that employing stronger teachers can in fact degrade student robustness, contradicting conventional intuition. This surprising result raises a fundamental question: what factors account for the variability in AD performance across teacher models, and why does greater teacher robustness not necessarily lead to more effective robustness transfer? In this work, we address this question by examining the role of *adversarial transferability* in robust knowledge distillation.

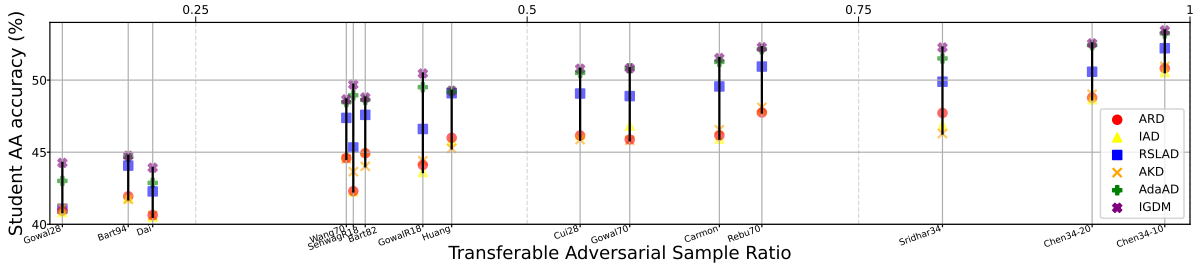
Previous studies have attributed the failure of robustness transfer in AD to the *robust saturation effect* (Zi et al., 2021), which posits that beyond a certain capacity threshold, further increases in teacher robustness or model size yield diminishing returns for the student. However, as shown in Figure 1b, even when teachers are ordered by architectural size within the same model family (e.g., WRN-28-10 in purple and WRN-70-16



(a) Student AA accuracy vs. Teacher AA accuracy.



(b) Student AA accuracy vs. Teacher architecture size. Colored regions represent the same architectures.



(c) Student AA accuracy vs. Fraction of adversarial examples that transfer from student to teacher (TAS).

Figure 1: Adversarial distillation results on CIFAR-10 with a ResNet-18 student. Detailed teacher information and full experimental results are provided in Section A.1.

in green), student robustness varies significantly. This suggests that capacity gap alone cannot fully explain the observed discrepancies.

To better understand this limitation, we analyze how the teacher's output confidence on student-crafted attacks influences the student's adversarial variance and overfitting through distillation. We find that highly confident (i.e., low-entropy) outputs from robust teachers exacerbate student variance under attack, resulting in unstable training. Our analysis further reveals that this instability arises from a lack of *transferable adversarial samples (TAS)*–student-generated adversarial inputs that remain effective against the teacher–whose abundance strongly correlates with successful robustness transfer, as demonstrated in Figure 1c.

Motivated by this insight, we propose *Sample-wise Adaptive Adversarial Distillation (SAAD)*, a novel approach that emphasizes samples with high adversarial transferability to improve robustness transfer. SAAD assigns lower weights to non-transferable samples, effectively mitigating their high-variance effects and improving the student model's robustness. We further introduce a clean distillation term weighted by inverse transferability, boosting clean accuracy while preserving robustness. Extensive experiments demonstrate that our method consistently improves student robustness in cases where superior teacher models did not translate into enhanced robustness under existing methods. Our contributions are as follows:

- We identify adversarial transferability as a key factor for effective adversarial distillation, explaining why stronger teachers can fail to improve student robustness.

- We propose *Sample-wise Adaptive Adversarial Distillation (SAAD)*, which selectively emphasizes transferable samples to mitigate high-variance effects and improve robustness.
- We show that our method consistently improves both robustness and clean accuracy across diverse settings, outperforming prior adversarial distillation approaches.

2 Related Works

Adversarial Attacks and Transferability. Based on the adversary's level of access to the victim model, adversarial attacks are distinguished as either white-box or black-box. In the white-box paradigm, the adversary has full access to the model parameters and gradients, which enables gradient-based attacks such as FGSM (Goodfellow et al., 2014), PGD (Madry et al., 2017), and stronger optimization-based methods (Carlini & Wagner, 2017; Croce & Hein, 2020; Li et al., 2024). In contrast, black-box attacks operate with limited knowledge of the target and are typically either query-based or transfer-based. Query-based attacks directly probe the model, including score-based methods (Uesato et al., 2018; Andriushchenko et al., 2020) and decision-based boundary attacks (Chen & Gu, 2020; Chen et al., 2020; 2021b). Transfer-based attacks rely on the adversarial transferability phenomenon, where adversarial examples created for one model succeed in misleading another (Szegedy et al., 2013; Papernot et al., 2016; 2017; Tramèr et al., 2017). In this context, the adversary typically constructs adversarial examples on surrogate models and leverages their transferability as an attack mechanism (Liu et al., 2016; Mahmood et al., 2021). Diverging from this conventional paradigm, our work re-purposes adversarial transferability as a diagnostic tool. We leverage it not to attack models, but to evaluate the efficacy of different teacher models within the adversarial distillation framework.

Adversarial Training. In response to adversarial attacks, adversarial training (AT) has emerged as one of the most effective defenses. In its standard form, known as PGD-AT (Madry et al., 2017), the model parameters $\boldsymbol{\theta}$ are optimized via a min-max formulation:

$$\arg\min_{\boldsymbol{\theta}} \ \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\Big[\mathrm{CE}(\mathbf{y}, f_{\boldsymbol{\theta}}(\mathbf{x}+\boldsymbol{\delta}))\Big], \quad \text{where} \quad \boldsymbol{\delta} = \arg\max_{\boldsymbol{\delta}\in\Delta} \mathrm{CE}(\mathbf{y}, f_{\boldsymbol{\theta}}(\mathbf{x}+\boldsymbol{\delta})) \tag{1}$$

Here, the inner maximization generates adversarial perturbations that maximize the cross-entropy loss, while the outer minimization trains the model to minimize this loss under the worst-case perturbation $\boldsymbol{\delta}$. To address trade-offs between robustness and accuracy, TRADES (Zhang et al., 2019) reformulates adversarial training by decoupling the loss into a clean classification term and a robustness regularization via KL divergence:

$$\arg\min_{\boldsymbol{\theta}} \ \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\mathrm{CE}(\mathbf{y}, f_{\boldsymbol{\theta}}(\mathbf{x})) + \lambda \cdot \max_{\boldsymbol{\delta}\in\Delta} \mathrm{KL}(f_{\boldsymbol{\theta}}(\mathbf{x}) \| f_{\boldsymbol{\theta}}(\mathbf{x}+\boldsymbol{\delta}))\right] \tag{2}$$

This formulation explicitly balances natural accuracy and robustness through the hyperparameter λ. MART (Wang et al., 2020) integrates per-sample weighting based on prediction confidence. Its objective can be described as:

$$\arg\min_{\boldsymbol{\theta}} \ \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\Big[\mathrm{CE}(\mathbf{y}, f_{\boldsymbol{\theta}}(\mathbf{x}+\boldsymbol{\delta})) + (1-w_y)\cdot \mathrm{KL}(f_{\boldsymbol{\theta}}(\mathbf{x}) \| f_{\boldsymbol{\theta}}(\mathbf{x}+\boldsymbol{\delta}))\Big] \tag{3}$$

where inner maximization to compute $\boldsymbol{\delta}$ is equal to the PGD-AT and the weight w_y is computed from the confidence of the true class prediction. This adaptively emphasizes hard examples and misclassified inputs during training. These variants have inspired a rich line of adversarial training research (Qin et al., 2019; Wu et al., 2020; Bai et al., 2021; Jin et al., 2022; Tack et al., 2022; Jin et al., 2023; Wei et al., 2023).

Adversarial Distillation. Adversarial distillation (AD) aims to transfer the robustness of a large, adversarially trained teacher model into a more compact student model. The dominant paradigm, and the focus of our work, is AT-based AD, which leverages teacher signals under a min-max adversarial training framework (Goldblum et al., 2020; Zhu et al., 2021; Zi et al., 2021; Maroto et al., 2022; Huang et al., 2023; Kuang et al., 2023; Lee et al., 2025). Unlike standard knowledge distillation (Hinton et al., 2015), which aligns clean predictions, this approach explicitly considers adversarially perturbed inputs during training to preserve robustness in the student.

Adversarial Robustness Distillation (ARD) (Goldblum et al., 2020) initiates this line of work by incorporating adversarial examples into the distillation process, showing that robust teachers can effectively guide student models when both are trained under adversarial settings. RSLAD (Zi et al., 2021) builds on this by integrating teacher outputs directly into the generation of adversarial examples, encouraging smoother teacher logits and more stable student learning, and further reports a *robust saturation effect*: a student's robustness increases with teacher strength only up to a moderately larger teacher and then declines as teacher capacity outpaces the student. Introspective Adversarial Distillation (IAD) (Zhu et al., 2021) proposes a confidence-based modulation of the teacher signal, weighting the distillation loss by the estimated reliability of the teacher under adversarial inputs. AdaAD (Huang et al., 2023) introduces a more sophisticated approach where the teacher is actively involved in the inner maximization step, generating adversarial examples that are optimized with respect to both the student and the teacher. Most recently, IGDM (Lee et al., 2025) indirectly distills the gradient information of the teacher model to enhance the robustness further. Table 11 summarizes the inner maximization and outer minimization objectives used by representative AD methods.

While the aforementioned methods distill from a single robust teacher, another line of research employs multiple teachers to address the trade-off between clean accuracy and robustness (Zhao et al., 2022; Deng et al., 2024). AD has also been applied to broader robustness contexts such as class imbalance (Yue et al., 2023; Zhao et al., 2024; Cho et al., 2025b), incremental learning (Cho et al., 2025a), and self-distillation (Jung et al., 2024). Another line of research transfers robustness using non-AT-based methods. These approaches often leverage gradient or feature matching on clean inputs (Shafahi et al., 2019; Chan et al., 2020; Awais et al., 2021; Chen et al., 2021a; Muhammad et al., 2021; Shao et al., 2021; Vaishnavi et al., 2022). As these methods are designed to replace the expensive PGD inner-loop, they optimize for a different trade-off and inherently sacrifice robustness. They are therefore orthogonal to our work, which focuses on diagnosing and solving the robust saturation phenomenon within the AT-based paradigm.

3 Robust Teacher Failures: Entropy, Variance, Transferability

In prior AD works, the teacher models are typically either large networks trained with methods such as TRADES (Zhang et al., 2019) or publicly available robust models widely adopted by early AD research (Zi et al., 2021; Huang et al., 2023; Lee et al., 2025). Although a new generation of SOTA robust models are now readily available on RobustBench (Croce et al., 2021), many recent AD studies have not focused on incorporating these specific models. One might naturally expect that leveraging stronger teachers would yield improved student robustness. However, our experiments across a diverse set of teachers in Figure 1 reveal that existing AD methods are highly susceptible to teacher choice, with even the most robust teachers leading to poor student robustness.

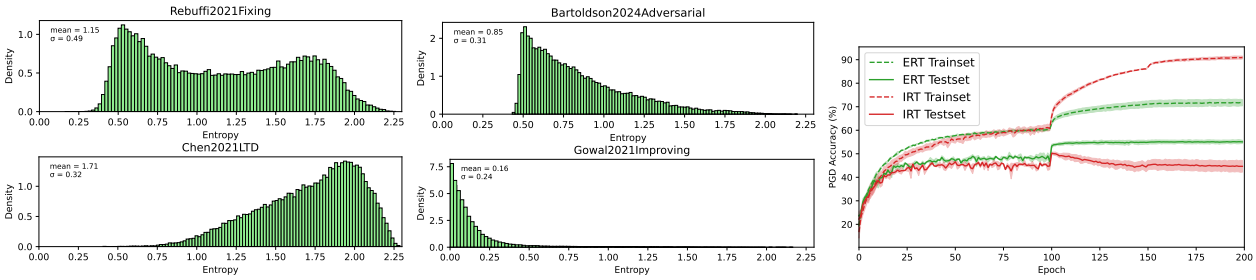
A simple explanation often given for this phenomenon is the so-called robust saturation effect (Zi et al., 2021), which attributes the diminishing gain of adversarial distillation to the capacity gap between the teacher and student models. However, as shown in Figure 1b, we find no consistent trend even when distillation outcomes are sorted by teacher architecture, indicating that the capacity gap alone cannot fully explain the failure modes. Accordingly, we introduce a new framework by dividing robust teachers into two categories: *Effective Robust Teachers* (ERTs) and *Ineffective Robust Teachers* (IRTs), defined by whether students distilled from them via recent AD methods, on average, outperform or underperform AT baselines (TRADES) in robust accuracy. To systematically compare these groups, we select representative teachers as summarized in Table 1, with additional details provided in Section A.1. For interpretability in subsequent analyses, we adopt RSLAD (Zi et al., 2021) as the baseline AD method and fix the student architecture to ResNet-18 trained on CIFAR-10.

3.1 Characterizing IRTs: Overconfidence and Overfitting

We observe two key distinctions between IRTs and ERTs. First, IRTs tend to produce lower-entropy outputs than ERTs, particularly on adversarial inputs generated by the student model. Figure 2a and Figure 2b show the density histograms of teacher-logit entropies evaluated on student-generated PGD-20 adversarial inputs. We find that IRTs yield highly confident predictions with significantly lower entropy, while ERTs maintain a broader entropy distribution, suggesting a more calibrated uncertainty. Importantly, a high output entropy

Table 1: Comparison of distillation outcomes using different teacher models categorized as Effective Robust Teachers (ERTs) and Ineffective Robust Teachers (IRTs). AA denotes AutoAttack accuracy (%) of the teacher (left) and the student (right); RO measures robust overfitting, computed as the gap between the student's best and last PGD-20 accuracy on the test set. AVar denotes the adversarial variance, and TAS refers to the ratio of transferable samples in the training dataset.

Teacher Info				Distillation Results			
Group	RobustBench name	Architecture	AA	AA	RO	AVar	TAS
ERT	`Rebuffi2021Fixing`	WRN-70-16	64.20	50.94	0.20	0.0267	0.677
	`Chen2021LTD`	WRN-34-10	56.94	52.21	0.15	0.0059	0.981
IRT	`Bartoldson2024Adversarial`	WRN-94-16	73.71	44.07	5.44	0.0834	0.199
	`Gowal2021Improving`	WRN-28-10	63.38	41.08	7.01	0.3058	0.149



(a) Entropy distributions on ERTs (b) Entropy distributions on IRTs (c) Robust accuracy dynamics

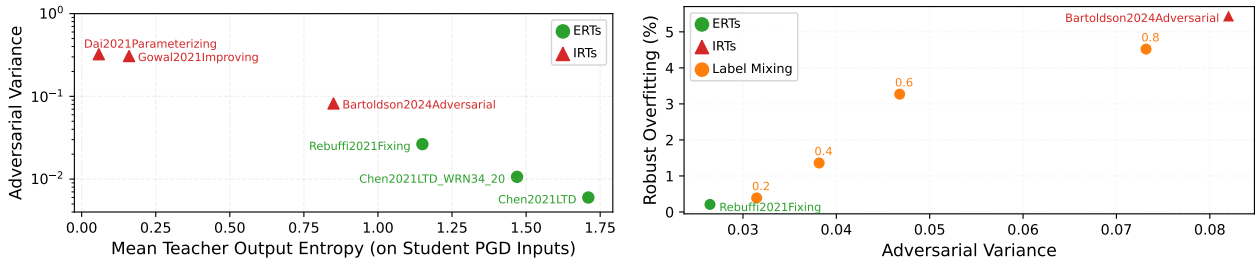
Figure 2: **(a)** Density histograms of teacher-logit entropies on student-generated PGD-20 adversarial training inputs for two ERTs. **(b)** Same, but for IRTs. **(c)** PGD-20 robust accuracy on training and test sets across epochs for students distilled from individual teachers within the ERT and IRT groups. Solid lines indicate group-wise averages, and shaded regions represent standard deviations across teachers in each group.

does not necessarily imply non-robustness of the teacher model. Despite exhibiting higher entropy, ERTs can correctly classify adversarial examples crafted on student models. This suggests that ERTs maintain a level of uncertainty around adversarial inputs without fully collapsing into overconfident predictions, whereas IRTs often yield overconfident outputs aligned closely with the true label, even under attack.

Second, students distilled from IRTs exhibit pronounced robust overfitting, whereas those distilled from ERTs maintain stable generalization. This effect is visualized in Figure 2c, where the PGD-20 robust accuracy on the training and test sets for IRTs diverges significantly after the learning rate decay—a characteristic pattern of robust overfitting driven by the disruption of the min–max balance caused by the decay (Wang et al., 2023b). To quantify this, we report robust overfitting (RO) as the gap between the student's best and last PGD-20 accuracy on the test set in Table 1; IRT-distilled students exhibit large RO values, while ERT students show minimal overfitting. These results demonstrate a clear empirical link between overconfident teacher outputs and robust overfitting in the student. While the mechanism behind this link remains unclear, the consistency of these patterns across multiple IRTs suggests a deeper connection. In the next section, we formally investigate this connection by analyzing adversarial variance as a potential explanatory factor.

3.2 Adversarial Variance Analysis on Adversarial Distillation

To investigate how overconfident soft labels from robust teachers induce robust overfitting in students, we extend the classical bias–variance decomposition of expected risk to the adversarial distillation setting by introducing adversarial variance. This formulation unifies earlier decompositions from adversarial training (Yu et al., 2021) and knowledge distillation (Zhou et al., 2021), and helps account for teacher-dependent variation in adversarial distillation. We note that $f_{\hat{\boldsymbol{\theta}}(\mathcal{D})} : \mathcal{X} \to \mathbb{R}^C$ represents the student model adversarially trained on dataset $\mathcal{D}$ under soft-label supervision from a fixed teacher. While the teacher remains unchanged



(a) Entropy vs. Adversarial Variance

(b) Adversarial Variance vs. Robust Overfitting

Figure 3: (a) Teachers with lower entropy on student-generated PGD inputs induce higher adversarial variance in the student. (b) Higher adversarial variance is associated with increased robust overfitting. Orange points show experiments where true labels are mixed into `Rebuffi2021Fixing` outputs. The numeric labels indicate the proportion of true label supervision.

during training, its output distribution governs the soft labels used in the distillation process, thus indirectly shaping $\hat{\boldsymbol{\theta}}$ and the student's final behavior. For a test sample $\mathbf{x}$ with ground truth label $\mathbf{y} = t(\mathbf{x}) \in \mathbb{R}^C$, we consider the worst-case perturbation

$$\boldsymbol{\delta}(\mathbf{x}, \mathbf{y}, \mathcal{D}) \in \arg\max_{\boldsymbol{\delta} \in \Delta} L_{\max}\big(f_{\hat{\boldsymbol{\theta}}(\mathcal{D})}(\mathbf{x} + \boldsymbol{\delta}), \mathbf{y}\big), \tag{4}$$

and define

$$\hat{\mathbf{y}} := f_{\hat{\boldsymbol{\theta}}(\mathcal{D})}(\mathbf{x} + \boldsymbol{\delta}(\mathbf{x}, \mathbf{y}, \mathcal{D})), \qquad \bar{\mathbf{y}} := \frac{1}{Z} \exp\left(\mathbb{E}_{\mathcal{D}}[\log \hat{\mathbf{y}}]\right), \tag{5}$$

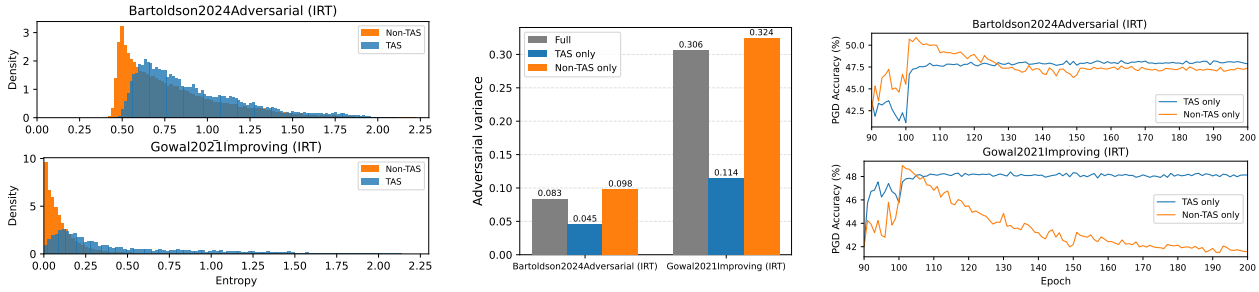
where Z is the normalization constant to ensure $\bar{\mathbf{y}}$ lies in the probability simplex. Then, the expected adversarial cross-entropy risk admits the following decomposition:

$$\text{ARisk} = \mathbb{E}_{\mathbf{x},\mathcal{D}}\left[\text{CE}(\mathbf{y}, \hat{\mathbf{y}})\right] = \underbrace{\mathbb{E}_{\mathbf{x}}\left[-\mathbf{y} \log \mathbf{y}\right]}_{\text{Intrinsic Noise}} + \underbrace{\mathbb{E}_{\mathbf{x}}\left[\mathbf{y} \log \frac{\mathbf{y}}{\bar{\mathbf{y}}}\right]}_{\text{Adversarial Bias}} + \underbrace{\mathbb{E}_{\mathbf{x},\mathcal{D}}\left[\text{KL}(\bar{\mathbf{y}} \,\|\, \hat{\mathbf{y}})\right]}_{\text{Adversarial Variance}}, \tag{6}$$

where $\text{CE}(\mathbf{p}, \mathbf{q}) = -\sum_i p_i \log q_i$ is the cross-entropy loss. Detailed explanation and an algorithm for estimating the adversarial bias and variance are given in the Section A.2. This decomposition enables us to empirically analyze how the adversarial variance of student models in adversarial distillation varies depending on different teacher models.

Overconfident Soft Labels Induce High Adversarial Variance. We observe that adversarial variance increases when the teacher produces low-entropy predictions on student-generated PGD inputs. As shown in Figure 3a, robust teachers such as `Gowal2021Improving` and `Bartoldson2024Adversarial`—despite their high standalone robustness—exhibit low entropy and correspondingly high student variance, whereas higher-entropy teachers such as `Rebuffi2021Fixing` and `Chen2021LTD` yield more stable behavior. This suggests that overconfident teacher outputs undermine the regularizing effect of soft labels, amplifying variance during adversarial training. To probe this effect more directly, we conduct an interpolation experiment using `Rebuffi2021Fixing`, gradually injecting the ground-truth label into the teacher's logits. As illustrated in Figure 3b, adversarial variance increases monotonically with the interpolation coefficient, indicating that growing confidence in soft labels directly induces instability in the student's response.

High Adversarial Variance Causes Robust Overfitting. Analogous to classical statistical learning theory, we find that adversarial variance, measured over perturbed inputs, serves as a strong indicator of robust overfitting. As shown in Figure 3b, we observe a clear correlation between the magnitude of adversarial variance and the degree of overfitting to adversarial training data. While AD is generally expected to reduce variance and thereby mitigate overfitting, we find that this effect depends critically on the teacher's output distribution. In earlier AD studies, robust overfitting received limited attention, likely because ERTs inherently produce soft labels with sufficient uncertainty, resulting in low adversarial variance. However, as more powerful yet sharper teachers are adopted, understanding and controlling adversarial variance becomes essential for ensuring stability in robust distillation.



(a) TAS and Non-TAS group entropy on IRTs

(b) Variance on TAS and Non-TAS Subsets on IRTs

(c) Test Robust Accuracy Trained on TAS and Non-TAS Subsets on IRTs

Figure 4: **(a)** Density histograms of teacher-logit entropies on student-generated PGD-20 adversarial training input, separated into TAS and Non-TAS groups. **(b)** Bar chart of adversarial variance when training only on the TAS subset versus the Non-TAS subset (for each teacher). **(c)** PGD-20 robust accuracy on train (dashed) and test (solid) over epochs.

Overconfident soft labels from IRTs induce high adversarial variance in the student model, leading to robust overfitting. While this explains the mechanism of failure, it raises a deeper question: why do IRTs fail to provide meaningful supervision on student-generated adversarial examples? In the following section, we show that this limitation arises from a lack of transferability between the adversarial behaviors of the student and teacher models.

3.3 Sample-Level Transferability in Adversarial Distillation

We identify the lack of *transferable adversarial samples* (TAS) as the primary cause of failure in AD under IRT supervision. Specifically, when a student-crafted perturbation fails to induce a comparable adversarial shift in the teacher's prediction, the teacher's supervision signal becomes misaligned, thereby degrading the efficacy of robustness transfer. To formalize this notion, we conduct a sample-level analysis of behavioral alignment between student and teacher models under adversarial perturbations. We define a transferable adversarial sample as an input $\mathbf{x}$ for which the adversarial perturbation $\boldsymbol{\delta}_S$, crafted by the student, induces a response from the teacher that aligns more closely with its own adversarial response than with its original (clean) prediction. Formally, this condition is satisfied if:

$$\mathrm{KL}\big(f_T(\mathbf{x}+\boldsymbol{\delta}_S)\,\|\,f_T(\mathbf{x})\big) \geq \mathrm{KL}\big(f_T(\mathbf{x}+\boldsymbol{\delta}_S)\,\|\,f_T(\mathbf{x}+\boldsymbol{\delta}_T)\big), \tag{7}$$

indicating that the student's adversarial perturbation is aligned well with the teacher's $\boldsymbol{\delta}_T$.

In Figure 4a, we compare the output entropy of IRT teachers on student-generated adversarial inputs, separating samples into TAS vs. non-TAS categories. We observe that the TAS group maintains higher entropy, while the non-TAS group is concentrated in the low-entropy regime, indicating overconfident predictions. Furthermore, Figure 4b presents the adversarial variance observed when training is continued separately on each sample group following 90 epochs of warm-up on the full dataset. Non-TAS group results in significantly higher adversarial variance, suggesting that the supervision they provide is unstable due to misaligned adversarial behavior. Further, Figure 4c shows that this instability correlates with degraded generalization: models trained on the non-TAS group exhibit pronounced robust overfitting, whereas training on the TAS group preserves robust generalization. Taken together, these findings indicate that non-TAS, characterized by low entropy and high adversarial variance, induce unstable and misaligned supervision, ultimately leading to robust overfitting. Consequently, a high proportion of non-transferable examples impairs the efficacy of adversarial distillation, indicating the importance of TAS for successful robustness transfer.

As shown in Table 1, the proportion of TAS is substantially lower for IRTs compared to ERTs, further reinforcing the connection between transferability and successful robustness transfer. Moreover, Figure 1c demonstrates a positive correlation between the TAS ratio and the student model's robustness under AutoAttack, underscoring the predictive value of this metric. These observations suggest that the scarcity of

TAS is not merely a byproduct of poor distillation but a central cause of IRTs' inability to provide effective supervision. Thus, sample-level transferability emerges as a critical factor in explaining and potentially overcoming the limitations of adversarial distillation.

4 Main Method: Sample-wise Adaptive Adversarial Distillation

Our analysis reveals that the difference in distillation effectiveness between ERTs and IRTs arises from the entropy distribution of teacher logits and the resulting adversarial variance. ERTs produce higher-entropy outputs on student-generated adversarial inputs, which lead to lower adversarial variance and better generalization. In contrast, IRTs yield overconfident, low-entropy predictions that induce high adversarial variance and robust overfitting. This problem is further pronounced by the large portion of non-TAS samples under IRTs, where adversarial perturbations fail to meaningfully alter the teacher's outputs. These non-TAS samples dominate training, thereby exacerbating variance-driven overfitting.

While existing AD methods can be effective when the teacher provides a sufficient number of transferable samples, they apply the distillation objective uniformly across all data points, failing to distinguish between transferable and non-transferable samples. A simple alternative is to train only on transferable samples. However, as shown in Table 2, this approach yields inferior overall performance due to the reduced sample count, despite improved robustness over training only on non-transferable samples. These findings suggest that entirely discarding non-transferable samples is suboptimal, especially as adversarial training demands intensive data to achieve robustness (Schmidt et al., 2018).

Table 2: Study of the impact on adversarial distillation for transferable adversarial sample.

Setting	# of Data	Clean	AA
Full Data	50000	84.28	44.42
Excluding TAS	45161	83.93	43.05
Only on TAS	4839	80.70	44.00

Motivated by these insights, we propose Sample-wise Adaptive Adversarial Distillation (SAAD), which assigns higher weights to transferable adversarial samples during distillation. The weighting mechanism is derived from the transferable adversarial sample criterion defined in equation 7. A key challenge, however, is that computing the teacher-side perturbation $\boldsymbol{\delta}_T$, which is not required in standard adversarial distillation, incurs additional computational overhead, particularly for large teacher models. To address this, we note that from the student's perspective, the teacher outputs $f_T(\mathbf{x})$ and $f_T(\mathbf{x}+\boldsymbol{\delta}_T)$ remain fixed, while only the student-induced perturbation $\boldsymbol{\delta}_S$ varies. We further leverage the empirical observation that the teacher's output distribution on its own adversarial input, $f_T(\mathbf{x}+\boldsymbol{\delta}_T)$, typically exhibits higher entropy than on the clean input $f_T(\mathbf{x})$. According to equation 7, transferable samples are those for which the student's perturbation $\boldsymbol{\delta}_S$ sufficiently approximates the teacher's own adversarial behavior, thereby inducing a comparable increase in entropy in $f_T(\mathbf{x}+\boldsymbol{\delta}_S)$.

Based on this insight, SAAD assigns sample-wise weights proportional to the entropy of $f_T(\mathbf{x}+\boldsymbol{\delta}_S)$, effectively prioritizing transferable adversarial examples without incurring additional computational cost. A more detailed justification linking the TAS criterion to the entropy of $f_T(\mathbf{x}+\boldsymbol{\delta}_S)$ is provided in Section A.3.1. The resulting loss function is defined as:

$$L_{\text{SAAD}} = \frac{1}{N}\sum_{i=1}^{N} w_i \cdot L_{\text{AD}}(f_S, f_T, \mathbf{x}_i, \boldsymbol{\delta}_{S,i}), \quad w_i := H\left(f_T(\mathbf{x}_i+\boldsymbol{\delta}_{S,i})\right), \tag{8}$$

where L_{AD} denotes an existing AD method. In our implementation, we adopt IGDM (Lee et al., 2025) as the base method; additional details are provided in Section A.3.2.

By weighting adversarial distillation according to the entropy of the teacher's perturbed outputs, non-transferable samples receive negligible weight and are effectively suppressed. Although such samples exhibit low-entropy teacher logits, indicating limited utility for robustness, they still contain confident supervision aligned with the true label. To preserve this clean signal, we introduce a complementary clean distillation loss by assigning inverse weights $1-\tilde{w}_i$, where $\tilde{w}_i = w_i/\log C$ denotes the entropy normalized by the maximum

Table 3: Performance (%) of the teacher models. Teacher name correspond to ID in RobustBench.

Dataset	Teacher name	Architecture	Clean	AA
CIFAR-10	`Bartoldson2024Adversarial`	WRN-94-16	93.68	73.71
	`Gowal2021Improving`	WRN-28-10	87.50	63.38
CIFAR-100	`Wang2023Better`	WRN-70-16	75.22	42.66
Tiny-ImageNet	`Wang2023Better`	WRN-28-10	65.19	31.30

entropy for C classes:

$$L_{\text{SAAD-C}} = L_{\text{SAAD}} + \frac{1}{N}\sum_{i=1}^{N} \beta \cdot (1 - \tilde{w}_i) \cdot \text{KL}\left(f_T(\mathbf{x}_i) \,\|\, f_S(\mathbf{x}_i)\right), \tag{9}$$

for the clean distillation weight β. The second term thus reintroduces non-transferable samples into the clean distillation process, allowing the student to learn clean knowledge from the teacher. Such a clean distillation term has appeared in prior AD methods (Zi et al., 2021; Huang et al., 2023; Lee et al., 2025), but those works set their coefficient to zero in practice, as robustness losses outweighed the clean accuracy gains. In contrast, by restricting clean distillation to non-transferable samples, we achieve substantial improvements in clean accuracy with only marginal robustness degradation.

5 Experimental Results

5.1 Experiment Setup

Adversarial Distillation Setting. We conduct experiments on CIFAR-10, CIFAR-100 (Krizhevsky et al., 2009), and Tiny-ImageNet (Le & Yang, 2015), using standard data augmentations (random crop and horizontal flip). We compare baseline adversarial training methods—PGD-AT (Madry et al., 2017) and TRADES (Zhang et al., 2019)—with six adversarial distillation approaches: ARD (Goldblum et al., 2020), IAD (Zhu et al., 2021), RSLAD (Zi et al., 2021), AKD (Maroto et al., 2022), AdaAD (Huang et al., 2023), and IGDM (Lee et al., 2025). Further details are provided in Section B.

Teacher and Student Models. We employ robust teacher models summarized in Table 3, including state-of-the-art entries from RobustBench (Croce et al., 2021)[1] for CIFAR-10 and CIFAR-100. To broaden our evaluation, we also include a variant with a different architecture for CIFAR-10. All selected teachers fall under the IRT category defined in Section 3, meaning that despite strong standalone robustness, they fail to effectively transfer robustness through existing adversarial distillation. As student architectures, we use ResNet-18 (He et al., 2016a) and MobileNetV2 (Sandler et al., 2018) for CIFAR datasets and PreActResNet-18 (He et al., 2016b) for Tiny-ImageNet.

Evaluation Setting. We evaluate each model using five metrics: Clean, FGSM, PGD, C&W, and AutoAttack (AA) accuracy. Clean accuracy is measured on the original test set without perturbation. FGSM and PGD accuracies are obtained using adversarial examples generated by the fast gradient sign method (Goodfellow et al., 2014) and a 20-step projected gradient descent attack (Madry et al., 2017), respectively. C&W accuracy is measured under the optimization-based attack proposed in (Carlini & Wagner, 2017), while AA reports worst-case accuracy under the AutoAttack ensemble (Croce & Hein, 2020). All adversarial attacks are conducted under an l_∞-norm constraint of 8/255.

5.2 Adversarial Distillation Results

Table 4 and Table 6 summarize adversarial robustness across datasets and methods. SAAD consistently achieves the best AutoAttack accuracy across all settings, outperforming conventional adversarial training

[1]Accessed Feb 02, 2026: `https://robustbench.github.io/`

Table 4: Adversarial distillation results using two teacher and two student models on CIFAR-10. Clean, FGSM, PGD, C&W, and AA columns report accuracy (%) under each evaluation setting. Results are averaged over three random seeds.

Model	Method	Bartoldson2024Adversarial					Gowal2021Improving				
		Clean	FGSM	PGD	C&W	AA	Clean	FGSM	PGD	C&W	AA
ResNet-18	PGD-AT	84.27	52.10	42.34	42.29	40.85	84.27	52.10	42.34	42.29	40.85
	TRADES	82.70	57.14	48.81	48.08	46.46	82.70	57.14	48.81	48.08	46.46
	ARD	84.63	56.57	44.48	43.44	41.66	84.39	52.47	42.18	42.22	40.79
	IAD	84.43	56.64	44.82	43.47	41.80	84.28	52.25	42.16	42.19	40.70
	RSLAD	84.28	57.20	47.17	46.07	44.42	83.83	51.59	42.03	42.22	40.57
	AKD	84.62	56.29	44.42	43.43	41.79	84.32	52.13	42.38	42.44	40.95
	AdaAD	85.07	57.54	47.16	46.06	44.55	85.04	53.85	44.65	44.90	43.27
	IGDM	84.75	58.38	47.56	46.43	44.94	85.67	58.14	48.58	46.98	44.76
	SAAD-C	**85.54**	**61.92**	53.18	52.05	50.14	**86.39**	**60.55**	51.91	52.06	49.72
	SAAD	84.27	61.44	**53.39**	**52.39**	**50.34**	83.69	59.74	**52.89**	**52.36**	**50.35**
MobileNetV2	PGD-AT	83.52	54.92	44.90	44.29	41.54	83.52	54.92	44.90	44.29	41.54
	TRADES	81.79	56.50	49.90	47.54	46.50	81.79	56.50	49.90	47.54	46.50
	ARD	83.66	55.09	44.71	43.49	41.24	83.62	54.62	44.60	44.18	41.47
	IAD	83.85	55.69	44.98	43.62	41.35	83.63	54.81	44.73	44.19	41.51
	RSLAD	83.22	55.54	46.09	44.80	42.56	83.41	54.60	45.01	44.41	41.78
	AKD	83.60	55.13	44.31	43.29	41.03	83.54	54.79	44.83	44.19	41.55
	AdaAD	84.42	56.38	46.16	44.99	43.01	84.37	54.40	44.40	44.59	41.95
	IGDM	84.07	57.31	47.39	45.43	43.57	84.13	57.93	48.70	47.43	44.83
	SAAD-C	**85.16**	**60.53**	52.72	51.26	49.34	**84.81**	**58.17**	51.09	**50.45**	48.08
	SAAD	82.04	59.48	**53.69**	**51.68**	**49.88**	80.60	56.85	**51.78**	50.25	**48.29**

as well as prior distillation techniques. Unlike existing AD methods whose performance varies significantly depending on the teacher model, SAAD maintains strong robustness even under IRT teachers. This suggests that weighting transferable samples during training is crucial for stable robustness transfer in adversarial distillation. Moreover, SAAD-C, which incorporates clean supervision, improves clean accuracy while preserving robustness.

We attribute the superior performance under IRTs to the mitigation of robust overfitting. As discussed in Section 3, robust overfitting arises from high adversarial variance (AVar), particularly when the teacher produces overconfident soft labels. To address this, SAAD introduces a sample-wise weighting scheme that effectively suppresses variance. As shown in Table 5, our method substantially reduces adversarial variance and dramatically mitigates robust overfitting (RO), while increasing the ratio of transferable adversarial samples (TAS). These findings confirm that lowering adversarial variance via our weighting scheme is the key factor driving the improved generalization reported in the main tables.

Table 5: Mitigation of robust overfitting and reduction of adversarial variance via sample-wise weighting with an IRT teacher.

Method	AVar	RO	TAS
Baseline	0.0834	5.44	0.199
SAAD	0.0385	0.93	0.326

5.3 Generalization and Ablation Studies

Adversarial Distillation with ERT. Table 7 demonstrates SAAD's performance in a high-transferability scenario using an ERT. In this setting, SAAD matches or slightly exceeds IGDM. This finding highlights a key aspect of our method: even though SAAD's primary mechanism targets low-transferability, it maintains strong performance in this favorable scenario—a crucial feature given that a teacher's effectiveness is often unknown in practice. Therefore, this result validates SAAD as a robust default: it incurs no performance

Table 6: Adversarial distillation results using ResNet-18 and PreActResNet-18 as student models for CIFAR-100 and Tiny-ImageNet, respectively, with the `Wang2023Better` teacher (WRN-70-16 for CIFAR-100 and WRN-28-10 for Tiny-ImageNet). Clean, FGSM, PGD, C&W, and AA columns report accuracy (%) under each evaluation setting. Results are averaged over three random seeds.

Method	CIFAR-100					Tiny-ImageNet				
	Clean	FGSM	PGD	C&W	AA	Clean	FGSM	PGD	C&W	AA
PGD-AT	56.17	24.74	19.65	19.79	18.66	45.71	15.75	11.67	11.82	10.91
TRADES	53.37	28.72	25.12	23.11	22.32	42.06	19.58	17.15	14.03	13.33
ARD	58.13	29.71	24.97	22.22	20.84	55.69	30.13	26.62	22.09	19.90
IAD	57.59	29.85	25.21	22.41	21.00	53.75	29.85	26.95	22.40	20.56
RSLAD	56.68	30.87	27.27	23.91	22.66	53.18	30.14	27.69	22.86	21.42
AKD	58.22	29.14	24.35	21.80	20.61	54.48	27.62	23.59	19.56	17.73
AdaAD	58.57	31.72	28.00	24.40	23.15	<u>57.26</u>	31.81	28.80	23.64	22.11
IGDM	56.36	32.95	29.68	25.91	24.81	57.15	31.98	29.02	23.94	22.52
SAAD-C	**59.57**	**36.05**	<u>32.52</u>	<u>28.72</u>	<u>27.21</u>	**57.33**	<u>33.06</u>	<u>29.62</u>	<u>24.16</u>	<u>22.69</u>
SAAD	<u>59.11</u>	<u>36.01</u>	**32.71**	**29.36**	**27.58**	57.16	**33.26**	**29.95**	**24.87**	**23.42**

Table 7: Adversarial distillation results on ResNet-18 for CIFAR-10 using an ERT teacher (`Chen2021LTD_WRN34_20`).

Method	Clean	FGSM	PGD	C&W	AA
ARD	85.57	61.07	52.13	50.72	48.78
IAD	84.64	60.78	53.10	50.73	48.67
RSLAD	84.12	60.37	54.68	51.96	50.58
AKD	84.51	60.12	52.17	50.71	49.03
AdaAD	85.10	61.89	56.57	53.59	52.43
IGDM	85.31	62.90	57.28	53.91	52.55
SAAD	85.78	62.77	57.25	53.83	52.69

degradation in favorable settings (with an ERT) while significantly improving robustness when transferability is weak (with an IRT).

OODRobustBench Evaluation. To evaluate the generalization of adversarial robustness beyond in-distribution test sets, we additionally conduct experiments on OODRobustBench (Li et al., 2024), a benchmark specifically designed to assess robustness under distribution shifts. It includes two major types of shifts: dataset shifts and threat shifts. For OOD_d, we report both the clean accuracy on naturally shifted datasets (C-OOD_d) and the robust accuracy under MM5 adversarial attacks (Gao et al., 2022) (R-OOD_d), which encompass corruptions such as noise and blur. On the other hand, OOD_t evaluates robustness against six unseen attack types, including both large-ϵ l_p-norm attacks and non-l_p threat models. As shown in Table 8, SAAD consistently outperforms existing AD methods across both dataset and threat shifts.

Black-box Attack Evaluation. We further evaluate the students against query-based black-box attacks to verify their robustness in scenarios where gradient information is unavailable. We employ RayS (Chen & Gu, 2020), Square Attack (Andriushchenko et al., 2020), and SPSA (Uesato et al., 2018) using the ResNet-18 students distilled from the `Gowal2021Improving` teacher. As presented in Table 8, SAAD exhibits substantially stronger robustness than other distillation methods in this setting. The fact that SAAD maintains high accuracy against these diverse query-based attacks confirms that our method effectively defends against threats regardless of the attacker's access to model gradients.

Table 8: OODRobustBench results and Black-box attack robustness on CIFAR-10. All methods are distilled from the `Gowal2021Improving` teacher to ResNet-18 student.

Method	AA	OODRobustBench			Black-box Attacks		
		C-OOD$_d$	R-OOD$_d$	OOD$_t$	RayS(40k)	Square(5k)	SPSA(40k)
ARD	40.79	73.92	25.87	18.25	46.42	49.57	51.26
IAD	40.70	73.80	25.97	18.63	46.49	49.50	51.28
RSLAD	40.57	74.97	27.25	20.77	48.21	50.89	52.93
AKD	40.95	73.87	25.99	18.25	46.82	49.68	50.95
AdaAD	43.27	74.36	27.43	20.62	48.68	52.08	53.44
IGDM	44.76	71.19	30.40	24.83	49.67	52.01	53.83
SAAD-C	49.72	**76.34**	34.52	26.06	**55.57**	**58.96**	**60.19**
SAAD	**50.35**	74.47	**35.36**	**27.19**	55.47	58.61	60.00

6 Conclusion

In this paper, we challenged the common assumption that a more robust teacher necessarily yields a more robust student in adversarial distillation. We showed that the key bottleneck is not the capacity gap but the transferability of adversarial perturbations between teacher and student. To address this, we introduced Sample-wise Adaptive Adversarial Distillation (SAAD), which dynamically up-weights those examples whose adversarial attacks on the student remain effective for the teacher, and proposed a complementary clean distillation variant (SAAD-C) to recover clean accuracy from non-transferable samples. We experimentally demonstrated that our approach consistently outperforms existing AD methods and even standard adversarial training when transferability is limited.

References

Maksym Andriushchenko, Francesco Croce, Nicolas Flammarion, and Matthias Hein. Square attack: a query-efficient black-box adversarial attack via random search. In *European conference on computer vision*, pp. 484–501. Springer, 2020.

Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *International conference on machine learning*, pp. 274–283. PMLR, 2018.

Muhammad Awais, Fengwei Zhou, Hang Xu, Lanqing Hong, Ping Luo, Sung-Ho Bae, and Zhenguo Li. Adversarial robustness for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8568–8577, 2021.

Yang Bai, Yuyuan Zeng, Yong Jiang, Shu-Tao Xia, Xingjun Ma, and Yisen Wang. Improving adversarial robustness via channel-wise activation suppressing. *arXiv preprint arXiv:2103.08307*, 2021.

Brian R Bartoldson, James Diffenderfer, Konstantinos Parasyris, and Bhavya Kailkhura. Adversarial robustness limits via scaling-law and human-alignment studies. *arXiv preprint arXiv:2404.09349*, 2024.

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

Yair Carmon, Aditi Raghunathan, Ludwig Schmidt, John C Duchi, and Percy S Liang. Unlabeled data improves adversarial robustness. *Advances in neural information processing systems*, 32, 2019.

Alvin Chan, Yi Tay, and Yew-Soon Ong. What it thinks is important is important: Robustness transfers through input gradients. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 332–341, 2020.

Dian Chen, Hongxin Hu, Qian Wang, Li Yinli, Cong Wang, Chao Shen, and Qi Li. Cartl: Cooperative adversarially-robust transfer learning. In *International Conference on Machine Learning*, pp. 1640–1650. PMLR, 2021a.

Erh-Chung Chen and Che-Rung Lee. Ltd: Low temperature distillation for robust adversarial training. *arXiv preprint arXiv:2111.02331*, 2021.

Jianbo Chen, Michael I Jordan, and Martin J Wainwright. Hopskipjumpattack: A query-efficient decision-based attack. In *2020 ieee symposium on security and privacy (sp)*, pp. 1277–1294. IEEE, 2020.

Jinghui Chen and Quanquan Gu. Rays: A ray searching method for hard-label adversarial attack. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 1739–1747, 2020.

Mingyang Chen, Junda Lu, Yi Wang, Jianbin Qin, and Wei Wang. Dair: A query-efficient decision-based attack on image retrieval systems. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 1064–1073, 2021b.

Seungju Cho, Hongsin Lee, and Changick Kim. Enhancing robustness in incremental learning with adversarial training. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 2518–2526, 2025a.

Seungju Cho, Hongsin Lee, and Changick Kim. Long-tailed adversarial training with self-distillation. In *The Thirteenth International Conference on Learning Representations*, 2025b. URL `https://openreview.net/forum?id=vM94dZiqx4`.

Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. Certified adversarial robustness via randomized smoothing. In *international conference on machine learning*, pp. 1310–1320. PMLR, 2019.

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

Francesco Croce, Maksym Andriushchenko, Vikash Sehwag, Edoardo Debenedetti, Nicolas Flammarion, Mung Chiang, Prateek Mittal, and Matthias Hein. Robustbench: a standardized adversarial robustness benchmark. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2021. URL `https://openreview.net/forum?id=SSKZPJCt7B`.

Jiequan Cui, Zhuotao Tian, Zhisheng Zhong, Xiaojuan Qi, Bei Yu, and Hanwang Zhang. Decoupled kullback-leibler divergence loss. *arXiv preprint arXiv:2305.13948*, 2023.

Sihui Dai, Saeed Mahloujifar, and Prateek Mittal. Parameterizing activation functions for adversarial robustness. In *2022 IEEE Security and Privacy Workshops (SPW)*, pp. 80–87. IEEE, 2022.

Nilaksh Das, Madhuri Shanbhogue, Shang-Tse Chen, Fred Hohman, Li Chen, Michael E Kounavis, and Duen Horng Chau. Keeping the bad guys out: Protecting and vaccinating deep learning with jpeg compression. *arXiv preprint arXiv:1705.02900*, 2017.

Jieren Deng, Aaron Palmer, Rigel Mahmood, Ethan Rathbun, Jinbo Bi, Kaleel Mahmood, and Derek Aguiar. Distilling adversarial robustness using heterogeneous teachers. *arXiv preprint arXiv:2402.15586*, 2024.

Ruize Gao, Jiongxiao Wang, Kaiwen Zhou, Feng Liu, Binghui Xie, Gang Niu, Bo Han, and James Cheng. Fast and reliable evaluation of adversarial robustness with minimum-margin attack. In *International Conference on Machine Learning*, pp. 7144–7163. PMLR, 2022.

Micah Goldblum, Liam Fowl, Soheil Feizi, and Tom Goldstein. Adversarially robust distillation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 3996–4003, 2020.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

Sven Gowal, Chongli Qin, Jonathan Uesato, Timothy Mann, and Pushmeet Kohli. Uncovering the limits of adversarial training against norm-bounded adversarial examples. *arXiv preprint arXiv:2010.03593*, 2020.

Sven Gowal, Sylvestre-Alvise Rebuffi, Olivia Wiles, Florian Stimberg, Dan Andrei Calian, and Timothy A Mann. Improving robustness using generated data. *Advances in neural information processing systems*, 34:4218–4233, 2021.

Sorin Grigorescu, Bogdan Trasnea, Tiberiu Cocias, and Gigel Macesanu. A survey of deep learning techniques for autonomous driving. *Journal of Field Robotics*, 37(3):362–386, 2020.

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

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14*, pp. 630–645. Springer, 2016b.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Bo Huang, Mingyang Chen, Yi Wang, Junda Lu, Minhao Cheng, and Wei Wang. Boosting accuracy and robustness of student models via adaptive adversarial distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24668–24677, 2023.

Hanxun Huang, Yisen Wang, Sarah Erfani, Quanquan Gu, James Bailey, and Xingjun Ma. Exploring architectural ingredients of adversarially robust deep neural networks. *Advances in neural information processing systems*, 34:5545–5559, 2021.

Pavel Izmailov, Dmitrii Podoprikhin, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson. Averaging weights leads to wider optima and better generalization. *arXiv preprint arXiv:1803.05407*, 2018.

Gaojie Jin, Xinping Yi, Wei Huang, Sven Schewe, and Xiaowei Huang. Enhancing adversarial training with second-order statistics of weights. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15273–15283, 2022.

Gaojie Jin, Xinping Yi, Dengyu Wu, Ronghui Mu, and Xiaowei Huang. Randomized adversarial training via taylor expansion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16447–16457, 2023.

Jaewon Jung, Hongsun Jang, Jaeyong Song, and Jinho Lee. Peeraid: Improving adversarial distillation from a specialized peer tutor. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24482–24491, 2024.

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

Huafeng Kuang, Hong Liu, Yongjian Wu, Shin'ichi Satoh, and Rongrong Ji. Improving adversarial robustness via information bottleneck distillation. *Advances in Neural Information Processing Systems*, 36:10796–10813, 2023.

Ya Le and Xuan Yang. Tiny imagenet visual recognition challenge. *CS 231N*, 7(7):3, 2015.

Hongsin Lee, Seungju Cho, and Changick Kim. Indirect gradient matching for adversarial robust distillation. In *The Thirteenth International Conference on Learning Representations*, 2025. URL `https://openreview.net/forum?id=juKVq5dWTR`.

Lin Li, Yifei Wang, Chawin Sitawarin, and Michael W. Spratling. OODRobustbench: a benchmark and large-scale analysis of adversarial robustness under distribution shift. In *Forty-first International Conference on Machine Learning*, 2024. URL `https://openreview.net/forum?id=kAFevjEYsz`.

Yanpei Liu, Xinyun Chen, Chang Liu, and Dawn Song. Delving into transferable adversarial examples and black-box attacks. *arXiv preprint arXiv:1611.02770*, 2016.

Xingjun Ma, Yuhao Niu, Lin Gu, Yisen Wang, Yitian Zhao, James Bailey, and Feng Lu. Understanding adversarial attacks on deep learning based medical image analysis systems. *Pattern Recognition*, 110: 107332, 2021.

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.

Kaleel Mahmood, Rigel Mahmood, and Marten van Dijk. On the robustness of vision transformers to adversarial examples. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 7838–7847, October 2021.

Javier Maroto, Guillermo Ortiz-Jiménez, and Pascal Frossard. On the benefits of knowledge distillation for adversarial robustness. *CoRR*, abs/2203.07159, 2022. doi: 10.48550/ARXIV.2203.07159. URL `https://doi.org/10.48550/arXiv.2203.07159`.

Awais Muhammad, Fengwei Zhou, Chuanlong Xie, Jiawei Li, Sung-Ho Bae, and Zhenguo Li. Mixacm: Mixup-based robustness transfer via distillation of activated channel maps. *Advances in neural information processing systems*, 34:4555–4569, 2021.

Nicolas Papernot, Patrick McDaniel, and Ian Goodfellow. Transferability in machine learning: from phenomena to black-box attacks using adversarial samples. *arXiv preprint arXiv:1605.07277*, 2016.

Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z Berkay Celik, and Ananthram Swami. Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia conference on computer and communications security*, pp. 506–519, 2017.

Hyejin Park and Dongbo Min. Dynamic guidance adversarial distillation with enhanced teacher knowledge. In *European Conference on Computer Vision*, pp. 204–219. Springer, 2024.

Chongli Qin, James Martens, Sven Gowal, Dilip Krishnan, Krishnamurthy Dvijotham, Alhussein Fawzi, Soham De, Robert Stanforth, and Pushmeet Kohli. Adversarial robustness through local linearization. *Advances in Neural Information Processing Systems*, 32, 2019.

Sylvestre-Alvise Rebuffi, Sven Gowal, Dan A Calian, Florian Stimberg, Olivia Wiles, and Timothy Mann. Fixing data augmentation to improve adversarial robustness. *arXiv preprint arXiv:2103.01946*, 2021.

Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4510–4520, 2018.

Ludwig Schmidt, Shibani Santurkar, Dimitris Tsipras, Kunal Talwar, and Aleksander Madry. Adversarially robust generalization requires more data. *Advances in neural information processing systems*, 31, 2018.

Vikash Sehwag, Saeed Mahloujifar, Tinashe Handina, Sihui Dai, Chong Xiang, Mung Chiang, and Prateek Mittal. Robust learning meets generative models: Can proxy distributions improve adversarial robustness? *arXiv preprint arXiv:2104.09425*, 2021.

Ali Shafahi, Parsa Saadatpanah, Chen Zhu, Amin Ghiasi, Christoph Studer, David Jacobs, and Tom Goldstein. Adversarially robust transfer learning. *arXiv preprint arXiv:1905.08232*, 2019.

Rulin Shao, Jinfeng Yi, Pin-Yu Chen, and Cho-Jui Hsieh. How and when adversarial robustness transfers in knowledge distillation? *arXiv preprint arXiv:2110.12072*, 2021.

Kaustubh Sridhar, Oleg Sokolsky, Insup Lee, and James Weimer. Improving neural network robustness via persistency of excitation. In *2022 American Control Conference (ACC)*, pp. 1521–1526. IEEE, 2022.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

Jihoon Tack, Sihyun Yu, Jongheon Jeong, Minseon Kim, Sung Ju Hwang, and Jinwoo Shin. Consistency regularization for adversarial robustness. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 8414–8422, 2022.

Florian Tramèr, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. The space of transferable adversarial examples. *arXiv preprint arXiv:1704.03453*, 2017.

Jonathan Uesato, Brendan O'donoghue, Pushmeet Kohli, and Aaron Oord. Adversarial risk and the dangers of evaluating against weak attacks. In *International conference on machine learning*, pp. 5025–5034. PMLR, 2018.

Pratik Vaishnavi, Kevin Eykholt, and Amir Rahmati. Transferring adversarial robustness through robust representation matching. In *31st USENIX security symposium (USENIX Security 22)*, pp. 2083–2098, 2022.

Ningfei Wang, Yunpeng Luo, Takami Sato, Kaidi Xu, and Qi Alfred Chen. Does physical adversarial example really matter to autonomous driving? towards system-level effect of adversarial object evasion attack. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4412–4423, 2023a.

Yifei Wang, Liangchen Li, Jiansheng Yang, Zhouchen Lin, and Yisen Wang. Balance, imbalance, and rebalance: Understanding robust overfitting from a minimax game perspective. *Advances in neural information processing systems*, 36:15775–15798, 2023b.

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*, 2020.

Zekai Wang, Tianyu Pang, Chao Du, Min Lin, Weiwei Liu, and Shuicheng Yan. Better diffusion models further improve adversarial training. In *International Conference on Machine Learning (ICML)*, 2023c.

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*, pp. 8193–8201, 2023.

Dongxian Wu, Shu-Tao Xia, and Yisen Wang. Adversarial weight perturbation helps robust generalization. *Advances in Neural Information Processing Systems*, 33:2958–2969, 2020.

Cihang Xie, Yuxin Wu, Laurens van der Maaten, Alan L Yuille, and Kaiming He. Feature denoising for improving adversarial robustness. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 501–509, 2019.

Yaodong Yu, Zitong Yang, Edgar Dobriban, Jacob Steinhardt, and Yi Ma. Understanding generalization in adversarial training via the bias-variance decomposition. *arXiv preprint arXiv:2103.09947*, 2021.

Xinli Yue, Mou Ningping, Qian Wang, and Lingchen Zhao. Revisiting adversarial robustness distillation from the perspective of robust fairness. *Advances in Neural Information Processing Systems*, 36:30390–30401, 2023.

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*, pp. 7472–7482. PMLR, 2019.

Qingzhao Zhang, Shengtuo Hu, Jiachen Sun, Qi Alfred Chen, and Z Morley Mao. On adversarial robustness of trajectory prediction for autonomous vehicles. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15159–15168, 2022.

Shiji Zhao, Jie Yu, Zhenlong Sun, Bo Zhang, and Xingxing Wei. Enhanced accuracy and robustness via multi-teacher adversarial distillation. In *European Conference on Computer Vision*, pp. 585–602. Springer, 2022.

Shiji Zhao, Ranjie Duan, Xizhe Wang, and Xingxing Wei. Improving adversarial robust fairness via anti-bias soft label distillation. *Advances in Neural Information Processing Systems*, 37:89125–89149, 2024.

Helong Zhou, Liangchen Song, Jiajie Chen, Ye Zhou, Guoli Wang, Junsong Yuan, and Qian Zhang. Rethinking soft labels for knowledge distillation: A bias-variance tradeoff perspective. *arXiv preprint arXiv:2102.00650*, 2021.

Jianing Zhu, Jiangchao Yao, Bo Han, Jingfeng Zhang, Tongliang Liu, Gang Niu, Jingren Zhou, Jianliang Xu, and Hongxia Yang. Reliable adversarial distillation with unreliable teachers. *arXiv preprint arXiv:2106.04928*, 2021.

Bojia Zi, Shihao Zhao, Xingjun Ma, and Yu-Gang Jiang. Revisiting adversarial robustness distillation: Robust soft labels make student better. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 16443–16452, 2021.

A Implementation Details

A.1 Robust Teacher and Analysis Details

Table 9 summarizes the teacher models used in our study, all selected from RobustBench (Croce et al., 2021). Table 10 presents the results of adversarial distillation from each teacher into a ResNet-18 student, using six recent distillation methods. For each teacher, we report the student's AutoAttack accuracy under each method, the average accuracy across methods (Mean AA), and the transferable adversarial sample ratio (TAS). As introduced in Section 3, we divide teachers into two groups for clear interpretation: *Effective Robust Teachers* (ERTs) and *Ineffective Robust Teachers* (IRTs). This categorization is based on the Mean AA value: a teacher is labeled as an ERT if the Mean AA exceeds the TRADES baseline (46.46%) by more than 3 percentage points, and as an IRT if it falls below that baseline by more than 3 points. The $\pm 3\%$ band and the ERT/IRT split are used only in Section 3 for analysis, to highlight clearly separated case. All analyses presented in the main paper—including TAS, adversarial variance (AVar), robust overfitting (RO), and baseline comparisons in Table 5—are conducted using RSLAD as the baseline AD method.

Table 9: Summary of teacher models used in our study, selected from RobustBench. 'Abbr.' denotes the shorthand identifier used in Figure 1. **Bolded entries** correspond to the teachers analyzed in Section 3; in that section, their RobustBench names are shown without architecture suffixes (e.g., `Bartoldson2024Adversarial_WRN-94-16` is referred to as `Bartoldson2024Adversarial`)

Abbr.	RobustBench name	Architecture	Size(M)	Clean	AA
Bart94	**Bartoldson2024Adversarial_WRN-94-16** (Bartoldson et al., 2024)	WRN-94-16	365.92	93.68	73.71
Bart82	Bartoldson2024Adversarial_WRN-82-8 (Bartoldson et al., 2024)	WRN-82-8	79.13	93.11	71.59
Wang70	Wang2023Better_WRN-70-16 (Wang et al., 2023c)	WRN-70-16	266.80	93.25	70.69
Cui28	Cui2023Decoupled_WRN-28-10 (Cui et al., 2023)	WRN-28-10	36.48	92.16	67.73
Gowal70	Gowal2020Uncovering_70_16_extra (Gowal et al., 2020)	WRN-70-16	266.80	91.10	65.87
Rebu70	**Rebuffi2021Fixing_70_16_cutmix_ddpm** (Rebuffi et al., 2021)	WRN-70-16	266.80	88.54	64.20
Gowal28	**Gowal2021Improving_28_10_ddpm_100m** (Gowal et al., 2021)	WRN-28-10	36.48	87.50	63.38
Huang	Huang2021Exploring_ema (Huang et al., 2021)	WRN-34-R	68.12	91.23	62.54
Dai	Dai2021Parameterizing (Dai et al., 2022)	WRN-28-10	36.48	87.02	61.55
Sridhar34	Sridhar2021Robust_34_15 (Sridhar et al., 2022)	WRN-34-15	108.53	86.53	60.41
Carmon	Carmon2019Unlabeled (Carmon et al., 2019)	WRN-28-10	36.48	89.69	59.53
GowalR18	Gowal2021Improving_R18_ddpm_100m (Gowal et al., 2021)	PreActRN-18	12.55	87.35	58.50
Chen34-20	Chen2021LTD_WRN34_20 (Chen & Lee, 2021)	WRN-34-20	184.53	86.03	57.71
Chen34-10	**Chen2021LTD_WRN34_10** (Chen & Lee, 2021)	WRN-34-10	46.16	85.21	56.94
SehwagR18	Sehwag2021Proxy_R18 (Sehwag et al., 2021)	RN-18	11.17	84.59	55.54

A.2 Adversarial Variance Details

In this section, we provide further technical details on the computation of adversarial variance and the associated decomposition of adversarial error. Algorithm 1 outlines the procedure used throughout the main paper to estimate adversarial variance under AD. This algorithm quantifies how much the learned student model varies when trained on different subsets of the training data, using a fixed AD method. As shown in the algorithm, we split the full training dataset $\mathcal{D}$ into N disjoint subsets and independently train student models on each subset using a fixed AD method. This allows us to observe how much the resulting models vary in their outputs under adversarial evaluation. For each trained student, the variation is measured by evaluating each model's prediction at a fixed test point $(\mathbf{x}, \mathbf{y})$ under its corresponding adversarial input, and computing the KL divergence between these predictions and their geometric mean, reflecting how much model outputs fluctuate due to data-induced randomness. Following (Yu et al., 2021), we set the number of splits $N = 2$, corresponding to training each student on half of CIFAR-10 (25,000 examples). To obtain more stable estimates, we repeat this procedure $K = 2$ times with different random splits.

To further clarify the theoretical interpretation of our measure, we restate and prove a bias–variance decomposition of the expected adversarial error, following formulations introduced in prior works (Yu et al., 2021; Zhou et al., 2021). We explicitly show how the adversarial prediction error decomposes into three terms: intrinsic noise, adversarial bias, and adversarial variance. This derivation justifies our use of KL-

Table 10: AutoAttack accuracy (%) of student models distilled from each RobustBench teacher using different methods. Each row corresponds to a teacher model shown, and columns represent distillation methods in Figure 1. Mean AA denotes the average performance across all methods for a given teacher. TAS indicates the transferable adversarial sample ratio with RSLAD method. **Bold Mean AA** values indicate ERTs, whose students outperform the TRADES baseline (46.46%) by more than 3 percentage points. Underlined Mean AA values denote IRTs, whose students fall short of the baseline by more than 3 points.

Abbr.	ARD	IAD	RSLAD	AKD	AdaAD	IGDM	Mean AA	TAS
Bart94	41.94	41.83	44.07	41.73	44.68	44.75	43.17	0.1991
Bart82	44.93	45.07	47.59	44.02	48.63	48.80	46.51	0.3779
Wang70	44.59	44.75	47.38	44.53	48.47	48.66	46.40	0.3656
Cui28	46.16	46.20	49.07	45.87	50.50	50.79	48.10	0.5400
Gowal70	45.88	46.84	48.89	45.82	50.82	50.82	48.18	0.5774
Rebu70	47.75	47.95	50.94	48.11	52.14	52.28	**50.20**	0.6770
Gowal28	40.94	40.85	41.08	41.13	43.01	44.26	42.05	0.1494
Huang	46.00	45.81	49.09	45.26	49.20	49.26	47.44	0.4431
Dai	40.64	40.52	42.28	40.61	42.87	43.92	41.81	0.2175
Sridhar34	47.72	46.89	49.89	46.29	51.50	52.26	48.84	0.8133
Carmon	46.16	45.94	49.56	46.56	51.26	51.53	48.50	0.6450
GowalR18	44.12	43.62	46.61	44.41	49.51	50.46	46.72	0.4213
Chen34-20	48.78	48.67	50.58	49.03	52.43	52.55	**50.34**	0.9262
Chen34-10	50.82	50.55	52.21	50.95	53.24	53.45	**51.87**	0.9810
SehwagR18	42.30	42.28	45.33	43.65	48.95	49.69	45.37	0.3689

based adversarial variance as a meaningful quantity for analyzing robustness under adversarial training and distillation.

We aim to show that the expected adversarial error satisfies the following lemma:

Lemma 1 (Decomposition of Expected Adversarial Error)**.** *Let* $(\mathbf{x}, \mathbf{y})$ *be a test example, where* $\mathbf{y} = [y_1, \ldots, y_C]^\top \in \Delta^{C-1}$ *is the target class–probability vector over* C *classes. Let* $\mathcal{D}$ *denote the training dataset. Define the adversarial prediction*

$$\hat{\mathbf{y}} \;=\; f_{\hat{\boldsymbol{\theta}}(\mathcal{D})}\big(\mathbf{x} + \boldsymbol{\delta}(\mathbf{x}, \mathbf{y}, \mathcal{D})\big), \tag{10}$$

where $\boldsymbol{\delta}(\mathbf{x}, \mathbf{y}, \mathcal{D})$ *is the worst–case perturbation within the chosen threat model. Let the* normalized geometric mean *of predictions across* $\mathcal{D}$ *be*

$$\bar{\mathbf{y}} \;:=\; \frac{1}{Z(\mathbf{x})} \exp\Big(\mathbb{E}_{\mathcal{D}}[\log \hat{\mathbf{y}}]\Big), \qquad Z(\mathbf{x}) \text{ chosen so } \sum_{c=1}^{C} \bar{y}_c = 1. \tag{11}$$

The expected adversarial cross–entropy

$$\mathrm{CE}(\mathbf{y}, \hat{\mathbf{y}}) \;:=\; -\sum_{c=1}^{C} y_c \log \hat{y}_c \tag{12}$$

admits the decomposition

$$\mathbb{E}_{\mathbf{x},\mathcal{D}}\left[\mathrm{CE}(\mathbf{y}, \hat{\mathbf{y}})\right] = \underbrace{\mathbb{E}_{\mathbf{x}}\left[-\mathbf{y} \log \mathbf{y}\right]}_{\textit{Intrinsic Noise}} + \underbrace{\mathbb{E}_{\mathbf{x}}\left[\mathbf{y} \log \frac{\mathbf{y}}{\bar{\mathbf{y}}}\right]}_{\textit{Adversarial Bias}} + \underbrace{\mathbb{E}_{\mathbf{x},\mathcal{D}}\left[\mathrm{KL}(\bar{\mathbf{y}} \,\|\, \hat{\mathbf{y}})\right]}_{\textit{Adversarial Variance}}. \tag{13}$$

Proof. Fix a test point $(\mathbf{x}, \mathbf{y})$. By definition,

$$\mathrm{CE}(\mathbf{y}, \hat{\mathbf{y}}) = -\,\mathbf{y} \log \hat{\mathbf{y}}. \tag{14}$$

Algorithm 1 Estimating Adversarial Variance under Adversarial Distillation

Require: Test point $(\mathbf{x}, \mathbf{y})$, dataset $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{n}$, teacher f_T, number of splits N, repetitions K

1: **for** $k = 1$ **to** K **do**

2: Randomly split $\mathcal{D}$ into $\{\mathcal{D}_j^{(k)}\}_{j=1}^{N}$.

3: **for** $j = 1$ **to** N **do**

4: Adversarially distill student $f_S(\,\cdot\,;\boldsymbol{\theta})$ on $\mathcal{D}_j^{(k)}$ with f_T:

$$\hat{\boldsymbol{\theta}}\big(\mathcal{D}_j^{(k)}\big) \;\approx\; \arg\min_{\boldsymbol{\theta}} \frac{1}{|\mathcal{D}_j^{(k)}|} \sum_{i \in \mathcal{D}_j^{(k)}} \Big[\max_{\boldsymbol{\delta}_{S,i} \in \Delta} L_{\mathrm{AD}}(f_S, f_T, \mathbf{x}_i, \boldsymbol{\delta}_{S,i}) \Big].$$

5: Find adversarial perturbation $\boldsymbol{\delta}_j := \boldsymbol{\delta}(\mathbf{x}, \mathbf{y}, \mathcal{D}_j^{(k)})$ that approximately solves

$$\max_{\boldsymbol{\delta} \in \Delta} L_{\max}\big(f_{\hat{\boldsymbol{\theta}}(\mathcal{D}_j^{(k)})}(\mathbf{x} + \boldsymbol{\delta})\,, \mathbf{y}\big),$$

6: Evaluate the adversarial student prediction $\hat{\mathbf{y}}_j := f_S(\mathbf{x} + \boldsymbol{\delta}_j;\, \hat{\boldsymbol{\theta}}(\mathcal{D}_j^{(k)}))$.

7: **end for**

8: Aggregate via the geometric mean on the simplex:

$$\bar{\mathbf{y}} := \frac{1}{Z} \exp\left(\frac{1}{N} \sum_{j=1}^{N} \log \hat{\mathbf{y}}_j \right), \quad Z \text{ is a normalization constant.}$$

9: Compute the KL-based variance for split k:

$$\widehat{\mathrm{AVar}}_{\mathrm{KL}}\big(\mathbf{x}, \mathbf{y}, \mathcal{D}^{(k)}\big) = \frac{1}{N} \sum_{j=1}^{N} \mathrm{KL}\big(\bar{\mathbf{y}} \,\|\, \hat{\mathbf{y}}_j\big).$$

10: **end for**

11: **Return** the averaged adversarial variance

$$\widehat{\mathrm{AVar}}_{\mathrm{KL}}(\mathbf{x}, \mathbf{y}) = \frac{1}{K} \sum_{k=1}^{K} \widehat{\mathrm{AVar}}_{\mathrm{KL}}\big(\mathbf{x}, \mathbf{y}, \mathcal{D}^{(k)}\big).$$

We add and subtract the term $\mathbf{y} \log \bar{\mathbf{y}}$, yielding:

$$\mathrm{CE}(\mathbf{y}, \hat{\mathbf{y}}) = -\,\mathbf{y} \log \hat{\mathbf{y}} = -\,\mathbf{y} \log \mathbf{y} + \mathbf{y} \log \frac{\mathbf{y}}{\bar{\mathbf{y}}} + \mathbf{y} \log \frac{\bar{\mathbf{y}}}{\hat{\mathbf{y}}}. \tag{15}$$

Taking expectation over $\mathcal{D}$ on both sides gives:

$$\mathbb{E}_{\mathcal{D}}\big[\mathrm{CE}(\mathbf{y}, \hat{\mathbf{y}})\big] = -\,\mathbf{y} \log \mathbf{y} + \mathbf{y} \log \frac{\mathbf{y}}{\bar{\mathbf{y}}} + \mathbb{E}_{\mathcal{D}}\Big[\mathbf{y} \log \frac{\bar{\mathbf{y}}}{\hat{\mathbf{y}}}\Big]. \tag{16}$$

The first term corresponds to the intrinsic noise, the second to adversarial bias, and it remains to show that the third term equals the adversarial variance:

$$\mathbb{E}_{\mathcal{D}}\Big[\mathbf{y} \log \frac{\bar{\mathbf{y}}}{\hat{\mathbf{y}}}\Big] = \mathbb{E}_{\mathcal{D}}\big[\mathrm{KL}(\bar{\mathbf{y}} \| \hat{\mathbf{y}})\big]. \tag{17}$$

To see this, recall that $\log \bar{\mathbf{y}} = \mathbb{E}_{\mathcal{D}}[\log \hat{\mathbf{y}}] - \log Z$ component-wise. For each class c,

$$\mathbb{E}_{\mathcal{D}}\Big[y_c \log \tfrac{\bar{y}_c}{\hat{y}_c}\Big] = y_c \left(\mathbb{E}_{\mathcal{D}}[\log \hat{y}_c] - \log Z\right) - y_c\, \mathbb{E}_{\mathcal{D}}[\log \hat{y}_c] = -y_c \log Z. \tag{18}$$

Summing over all classes gives:

$$\mathbb{E}_{\mathcal{D}}\Big[\mathbf{y}\log\frac{\bar{\mathbf{y}}}{\hat{\mathbf{y}}}\Big] = -\log Z\sum_c y_c = -\log Z. \tag{19}$$

On the other hand:

$$\begin{aligned}\mathbb{E}_{\mathcal{D}}[\mathrm{KL}(\bar{\mathbf{y}}\|\hat{\mathbf{y}})] &= \mathbb{E}_{\mathcal{D}}\left[\sum_c \bar{y}_c\log\frac{\bar{y}_c}{\hat{y}_c}\right]\\ &= \sum_c \bar{y}_c\left(\mathbb{E}_{\mathcal{D}}[\log\hat{y}_c] - \log Z - \mathbb{E}_{\mathcal{D}}[\log\hat{y}_c]\right)\\ &= -\log Z\sum_c \bar{y}_c = -\log Z.\end{aligned} \tag{20}$$

Since $\sum_c y_c = \sum_c \bar{y}_c = 1$, the two expressions are equal, completing the proof. □

A.3 Proposed Method Details

A.3.1 From TAS to the surrogate loss

Let C be the number of classes and Δ^{C-1} the probability simplex. For an input $\mathbf{x}$, define the teacher's predictive distributions under student- and teacher-crafted adversarial perturbations by

$$\boldsymbol{p}(\mathbf{x}) \coloneqq f_T(\mathbf{x}+\boldsymbol{\delta}_S) \in \Delta^{C-1}, \qquad \boldsymbol{q}(\mathbf{x}) \coloneqq f_T(\mathbf{x}+\boldsymbol{\delta}_T) \in \Delta^{C-1}, \tag{21}$$

with components $p_i = [\boldsymbol{p}(\mathbf{x})]_i$ and $q_i = [\boldsymbol{q}(\mathbf{x})]_i$. We define the TAS score by

$$\mathrm{TAS}(\mathbf{x}) \;:=\; \mathrm{KL}\big(\boldsymbol{p}(\mathbf{x})\,\|\,f_T(\mathbf{x})\big) \;-\; \mathrm{KL}\big(\boldsymbol{p}(\mathbf{x})\,\|\,\boldsymbol{q}(\mathbf{x})\big). \tag{22}$$

We call $\mathbf{x}$ a transferable adversarial sample if $\mathrm{TAS}(\mathbf{x}) \geq 0$, which is equivalent to

$$\mathrm{KL}\big(f_T(\mathbf{x}+\boldsymbol{\delta}_S)\,\|\,f_T(\mathbf{x})\big) \;\geq\; \mathrm{KL}\big(f_T(\mathbf{x}+\boldsymbol{\delta}_S)\,\|\,f_T(\mathbf{x}+\boldsymbol{\delta}_T)\big). \tag{23}$$

In the main text, for analysis, we use $\mathrm{TAS}(\mathbf{x}) \geq 0$ to decide whether a sample is TAS; otherwise it is non-TAS. It cleanly separates samples and reveals how the two groups differ in (i) the entropy of teacher logits on student-crafted adversarial inputs (Figure 4a), (ii) adversarial variance when training on each subset after a warm-up phase (Figure 4b), and (iii) robust overfitting trajectories (Figure 4c). These results show that the non-TAS subset concentrates low-entropy, high-variance supervision and drives robust overfitting, whereas the TAS subset exhibits higher entropy and more stable generalization.

While the binary split is useful for diagnostics, transferability is not inherently binary for training: samples lie at different distances from the boundary $\mathrm{TAS}(\mathbf{x})=0$, stochasticity near that boundary can flip membership, and discarding non-TAS samples is data-inefficient. Therefore, for training we replace the binary split by a continuous score, which we map to sample weights as in equation 8.

For training-time efficiency, we avoid computing the teacher-side perturbation $\boldsymbol{\delta}_T$ and instead use an entropy-based proxy on $f_T(\mathbf{x}+\boldsymbol{\delta}_S)$ in equation 8. Assuming a strong white-box $\boldsymbol{\delta}_T$ yields a high-entropy, non-degenerate teacher distribution $\boldsymbol{q}(\mathbf{x}) = f_T(\mathbf{x}+\boldsymbol{\delta}_T)$ (so $m = \min_i q_i > 0$), we have:

Lemma 2 (Entropy-based lower bound on TAS)**.** *Assume the teacher's adversarial output satisfies* $m = \min_i q_i > 0$*, where* $\boldsymbol{q}(\mathbf{x}) = f_T(\mathbf{x}+\boldsymbol{\delta}_T)$ *and* $q_i = [\boldsymbol{q}(\mathbf{x})]_i$*. Then*

$$\mathrm{TAS}(\mathbf{x}) \;\geq\; H(f_T(\mathbf{x}+\boldsymbol{\delta}_S)) \;+\; \log m. \tag{24}$$

Proof. Let $H(\boldsymbol{p}) = -\sum_i p_i\log p_i$. By the definition of KL divergence,

$$\mathrm{KL}(\boldsymbol{p}\|\boldsymbol{q}) = -H(\boldsymbol{p}) \;-\; \sum_i p_i\log q_i \;\leq\; -H(\boldsymbol{p}) \;-\; \log m. \tag{25}$$

Table 11: Comparison of inner maximization ($L_{\max}$), outer minimization ($L_{\min}$) for various AD methods. As IGDM follows a modular design, L_{AD} can be any other AD outer minimization loss.

Method	Inner Maximization	Outer Minimization
ARD	$\mathrm{CE}\big(\mathbf{y}, f_S(\mathbf{x}+\boldsymbol{\delta})\big)$	$\mathrm{KL}\big(f_T(\mathbf{x}) \| f_S(\mathbf{x}+\boldsymbol{\delta})\big)$
RSLAD	$\mathrm{KL}\big(f_T(\mathbf{x}) \| f_S(\mathbf{x}+\boldsymbol{\delta})\big)$	$\mathrm{KL}\big(f_T(\mathbf{x}) \| f_S(\mathbf{x}+\boldsymbol{\delta})\big)$
AdaAD	$\mathrm{KL}\big(f_T(\mathbf{x}+\boldsymbol{\delta}) \| f_S(\mathbf{x}+\boldsymbol{\delta})\big)$	$\mathrm{KL}\big(f_T(\mathbf{x}+\boldsymbol{\delta}) \| f_S(\mathbf{x}+\boldsymbol{\delta})\big)$
IGDM	$\mathrm{KL}\big(f_T(\mathbf{x}+\boldsymbol{\delta}) \| f_S(\mathbf{x}+\boldsymbol{\delta})\big)$	$L_{\mathrm{AD}} + \alpha_{\mathrm{IGDM}} \cdot \mathrm{KL}\big(f_T(\mathbf{x}+\boldsymbol{\delta}) - f_T(\mathbf{x}-\boldsymbol{\delta}) \| f_S(\mathbf{x}+\boldsymbol{\delta}) - f_S(\mathbf{x}-\boldsymbol{\delta})\big)$

Since $\mathrm{TAS}(\mathbf{x}) = \mathrm{KL}\big(\boldsymbol{p} \| f_T(\mathbf{x})\big) - \mathrm{KL}(\boldsymbol{p} \| \boldsymbol{q})$, we obtain

$$\mathrm{TAS}(\mathbf{x}) \;\geq\; H(\boldsymbol{p}) + \log m = H(f_T(\mathbf{x}+\boldsymbol{\delta}_S)) + \log m. \tag{26}$$

□

This bound justifies using entropy as a surrogate for sample-level transferability: higher teacher entropy on $\mathbf{x}+\boldsymbol{\delta}_S$ increases a lower bound on $\mathrm{TAS}(\mathbf{x})$, providing a computation-friendly proxy without evaluating $\boldsymbol{\delta}_T$.

A.3.2 Revisiting Previous Adversarial Distillation Methods

Existing AD methods have largely been designed and evaluated under ERTs, with limited analysis conducted in the context of IRTs. While overall distillation performance is not effective under IRTs, as shown in Figure 1, Table 4, and Table 6, our analysis reveals that some AD methods still perform comparatively better than other AD methods even under IRTs. Table 11 summarizes representative AD methods, focusing on their inner maximization and outer minimization formulations. The differences in these formulations determine how each method responds to the adversarial signal provided by the teacher. Among them, IGDM further refines robustness alignment by implicitly matching gradients (adversarial direction) through logit difference minimization in the outer optimization. This design can be interpreted as encouraging transferability between the student and teacher. Empirically, IGDM consistently yields improved robustness compared to prior methods, particularly under IRTs, which motivates our decision to adopt IGDM as our baseline. Since IGDM is a modular design rather than a complete distillation framework, we adopt it in combination with AdaAD, following the original implementation (Lee et al., 2025). Therefore, the overall SAAD minimization loss is defined as:

$$L_{\mathrm{SAAD}} = \frac{1}{N}\sum_{i=1}^{N} w_i \cdot L_{\mathrm{AD}}(f_S, f_T, \mathbf{x}_i, \boldsymbol{\delta}_i), \quad w_i := H\left(f_T(\mathbf{x}_i+\boldsymbol{\delta}_i)\right), \tag{27}$$

where $H(\cdot)$ denotes the entropy function, and L_{AD} represents the base adversarial distillation loss using AdaAD with IGDM, formulated as:

$$\begin{aligned} L_{\mathrm{AD}}(f_S, f_T, \mathbf{x}_i, \boldsymbol{\delta}_i) = {} & \mathrm{KL}\left(f_T(\mathbf{x}_i+\boldsymbol{\delta}_i) \,\|\, f_S(\mathbf{x}_i+\boldsymbol{\delta}_i)\right) \\ & + \alpha_{\mathrm{IGDM}} \cdot \mathrm{KL}\left(f_T(\mathbf{x}_i+\boldsymbol{\delta}_i) - f_T(\mathbf{x}_i) \,\|\, f_S(\mathbf{x}_i+\boldsymbol{\delta}_i) - f_S(\mathbf{x}_i)\right). \end{aligned} \tag{28}$$

In results, our baseline implementation follows AdaAD with the IGDM module, along with an weight averaging (SWA) (Izmailov et al., 2018).

A.3.3 Fast Inner Maximization via Linear Approximation

We also introduce a lightweight inner maximization strategy to reduce computational cost. This technique is a practical enhancement to the SAAD framework by avoiding repeated teacher backpropagation, enabling efficient training without compromising robustness. Recent AD methods have adopted iterative inner maximization procedures involving teacher backpropagation (Huang et al., 2023; Lee et al., 2025), a strategy that has empirically led to stronger robustness in distilled students. However, as the size of the teacher model

Algorithm 2 Fast Inner Maximization with First-Order Teacher Logit Correction

Require: Input $\mathbf{x}$, true label y, student f_S, teacher f_T, step size η, perturbation bound ϵ, steps K, correction weight λ_{in}

Compute teacher logits on $\mathbf{x}$: ℓ_T
Compute gradient $\nabla_{\mathbf{x}} \ell_T^y(\mathbf{x})$, the input gradient of the logit corresponding to class y
Initialize perturbation: $\boldsymbol{\delta} \leftarrow 0.001 \cdot \mathcal{N}(0, I)$
for $k = 1$ to K **do**
  Compute correction term:

$$\Delta \ell^y \leftarrow \lambda_{\text{in}} \cdot \langle \nabla_{\mathbf{x}} \ell_T^y(\mathbf{x}), \boldsymbol{\delta} \rangle$$

  Construct corrected logits: $\tilde{\ell}_T \leftarrow \ell_T$, with $\tilde{\ell}_T^y \leftarrow \ell_T^y + \Delta \ell^y$
  Compute loss: $\mathcal{L}_{\text{KL}} := \text{KL}\left(\text{softmax}(\tilde{\ell}_T) \,\|\, f_S(\mathbf{x} + \boldsymbol{\delta})\right)$
  Update perturbation:

$$\boldsymbol{\delta} \leftarrow \text{Proj}_{\boldsymbol{\delta} \in \Delta}\left(\boldsymbol{\delta} + \eta \cdot \text{sign}\left(\nabla_{\boldsymbol{\delta}} \mathcal{L}_{\text{KL}}\right)\right)$$

  Project $\mathbf{x} + \boldsymbol{\delta} \in [0, 1]^d$
end for
return $\mathbf{x} + \boldsymbol{\delta}$

increases, this process becomes computationally expensive. To alleviate this cost, we introduce an approximation technique that reduces the number of teacher backpropagations from multiple iterations to a single step. Inspired by the locally linear behavior of adversarially trained models discussed in prior work (Lee et al., 2025), we linearize the teacher's output around the input $\mathbf{x}$ using a first-order Taylor expansion:

$$f_T(\mathbf{x} + \boldsymbol{\delta}) \approx f_T(\mathbf{x}) + \langle \nabla_{\mathbf{x}} f_T(\mathbf{x}), \boldsymbol{\delta} \rangle, \tag{29}$$

and substitute this into the inner maximization objective:

$$\max_{\boldsymbol{\delta} \in \Delta} \text{KL}\left(f_T(\mathbf{x} + \boldsymbol{\delta}) \,\|\, f_S(\mathbf{x} + \boldsymbol{\delta})\right), \tag{30}$$

which results in the following approximation:

$$\max_{\boldsymbol{\delta} \in \Delta} \text{KL}\left(f_T(\mathbf{x} + \boldsymbol{\delta}) \,\|\, f_S(\mathbf{x}) + \langle \nabla_{\mathbf{x}} f_S(\mathbf{x}), \boldsymbol{\delta} \rangle\right). \tag{31}$$

This formulation enables a technically efficient alternative to conventional inner maximization by bypassing the need for iterative teacher backpropagation while preserving the gradient-guided adversarial direction. Algorithm 2 details the procedure. Given a clean input and a true label, we first compute the teacher logits and the corresponding input gradient. At each step, the KL divergence is computed between the student output and the corrected teacher logits. By adopting this approximation, we achieve a nearly 4× speed-up in the inner maximization step compared to the original AdaAD formulation in Table 17, while maintaining comparable robustness and distillation performance.

A.3.4 Main Algorithm

The overall training procedure for SAAD and SAAD-C is outlined in Algorithm 3.

B Experimental Details

B.1 Adversarial Distillation Settings

We conduct experiments on CIFAR-10, CIFAR-100, and Tiny-ImageNet, applying standard data augmentations (random cropping and horizontal flipping). All student models are trained for 200 epochs using SGD with momentum 0.9, weight decay 5×10^{-4}, and an initial learning rate of 0.1, which is decayed by a factor of 10 at the 100th and 150th epochs. The batch size is fixed to 128 across all experiments.

Algorithm 3 Training Algorithm for SAAD and SAAD-C

Require: Teacher f_T, student f_S, training set $\mathcal{D}$, step size η, total epochs E, perturbation bound ϵ, inner steps K, weights $\lambda_{\text{in}}, \alpha_{\text{IGDM}}, \beta$

for epoch $= 1$ to E **do**
  for each minibatch $\{(\mathbf{x}_i, y_i)\}_{i=1}^{B} \sim \mathcal{D}$ **do**
    Generate $\boldsymbol{\delta}_i$ for each $\mathbf{x}_i$ using Algorithm 2
    Compute per-sample entropy: $w_i \leftarrow H(f_T(\mathbf{x}_i + \boldsymbol{\delta}_i))$
    Normalize: $\hat{w}_i \leftarrow \text{Normalize}(w_i)$
    Compute loss L_{AD} as in equation 28
    Compute total loss for SAAD equation 8:

$$\ell_i \leftarrow w_i \cdot L_{\text{AD}}(\mathbf{x}_i, \boldsymbol{\delta}_i)$$

    if SAAD-C **then**
      Add clean distillation:

$$\ell_i \leftarrow \ell_i + \beta \cdot (1 - \hat{w}_i) \cdot \text{KL}(f_T(\mathbf{x}_i) \,\|\, f_S(\mathbf{x}_i))$$

    end if
    $\mathcal{L} \leftarrow \frac{1}{B} \sum_{i=1}^{B} \ell_i$
    Update f_S via gradient descent: $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \eta \nabla_{\boldsymbol{\theta}} \mathcal{L}$
  end for
  if epoch ≥ 95 **then**
    Update SWA parameters
  end if
end for

We compare two adversarial training baselines—PGD-AT and TRADES—alongside six recent adversarial distillation methods: ARD, IAD, RSLAD, AKD, AdaAD, and IGDM. As IGDM is a modular technique, we evaluate its performance in conjunction with AdaAD (i.e., AdaAD+IGDM), which consistently yields the best results among IGDM-integrated variants. Following the original design, we set the IGDM hyperparameter α_{IGDM} to 1 for CIFAR-10, 20 for CIFAR-100, and 10 for Tiny-ImageNet. This configuration is also adopted in our SAAD framework to ensure consistency across evaluations.

For inner maximization in training, we adopt a standard multi-step attack setup with L_∞ perturbations bounded by $\epsilon = 8/255$, step size $2/255$, and 10 iterations. Each method follows its original inner maximization formulation, summarized in Table 11. Specifically, PGD-AT, ARD, IAD, and AKD use student-only cross-entropy loss; TRADES employs KL divergence between clean and perturbed predictions; RSLAD aligns student outputs with teacher predictions on clean inputs; while AdaAD and IGDM match student and teacher predictions under shared perturbations. Our proposed SAAD framework follows the inner-outer structure described in Algorithm 2.

B.2 Selected Hyperparameters

We document the selected values of the clean distillation weighting coefficient β, which controls the strength of the clean KL divergence term in SAAD-C. To determine β, we perform a small-scale grid search aiming to maximize clean accuracy while maintaining comparable AutoAttack robustness to the SAAD. On CIFAR-10, we set $\beta = 0.2$ across all combinations of ResNet-18 and MobileNetV2 students with either `Bartoldson2024Adversarial` or `Gowal2021Improving` teachers. On CIFAR-100 and Tiny-ImageNet, we apply a uniform setting of $\beta = 0.5$.

B.3 Additional Notes on Figures and Tables

Figure 1a. All teacher models consistently achieve AA accuracies above 55%, while the strongest student reaches only 54%. This ensures that student underperformance cannot be attributed to weak teacher robustness, and instead reflects the quality of robustness transfer.

Figure 1b. Colored vertical bands indicate the same teacher architecture. For visual clarity, we apply a small horizontal jitter within each colored band, so left–right offsets inside a band are purely for visualization and have no meaning.

Table 1. All distillation results are obtained with RSLAD. Robust overfitting (RO) is defined as the drop from the best test PGD-20 accuracy during training to the final accuracy at epoch 200. With our schedule the peak typically occurs around epochs 100–110.

Figure 3b. For `Rebuffi2021Fixing`, we interpolate the teacher's soft distribution with the one-hot label as

$$\boldsymbol{t}_\alpha(\mathbf{x}) \;=\; (1-\alpha)\, f_T(\mathbf{x}) \;+\; \alpha\, \mathbf{e}_y, \quad \alpha \in [0,1], \tag{32}$$

where $f_T(\mathbf{x})$ is the teacher's probability vector and $\mathbf{e}_y$ is the one-hot vector for class y. When $\alpha = 0$ the target is the teacher distribution; when $\alpha = 1$ it is purely one-hot. Orange points correspond to different α; numeric annotations indicate the one-hot proportion.

Table 2. CIFAR-10 with a ResNet-18 student distilled from `Bartoldson2024Adversarial` using RSLAD. The first 90 epochs are a warm-up on the full 50,000 examples. At epoch 90, we partition the training set into TAS and Non-TAS using Equation 7; this split is then fixed for the remaining epochs.

Table 5. CIFAR-10 with a ResNet-18 student distilled from `Bartoldson2024Adversarial`. "Baseline" denotes plain RSLAD (no sample-wise weighting).

Table 7. CIFAR-10 with a ResNet-18 student distilled from the ERT `Chen2021LTD_WRN34_20`. Under this ERT, SAAD matches or slightly exceeds IGDM, which is consistent with our design goal: SAAD targets failure modes that arise under low transferability, and it does not aim to outperform existing methods when robustness transfer is already strong. Although one might hope to simply *pick an ERT* and avoid transferability issues, in practice it is rarely known in advance whether a given teacher will behave as an ERT or an IRT, and the teacher is often fixed (e.g., from a model zoo or an upstream system). SAAD therefore serves as a robust default: it consistently improves robustness when transferability is weak, while incurring no degradation when transferability is strong.

C Additional Experiments

C.1 Statistical Report

The main paper reports only mean results to preserve readability and avoid excessive font reduction in dense tables. Here, we provide full results, including standard deviations for three random seeds. See Table 12 for CIFAR-10 and Table 13 for CIFAR-100 and Tiny-ImageNet.

C.2 Impact of Sample-wise Weighting Hyperparameters

As shown in Table 14, increasing the β value enhances clean accuracy by placing greater emphasis on clean distillation. While small values of β lead to modest improvements in clean accuracy with minimal reduction in adversarial robustness, overly large β values cause substantial drops in both PGD and AutoAttack performance. This trade-off suggests that a moderate value offers the best balance between clean accuracy and robustness.

Table 12: Adversarial distillation results using two teacher and two student models on CIFAR-10. Clean, FGSM, PGD, C&W, and AA columns report accuracy (%) under each evaluation setting. Results are averaged over three random seeds with standard deviations.

Model	Method	Bartoldson2024Adversarial					Gowal2021Improving				
		Clean	FGSM	PGD	C&W	AA	Clean	FGSM	PGD	C&W	AA
ResNet-18	PGD-AT	84.27±0.10	52.10±0.34	42.34±0.09	42.29±0.07	40.85±0.14	84.27±0.10	52.10±0.34	42.34±0.09	42.29±0.07	40.85±0.14
	TRADES	82.70±0.10	57.14±0.48	48.81±0.52	48.08±0.54	46.46±0.57	82.70±0.10	57.14±0.48	48.81±0.52	48.08±0.54	46.46±0.57
	ARD	84.63±0.15	56.57±0.45	44.48±0.26	43.44±0.45	41.66±0.53	84.39±0.28	52.47±0.35	42.18±0.32	42.22±0.21	40.79±0.40
	IAD	84.43±0.25	56.64±0.20	44.82±0.15	43.47±0.11	41.80±0.15	84.28±0.29	52.25±0.13	42.16±0.24	42.19±0.09	40.70±0.10
	RSLAD	84.28±0.11	57.20±0.62	47.17±0.33	46.07±0.42	44.42±0.34	83.83±0.36	51.59±0.44	42.03±0.31	42.22±0.28	40.57±0.38
	AKD	84.62±0.14	56.29±0.08	44.42±0.09	43.43±0.12	41.79±0.12	84.32±0.19	52.13±0.34	42.38±0.07	42.44±0.17	40.95±0.06
	AdaAD	85.07±0.07	57.54±0.21	47.16±0.21	46.06±0.18	44.55±0.17	85.04±0.11	53.85±0.23	44.65±0.28	44.90±0.15	43.27±0.18
	IGDM	84.75±0.18	58.38±0.32	47.56±0.25	46.43±0.21	44.94±0.16	85.67±0.22	58.14±0.44	48.58±0.18	46.98±0.40	44.76±0.52
	SAAD-C	**85.54**±0.01	**61.92**±0.11	53.18±0.21	52.05±0.30	50.14±0.24	**86.39**±0.01	**60.55**±0.08	51.91±0.45	52.06±0.14	49.72±0.16
	SAAD	84.27±0.18	61.44±0.25	**53.39**±0.23	**52.39**±0.28	**50.34**±0.08	83.69±0.18	59.74±0.14	**52.89**±0.40	**52.36**±0.18	**50.35**±0.22
MobileNetV2	PGD-AT	83.52±0.19	54.92±0.24	44.90±0.43	44.29±0.18	41.54±0.22	83.52±0.19	54.92±0.24	44.90±0.43	44.29±0.18	41.54±0.22
	TRADES	81.79±0.46	56.50±0.19	49.90±0.07	47.54±0.26	46.50±0.14	81.79±0.46	56.50±0.19	49.90±0.07	47.54±0.26	46.50±0.14
	ARD	83.66±0.39	55.09±0.22	44.71±0.06	43.49±0.11	41.24±0.09	83.62±0.09	54.62±0.48	44.60±0.31	44.18±0.25	41.47±0.17
	IAD	83.85±0.27	55.69±0.10	44.98±0.10	43.62±0.04	41.35±0.05	83.63±0.12	54.81±0.17	44.73±0.13	44.19±0.17	41.51±0.03
	RSLAD	83.22±0.18	55.54±0.30	46.09±0.79	44.80±0.84	42.56±0.60	83.41±0.36	54.60±0.06	45.01±0.20	44.41±0.23	41.78±0.16
	AKD	83.60±0.42	55.13±0.21	44.31±0.11	43.29±0.21	41.03±0.21	83.54±0.03	54.79±0.11	44.83±0.12	44.19±0.18	41.55±0.05
	AdaAD	84.42±0.12	56.38±0.23	46.16±0.07	44.99±0.12	43.01±0.11	84.37±0.08	54.40±0.39	44.40±0.41	44.59±0.40	41.95±0.49
	IGDM	84.07±0.09	57.31±0.14	47.39±0.02	45.43±0.05	43.57±0.11	84.13±0.49	57.93±0.10	48.70±0.33	47.43±0.12	44.83±0.19
	SAAD-C	**85.16**±0.10	**60.53**±0.18	52.72±0.13	51.26±0.20	49.34±0.09	**84.81**±0.23	**58.17**±0.15	51.09±0.35	**50.45**±0.29	48.08±0.23
	SAAD	82.04±0.04	59.48±0.20	**53.69**±0.19	**51.68**±0.28	**49.88**±0.16	80.60±0.23	56.85±0.02	**51.78**±0.05	50.25±0.05	**48.29**±0.10

Table 13: Adversarial distillation results using ResNet-18 and PreActResNet-18 as student models for CIFAR-100 and Tiny-ImageNet, respectively, with the `Wang2023Better` teacher (WRN-70-16 for CIFAR-100 and WRN-28-10 for Tiny-ImageNet). Clean, FGSM, PGD, C&W, and AA columns report accuracy (%) under each evaluation setting. Results are averaged over three random seeds with standard deviations.

Method	CIFAR-100					Tiny-ImageNet				
	Clean	FGSM	PGD	C&W	AA	Clean	FGSM	PGD	C&W	AA
PGD-AT	56.17±0.20	24.74±0.16	19.65±0.18	19.79±0.20	18.66±0.18	45.71±0.33	15.75±0.05	11.67±0.43	11.82±0.45	10.91±0.40
TRADES	53.37±0.36	28.72±0.26	25.12±0.12	23.11±0.11	22.32±0.14	42.06±0.11	19.58±0.04	17.15±0.19	14.03±0.34	13.33±0.29
ARD	58.13±0.22	29.71±0.14	24.97±0.13	22.22±0.27	20.84±0.12	55.69±0.49	30.13±0.01	26.62±0.07	22.09±0.18	19.90±0.10
IAD	57.59±0.13	29.85±0.30	25.21±0.06	22.41±0.13	21.00±0.14	53.75±0.49	29.85±0.14	26.95±0.16	22.40±0.12	20.56±0.30
RSLAD	56.68±0.34	30.87±0.18	27.27±0.07	23.91±0.09	22.66±0.19	53.18±0.11	30.14±0.27	27.69±0.03	22.86±0.04	21.42±0.13
AKD	58.22±0.16	29.14±0.19	24.35±0.03	21.80±0.06	20.61±0.13	54.48±0.37	27.62±0.04	23.59±0.33	19.56±0.16	17.73±0.12
AdaAD	58.57±0.49	31.72±0.04	28.00±0.10	24.40±0.48	23.15±0.30	57.26±0.11	31.81±0.18	28.80±0.15	23.64±0.09	22.11±0.05
IGDM	56.36±0.32	32.95±0.13	29.68±0.08	25.91±0.12	24.81±0.28	57.15±0.08	31.98±0.18	29.02±0.17	23.94±0.04	22.52±0.11
SAAD-C	**59.57**±0.66	**36.05**±0.38	32.52±0.31	28.72±0.38	27.21±0.34	**57.33**±0.16	33.06±0.38	29.62±0.23	24.16±0.32	22.69±0.35
SAAD	59.11±0.28	36.01±0.13	**32.71**±0.21	**29.36**±0.16	**27.58**±0.16	57.16±0.36	**33.26**±0.32	**29.95**±0.13	**24.87**±0.29	**23.42**±0.23

C.3 SWA analysis

To ensure consistency, we apply the SWA technique across all adversarial distillation baselines and evaluate their performance under identical conditions. As shown in Table 15, applying SWA generally improves the robustness of existing AD methods to some extent. Nevertheless, both SAAD and SAAD-C consistently outperform all baselines, achieving the highest robustness. This indicates that the observed gains are not merely due to SWA but rather attributable to the design of our proposed distillation framework.

C.4 Compatibility of SAAD Weighting with Other AD Methods

SAAD's entropy-based weighting scheme can be seamlessly integrated into the outer minimization of existing AD methods, as it does not require any modification to their inner optimization or loss formulation. To verify this, we apply the SAAD weighting design on top of ARD, RSLAD, and AdaAD. All experiments follow the same setting in the main paper, i.e., distillation on CIFAR-10 with a ResNet-18 student and

Table 14: Effect of β on CIFAR-10 with the `Gowal2021Improving` teacher and ResNet-18 student.

β	Clean	FGSM	PGD	C&W	AA
0	83.69	59.74	52.45	52.36	50.35
0.05	84.72	60.13	52.77	52.32	50.19
0.1	85.49	59.70	52.45	52.18	50.14
0.15	85.91	60.21	52.45	52.21	49.90
0.2	86.39	60.55	51.91	52.06	49.72
0.25	87.93	59.40	49.18	49.36	47.02
0.3	88.13	57.79	44.88	45.17	42.65
0.5	88.19	56.44	42.65	43.06	40.78

Table 15: AD with SWA results on CIFAR-10 with the `Gowal2021Improving` teacher and ResNet-18 student.

Method	Clean	FGSM	PGD	C&W	AA
ARD	85.33	58.06	48.69	47.68	46.12
IAD	85.01	58.39	48.94	47.78	46.10
RSLAD	84.74	59.96	49.40	48.16	46.60
AKD	85.20	57.95	48.51	47.44	46.04
AdaAD	<u>86.11</u>	56.70	48.08	48.05	46.19
IGDM	86.09	58.85	50.13	49.83	48.18
SAAD-C	**86.39**	**60.55**	<u>51.91</u>	<u>52.06</u>	<u>49.72</u>
SAAD	83.69	<u>59.74</u>	**52.89**	**52.36**	**50.35**

Table 16: Compatibility of SAAD weighting with other AD methods. We report test accuracy (%) on CIFAR-10 with ResNet-18 student distilled from the `Gowal2021Improving` teacher.

Method	Clean	FGSM	PGD	C&W	AA
ARD	84.39	52.47	42.18	42.22	40.79
ARD + SAAD weighting	81.95	56.48	50.04	50.39	48.09
RSLAD	83.83	51.59	42.03	42.22	40.57
RSLAD + SAAD weighting	81.74	57.03	50.54	50.46	48.53
AdaAD	85.04	53.85	44.65	44.90	43.27
AdaAD + SAAD weighting	84.16	58.93	52.16	51.99	49.97
IGDM	85.67	58.14	48.58	46.98	44.76
SAAD-C	86.39	60.55	51.91	52.06	49.72
SAAD	83.69	59.74	52.89	52.36	50.35

Table 17: Throughput (epochs/hour) and peak GPU memory (MB) on CIFAR-10.

Method	Epochs/hour	Memory (MB)
ARD	17.76	4670
IAD	10.93	4670
RSLAD	10.93	4670
AKD	17.85	4670
AdaAD	1.23	26795
IGDM	1.17	26795
SAAD-C	4.73	26301
SAAD	4.83	26301

the `Gowal2021Improving` teacher. In Table 16, applying SAAD weighting to existing methods consistently improves robustness across all attacks. However, these variants remain slightly less effective than the full SAAD method.

C.5 Computational Resource

Table 17 reports throughput and peak memory on an NVIDIA A6000 (Ubuntu, Python 3.8, PyTorch). The larger memory usage for AdaAD/IGDM/SAAD is primarily due to backpropagating through the teacher during the inner maximization: IGDM and AdaAD run iterative inner loops with multiple teacher backward passes, whereas SAAD replaces this loop with a first-order approximation requiring only a single backward pass, yielding almost four times speedup while keeping peak memory comparable. By contrast, lighter baselines (ARD/AKD/RSLAD) avoid teacher backpropagation in the inner loop and therefore use much less memory and train faster. Importantly, the "without incurring additional computational cost" claim in the main text refers to SAAD's entropy-based weighting itself. Applied to gradient-free on teacher for inner-maximization methods such as ARD and RSLAD, the weighting improves robustness (see Table 16); the weighting does not increase memory or computation.

C.6 Impact of Underconfident Soft Labels in Adversarial Distillation.

One potential concern is that underconfident (high-entropy) soft labels may provide noisy supervision and thus harm adversarial distillation. To evaluate this, we analyze the teacher's prediction reliability in high-entropy regions. For each teacher, we distill a student using RSLAD on CIFAR-10. After training, we generate 10-step PGD adversarial examples on the student from the training set and compute the entropy of the teacher's output on these perturbed inputs. We then sort the samples by teacher-output entropy on

Table 18: Transfer adversarial accuracy (%) of teachers on high-entropy subsets. Even for high-entropy samples, teacher accuracy remains very high, indicating that underconfident labels still provide reliable supervision.

Teacher	Top 20%	Top 40%	Top 60%	Top 80%	Average
Teacher A[†]	94.50	97.06	97.99	98.46	98.73
Teacher B[‡]	97.06	98.50	99.00	99.24	99.39

[†]`Bartoldson2024Adversarial`, [‡]`Gowal2021Improving`

Table 19: Impact of ELS on SAAD. ELS leads to only minor changes across all metrics, suggesting that correcting underconfident predictions provides limited additional benefit.

Method	Clean	FGSM	PGD	C&W	AA
SAAD	83.69	59.74	52.89	52.36	50.35
+ ELS	83.86	59.78	52.75	52.22	50.07

student-generated adversarial inputs (transfer attacks) and report the teacher's prediction accuracy (transfer adversarial accuracy) on the top 20%, 40%, 60%, and 80% highest-entropy subsets. As shown in Table 18, transfer adversarial accuracy remains consistently above 94% even in high-entropy regions, suggesting that underconfident soft labels largely provide stable supervision rather than introducing significant noise. These results support the validity of entropy-based weighting: low-entropy, overconfident predictions are more indicative of high-variance, overfitting-prone samples, whereas high-entropy samples help stabilize training by reducing adversarial variance and enabling more effective knowledge transfer.

While transfer adversarial accuracy is high, it does not reach 100%, implying that a small fraction of teacher predictions deviate from the ground truth. To investigate whether explicitly correcting such deviations yields further benefits, we incorporate the Error-Corrective Label Swapping (ELS) technique proposed in DGAD (Park & Min, 2024), which swaps the true-label and max-logit probabilities for misclassified cases on both clean and adversarial examples. We apply ELS on top of SAAD and evaluate the resulting student on CIFAR-10 with a ResNet-18 student distilled from the `Gowal2021Improving` teacher. The results in Table 19 show that applying ELS yields only marginal changes across all evaluation metrics. While ELS explicitly corrects teacher outputs in misclassified cases, the gains remain negligible. This indicates that underconfident soft labels, despite their uncertainty, do not substantially degrade supervision, and explicit correction offers limited practical benefit. This confirms that the teacher's uncertainty signals are inherently informative rather than noisy, validating our approach of leveraging raw soft labels directly through entropy weighting.