8 Appendix

8.1 Proof of Theorem 1

Proof. Due to the symmetry of the data distribution in Eq. (5), if we want to get the error rate of a linear classifier $y = \text{sign}(x^\intercal \mathbf{w})$, we only have to calculate the integral of the Gaussian $x \sim \mathcal{N}(\boldsymbol{\mu}_*, \boldsymbol{\Sigma}_*)$ over a half-space segmented by the classifier. Suppose that the half-space that goes through the origin is given by

$$\Omega_+ = \{x \in \mathcal{R}^2 | x^\intercal \mathbf{w} \geq 0\}, \tag{17}$$

based on the probability density of the Gaussian, the integral can be written as:

$$P = \frac{1}{2\pi\sqrt{\boldsymbol{\Sigma}_*}} \int_{\Omega_+} \exp\left[-\frac{1}{2}(x-\boldsymbol{\mu}_*)^\intercal \boldsymbol{\Sigma}_*^{-1}(x-\boldsymbol{\mu}_*)\right] dx. \tag{18}$$

To solve the integral (18), we need the coordinate transform as:

$$\Omega = \mathcal{R}^1 \times [c, \infty), \quad c = -\frac{\boldsymbol{\mu}_*^\intercal \mathbf{w}}{\sqrt{\mathbf{w}^\intercal \boldsymbol{\Sigma}_* \mathbf{w}}}. \tag{19}$$

With the transformation (19), the integral (18) becomes:

$$P = \frac{1}{\sqrt{2\pi}} \int_c^\infty \exp\left(-\frac{1}{2}x^2\right) dx, \tag{20}$$

where the integral over Ω can be evaluated by the error function.

Due to the monotony of the error equation, Eq. (20) can reach the optima when c achieves its optima. According to the definition of c in Eq. (19), we have that the minima of c can be achieved at $\mathbf{w} = \boldsymbol{\Sigma}_*^{-1}\boldsymbol{\mu}_*$, which exactly corresponds to the the optima of objective (6). Thus, the Bayesian error rate of the classifier can be derived as:

$$err_* = 1 - \frac{1}{\sqrt{2\pi}} \int_c^\infty \exp\left(-\frac{1}{2}x^2\right) dx, \tag{21}$$

where $c = -\frac{\boldsymbol{\mu}_*^\intercal \mathbf{w}}{\sqrt{\mathbf{w}^\intercal \boldsymbol{\Sigma}_* \mathbf{w}}}$, $\mathbf{w} = \boldsymbol{\Sigma}_*^{-1}\boldsymbol{\mu}_*$, which completes the proof.

8.2 Proof of Propositions 1 and 2

Propositions 1 and 2 are directly referred from [13]. Corresponding proof can be found in [13].

8.3 Experiment Setting

Training: For both CIFAR10 and SVHN datasets, we used very similar training schemes for different models. The images were first normalized to [0, 1]. ResNet [11] and Wide ResNet [33] with different number of layers were employed as the basic feature extractor. The standard perturbation size for the adversarial training was both set to be $\epsilon = \frac{8}{255}$. For all the implemented methods, the

Attack: l_∞*-norm,* $\epsilon = 0.031$ ***Dataset:*** *CIFAR10*			
Model	Total Epoch	Learning Rate Decay	Dynamic ρ (τ in FAT)
Natural	120	Decay at {60, 70, 90} epoch, to {0.01, 0.001, 0.0005}.	N/A
Standard AT			N/A
FAT			τ: {0,1,2,3} increased at {50, 75, 90} epoch.
DAAT			ρ: {0.05, 0.2, 0.3} increased at {75, 90} epoch.
TRADES	85	Decay at {60, 70} epoch, to {0.01, 0.001}.	N/A
DAAT-TRADES			ρ: {0.05, 0.15, 0.2, 0.25} increased at {30, 50, 70} epoch.
MART	90	Decay at {60, 70} epoch, to {0.01, 0.001}.	N/A
DAAT-MART			ρ: {0.05, 0.15, 0.25} increased at {20, 40, 60} epoch.
Attack: l_∞*-norm,* $\epsilon = 0.031$ ***Dataset:*** *SVHN*			
Model	Total Epoch	Learning Rate Decay	Dynamic ρ (τ in FAT)
Natural	120	Decay at {60, 70, 90} epoch, to {0.001, 0.0001, 0.00005}.	N/A
Standard AT			N/A
FAT			τ: {0,1,2,3} increased at {30, 50, 70} epoch.
DAAT			ρ: {0.05, 0.15, 0.2, 0.25} increased at {30, 50, 70} epoch.
TRADES	85	Decay at {30, 60, 70} epoch, to {0.001, 0.0001, 0.00005}.	N/A
DAAT-TRADES			ρ: {0.05, 0.15, 0.2, 0.25} increased at {30, 50, 70} epoch.
MART	90	Decay at {60, 70} epoch, to {0.001, 0.0001}.	N/A
DAAT-MART			ρ: {0.05, 0.15, 0.25} increased at {20, 40, 60} epoch.

Table 2: Implementation and training details for different models.

adversaries were initialized with the uniform random start, and the maximum PGD step $T = 10$, step size $\alpha = \frac{\epsilon}{4}$ to make sure that the worst-case adversarial examples could be obtained. All the DNN models for CIFAR10 were optimized with SGD with the initial learning rate of 0.1, the momentum of 0.9, and the weight decay of 0.0002. For SVHN, the initial learning rate was set to be 0.01. The batch size during training was set to be 128 for both datasets. For better performance, learning rate decay was applied at different epochs. Note that, for the implementation of some baselines, we did not directly use the learning rate decay scheme as introduced in the original paper because it may cause severe overfitting problem after the learning rate was decayed. More training details for different models can be found in Table 2.

Attacking: We used prevalent attack types, including FGSM, PGD-T, C&W [5] optimized by PGD, to evaluate the proposed DAAT method. The black-box attack was also employed for verification. The perturbation size of different attacks for both datasets was set to be $\epsilon = \frac{8}{255}$. To obtain stronger adversaries, the step size during the attacking was set to be $\alpha = \frac{\epsilon}{10}$ which was much smaller than that during training, and the number of perturbation steps was also increased to $T = 20, 40$ respectively to achieve different strength of attacks.

8.4 Investigation on the Calibration Network c

The performance of $c(\cdot)$ plays a critical role in the natural generalization of the robust model. It can be imagined that if the calibration network does not generalize well on the clean examples (e.g., not well-trained on the natural data, or trained with the adversarial data), it is not capable to provide instructive information to determine a proper perturbation size that does not hurt the natural accuracy too much while keeping the robustness. To investigate the effect

Attack: *l_∞-norm FGSM, $\epsilon = 0.031$* ***Dataset:*** *CIFAR10*		
Model	Nat. Acc.	Adv. Acc.
DAAT calibrated by c	88.31	**64.14** $\pm$ 0.17
DAAT calibrated by f	85.94	63.51 $\pm$ 0.23
DAAT $g + c$	**89.62**	58.64 $\pm$ 0.16

Table 3: Performance of different models to investigate the influence of the calibration network. The first two rows are DAAT models calibrated by the naturally trained c and adversarially trained f, respectively. The third row is the DAAT model whose classifier is replaced by c.

of the calibration network, we first evaluate the model's natural accuracy and robustness with a $c(\cdot)$ which is well-trained on the natural data. As in Table 3, the DAAT model calibrated by c significantly outperforms the DAAT model calibrated by f. A rational reason is that the classifier f is trained with the adversarial data, thus the adjustment provided by f is based on the decision boundary of the adversaries which is not helpful to the natural accuracy. The evidence consists in the model at the third row whose classifier is replaced by c. The calibration network c empowers the DAAT model with higher natural accuracy but with remarkably lower robustness, which indicates that c is capable to provide knowledge about the better natural decision boundary and does not fit well on the adversaries. Although we need to train an auxiliary network c per epoch, we can rescue the natural accuracy by a large margin at a small cost (about ten seconds per epoch compared with the standard AT). Even more to the point, FAT also utilizes the adversarially trained classifier as the model in the third row to generate friendly adversaries in practice, which explains why DAAT generally outperforms FAT on natural accuracy.

8.5 Learning Curves

In this subsection, we plot more learning curve pairs between the baseline models and DAAT-enhanced models for a better demonstration of the performance improvement. As we can see from Fig. 7, the natural accuracy of the proposed DAAT method can usually converge faster and better than the baselines. Meanwhile, the DAAT models can also achieve higher robustness. Note that, the models shown in Fig. 7a-7f are all trained from scratch. However, as in Fig. 7f, the MART model on the SVHN dataset cannot be properly trained at the first few epochs, which may lead to a worse convergence. Thus, for better performance, we employed a natural pretrained model to fine-tune the MART and DAAT-MART models. The result is shown in Fig. 7g. As we can see from the figure, the convergence of training is much faster and better, and the DAAT-MART model can still outperform the baseline.

8.6 Extended Experiments

Network: ResNet18, Attack: l_∞***-norm,*** $\epsilon = 0.031$***, Dataset: CIFAR10***								
Methods \ Attacks	Clean	White-box FGSM	PGD-20	PGD-40	C&W-20	C&W-40	Black-box (PDG-20) Non-Robust	AT
TRADES $\beta = 1.0$	85.40%	63.04%	49.22%	47.26%	48.43%	46.79%	84.17%	62.47%
DAAT-TRADES $\beta = 1.0$	**86.81%**	**63.62%**	**49.47%**	**47.54%**	**48.87%**	**47.21%**	**85.62%**	**64.08%**
Network: ResNet18, Attack: l_∞***-norm,*** $\epsilon = 0.031$***, Dataset: SVHN***								
Methods \ Attacks	Clean	White-box FGSM	PGD-20	PGD-40	C&W-20	C&W-40	Black-box (PDG-20) Non-Robust	AT
TRADES $\beta = 1.0$	92.69%	71.11%	**54.54%**	**52.20%**	50.33%	48.29%	87.54%	61.38%
DAAT-TRADES $\beta = 1.0$	**93.74%**	**71.55%**	54.47%	52.16%	**50.76%**	**48.67%**	**88.69%**	**62.08%**

Table 4: Robust accuracy of TRADES and DAAT-TRADES $\beta = 1.0$ models on CIFAR10 and SVHN under different attacks.

TRADES $\beta = 1.0$: In the TRADES model, the trade-off parameter β is for adjusting the attention between accuracy and robustness. We have reported the experimental results of TRADES $\beta = 6.0$ in Section 6. In this subsection, we further investigate the performance of TRADES $\beta = 1.0$. As shown in Table 4, the natural accuracy of TRADES model grows with the decreasing β, while the robustness decreases. However, no matter what the β is, DAAT can consistently help TRADES improve the natural accuracy while keeping or even boosting the robustness.

Wide ResNet-32-10: In Table 5, we also employ a larger network architecture (e.g., Wide ResNet-32-10) for evaluation. We use the same test setting as in [35]. As we can see from the table, DAAT model achieves remarkable robustness improvement especially on stronger attacks compared with AT and FAT.

Comparison with [1] [1] indeed share a similar motivation as our work, however the methodology is different. Compared with the simple addition and subtraction operations to ϵ_is, the proportional dynamic update rule of DAAT and calibration net guarantee a better performance. In our paper, we mainly compared with the published works. Here, we also compare with [1] in Table 6 for completeness. DAAT can outperform IAAT [1] in terms of both natural and robust accuracy.

8.7 Attackable and Robust Samples

In Fig. 4, it is obvious that for the DAAT model, there are not only a large number of adversaries which are on the surface of the perturbation ball but also a considerable number of adversaries which are close to the perturbation ball centre. The former set of natural examples is more attackable since the prediction can be changed with a small size of perturbation. On the contrary, the latter set of natural examples is more robust since it takes a larger perturbation

Attack: *l_∞-norm, $\epsilon = 0.031$* ***Dataset:*** *CIFAR10*				
Model	Clean	FGSM	PGD-20	C&W-30
AT [35]	87.30%	56.10%	45.80%	46.80%
FAT [35]	89.34%	65.52%	46.13%	46.82%
DAAT	**89.56%**	**66.99%**	**50.53%**	**51.04%**

Table 5: Evaluation of Wide ResNet-32-10 on CIFAR10.

Attack: *l_∞-norm, $\epsilon = 0.031$* ***Net:*** *ResNet18*				
Model	Clean	PGD-10	PGD-100	PGD-1000
IAAT [1]	87.26%	43.08%	41.16%	41.16%
DAAT	**88.31%**	**48.10%**	**46.05%**	**46.01%**

Table 6: Compared with IAAT [1] on CIFAR10.

to alter the prediction. In Fig. 8, we demonstrate some samples of the natural examples in different classes from the attackable set and robust set, respectively. From Fig. 8, we can find that the attackable examples generally contain more complex background and blurred textures, which means that they are more difficult to be correctly classified and are more easily to be attacked. The robust natural examples usually have a clean background and a clear outline where the intra-class characteristics are better preserved. Thus, larger perturbations are supposed to be applied to change the prediction.

8.8 Extra Time Consumption

We evaluate the extra time consumption of our calibration scheme in Table 7. We can find that DAAT takes approximately 20 minutes more time for retraining c with natural data, which is acceptable compared with the whole adversarial training process. Although by naturally training both g and c can achieve 0.37% improvement on natural accuracy, it can cause a $6\times$ slowdown. Thus, we believe our method is a satisfying trade-off between the performance and computational cost.

Attack: *l_∞-norm,* ***Net:*** *ResNet18,* ***Dataset:*** *CIFAR10*			
Calibration Network	Clean	Robust	Extra Time
Naturally trained $g + c$	**88.67%**	48.26%	$\sim$ 120min
Naturally trained c	88.31%	**48.89%**	$\sim$ **20min**

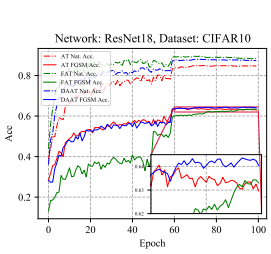
Table 7: Extra time consumption of different calibration schemes.

8.9 Compared with Non-adversarial Training Defenses.

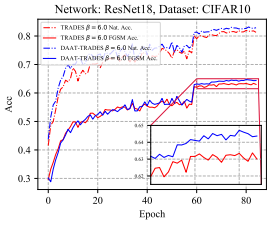
Adversarial training is not the only way to improve the model's adversarial robustness. Some methods attempt to robustify the model by smoothing the geometry of the classification landscape [25] or network pruning [17], etc. In Table 8, we compare DAAT with multiple non-adversarial training defenses. We can find that our method can significantly outperform these methods in terms of both robustness and accuracy.

Attack: *l_∞-norm,* ***Net:*** *ResNet18,* ***Dataset:*** *CIFAR10*		
Method	Clean	Robust
CURE [25]	83.11%	38.50%
DNR [17]	87.32%	40.41%
DAAT	**88.31%**	**48.89%**

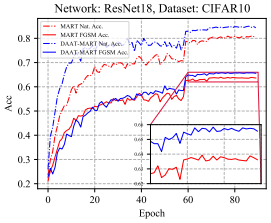
Table 8: Comparison with non-adversarial training methods.



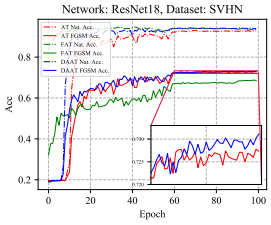
(a) CIFAR: AT/FAT/DAAT.



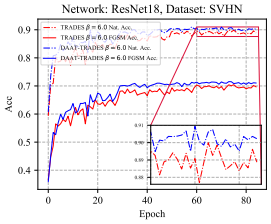
(b) CIFAR: TRADES/DAAT-TRADES.



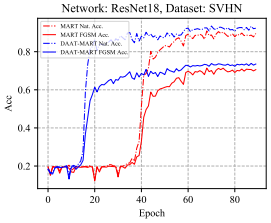
(c) CIFAR: MART/DAAT-MART.



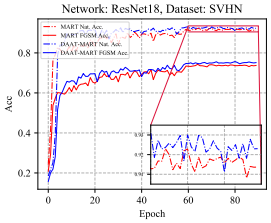
(d) SVHN: AT/FAT/DAAT.



(e) SVHN: TRADES/DAAT-TRADES.



(f) SVHN: MART/DAAT-MART.



(g) SVHN: Pretrained MART/DAAT-MART.

Fig. 7: Learning curve pairs between the baseline models and DAAT-enhanced models.

(a) Attackable examples.

(b) Robust examples.

Fig. 8: Samples of attackable and robust natural examples. The l_∞ distance between the attackable natural examples and their corresponding adversaries is less than 0.0001, while it is exactly the preset perturbation size 0.031 for the robust natural examples.