Table 1: Additional experiments in terms of classification Accuracy (%).

(a) Results of WRN28-10@CIFAR and ResNet101@ImageNet.

	CIFAR10		CIFAR100		ImageNet
Noise Rate	0.4	0.8	0.4	0.8	0.0
ERM	75.41	30.00	55.68	13.99	78.2
SELF[24]	**93.34**	67.41	72.48	42.06	-
Ours	93.23	**80.13**	**72.60**	**44.17**	**78.7**

(b) Parameters sensitivity to different datasets and noise rates.

	CIFAR10 (80% Noise)			CIFAR100 (40% Noise)		
α	0.8	0.9	0.95	0.8	0.9	0.95
Fix E_s=60	75.60	**78.58**	75.44	70.36	**71.38**	68.57
E_s	40	60	80	40	60	80
Fix α=0.9	68.27	78.58	**78.65**	70.30	**71.38**	67.32

We thank the reviewers for their valuable comments. We summarize major concerns from reviewers and respond to them appropriately as follows. We will add suggested experiments, references, and fix typos in the updated version.

To Reviewer #1: *Q1:* Insights on setting the hyper-parameter E_s. *A1:* The optimal value of E_s is related to the begining epoch of overfitting. Since the baseline TRADES uses a step learning-rate-decay schedule and we observe that it starts to overfit around the epoch of the first learning rate decay (the 75-th epoch, see the blue curve in Figure 1), we simply set the E_s to a slightly smaller value 70. However, we also find that our method is not sensitive to E_s: using $\mathrm{E}_s = 60$ has similar performance as $\mathrm{E}_s = 70$ (see Figure 1, the red and green curves). This phenomenon is consistent with Table 2b in the main body, where we show our method is not sensitive to various hyper-parameters (see also Table 1b).

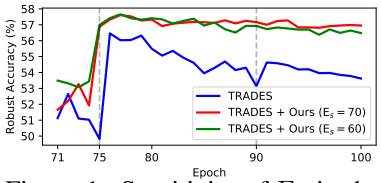

Figure 1: Sensitivity of E_s in the adversarial learning.

Q2: Extra experiments on other model structures. *A2:* Besides ResNet34@CIFAR10/100 and ResNet50@ImageNet in main body, we also report the results of WRN28-10@CIFAR10/100 and ResNet101@ImageNet in Table 1a, where our approach outperforms ERM and the reported numbers of SELF[24] in most entries, sometimes by a large margin.

To Reviewer #2: *Q1:* Claim of applicability to any deep supervised learning task; validity of premise in new tasks. *A1:* We point out a potential misunderstanding about our paper, where we did NOT claim that our method is applicable to *any* deep supervised learning task. Though not for any task, our method is robustly and reliably effective for *a wide range of tasks*: classification with label noise, selective classification, adversarial learning, vanishing double descent, etc. As a new task, we run experiments on OOD generalization task (see Table 2 and A4@R#3), where ours has better performance. The premise that the model can guess the right predictions follows the observation that deep models fit the clean samples first, which is justified by [14,18,32] and our extensive experiments in the above-mentioned broad tasks. *Q2:* Extreme failure cases. *A2:* In Figure 3 of main body, ERM performs better only in the extreme case where the model capacity is more than $10\times$ smaller than standard ResNet-18, where ERM's test accuracy is poor ($\leq 78\%$, i.e., significant underfitting occurs). For other extreme cases when the data is complex or when there are few training samples, though our method might fail, ERM will perform poorly too, due to the information-theoretical limit. We argue that studying models with enough capacity, realistic (amount of) input data, and reasonable performance (e.g., $\geq 90\%$ test accuracy) might be of more interests to the community, where our method consistently outperforms ERM.

To Reviewer #3: *Q1:* Cost of maintaining probability vectors. *A1:* The cost is not high. Take the large-scale ImageNet as an example. The storage of such vectors in single precision format for the entire dataset requires $1.2 \times 10^6 \times 1000 \times 32$ bit ≈ 4.47GB, which is acceptable since modern GPUs usually have no less than 11GB memory. Moreover, the vectors can be stored on CPU memory or even disk and loaded along with the images to further reduce the cost.

Table 2: Average Accuracy (%) on CIFAR10-C at various corruption levels.

Level	1	3	5
ERM	88.44	77.26	58.91
Ours	**89.41**	**78.83**	**60.77**

Q2: Robustness to other corruption percentages and datasets. *A2:* In Table 1b, we conduct extra experiments on two datasets with different noise rates. The results indicates that our approach is robust to the values of hyper-parameters in various settings.

Q3: Additional large-scale experiments. *A3:* Please refer to Table 1a and A2@R#1 for additional results on ImageNet.

Q4: OOD generalization. *A4:* In Table 2, we report the average accuracy using ResNet-34 on CIFAR10-C over 15 corruptions. Under various corruption levels, our method consistently outperforms ERM by a considerable margin, indicating that self-adaptive training provides implicit regularization for OOD generalization.

To Reviewer #4: *Q1:* Results on uncorrupted CIFAR. *A1:* On CIFAR10/100, the test accuracy is 95.32%/78.42% for ERM, and 95.17%/78.69% for ours.

Q2: Contradiction between L30 and L103. *A2:* We will make it clear: in the first few iterations, though the model learns to fit the correct labels in a *progressive* manner (as in L30), its predictions are very unstable, especially in the very beginning of the training procedure (as in L103). The instability is due to the use of regularization such as data augmentation (as in L104).

Q3: Further investigation of sample weights. *A3:* The minimum value of w_i is not bounded by α due to the moving-average scheme that accumulates the predictions. Following the procedure in Figure 5 of main body, we display the average sample weights in Figure 2, where the white areas represent the case that no sample lies in the cell. We see that the weights on the diagonal are higher.

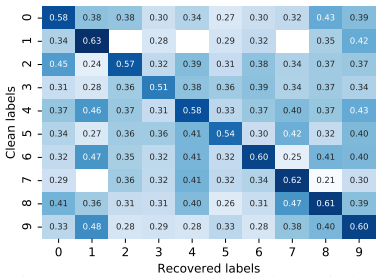

Figure 2: Average sample weights w_i under various labels.