Supplementary Material for Paper 3233: Towards Fast Training of More Robust Models Against Adversarial Attacks

Acc.(%)	clean	FGSM	CE1000	MI	Ori-CW	DF-l2
R-MC-LA+	92.8	75.6	61.4	65.5	65.4 / 88.2	77.8
TRADES	84.9	61.1	56.4	58.0	81.2	81.6

Table 1: The classification accuracy against various white-box attacks on CIFAR10. We use training budget $\epsilon_x = 4$.

1. Introduction

First, we provide two more experiments on CIFAR10. Then, we provide the results on CIFAR100 dataset. Next, to validate our motivation, we compare the gradient magnitude of different models. At last, we provide another heuristic solution to the problem of generating adversarial labels.

2. Two More Experiments on CIFAR10

2.1. Against Other White-box Attacks

Table 1 shows the accuracy against other white-box attacks. We compare with the winner, TRADES [6], in NeurIPS18 Adversarial Vision Challenge. We use the default settings in the Cleverhands package to generate the attacks. “MI” refers to the MI-FGSM method [2]. “Ori-CW” refers to the original CW attack [1], and the two numbers refer to two common sets of hyper-parameters: {const=100, confid=0, lr=1e-1, max iter=1e2} / {const=100, confid=0, lr=1e-2, max iter=1e3}. “DF-l2” refers to the DeepFool attack with l_2-norm [4]. We see that our models generally outperform the baseline, except against the DeepFool attack. We note that our network is smaller, and our training method is an-order-of-magnitude faster[1].

2.2. Effect of Number of Training Images

We vary the number of training images per class. The results are shown in Table 2. This is aligned with the claim in [5] that adversarial training requires more data than regular training.

Acc.(%)	clean	FGSM	CE20	CE100	CW100
R-MC-LA (5K)	90.7	69.6	55.3	53.8	52.8
R-MC-LA (2K)	85.6	56.1	42.8	41.1	40.2
R-MC-LA (0.5K)	73.3	33.7	25.1	24.5	24.0

Table 2: The classification accuracy of R-MC-LA models, trained using different data size on CIFAR10. The number in the parenthesis means the number of images per class. We use $\epsilon_x = 8, \beta = 9$.

Acc.(%)	clean	FGSM	CE20	CE100	CW100
R-MC-LA ($\epsilon_x = 8$)	**68.7**	30.5	23.2	22.7	20.6
R-MC-LA+ ($\epsilon_x = 8$)	66.2	31.3	23.1	22.4	20.0
R-MC-LA9 ($\epsilon_x = 8$)	**68.7**	33.7	23.1	22.0	20.1
R-MC-LA9+ ($\epsilon_x = 8$)	68.2	**36.9**	**26.7**	**25.3**	22.1
Madry*	61.9	28.8	23.7	23.4	**24.5**

Table 3: The classification accuracy of R-MC-LA models against white-box attacks on CIFAR100. The models are trained using different perturbation budget. We use $\beta = 11$.

3. CIFAR100 Dataset

In this section we report the results against white-box attacks on CIFAR100 [3] dataset. It has 100 classes, 50K training images and 10K test images. In addition to the basic R-MC-LA models, we also try a slightly modified version, denoted by R-MC-LA9. Specifically, when generating the adversarial label, we distribute the ϵ_y to the top-9 non-groundtruth classes with largest loss, instead of to all the non-groundtruth classes. This modification brings several percentage gain. The results are shown in Table 3. We see that our models outperform the state-of-the-art on clean image and against FGSM, and perform comparably on multi-step attacks. We hypothesize that CIFAR100 is more difficult than CIFAR10 and SVHN for adversarial training because of much fewer images per class.

4. Gradient Magnitude

Table 4 provides the gradient magnitude results on three datase. For CIFAR10 and SVHN, madry's method is trained

[1] [6] uses WRN-34, and 20-step PGD attack during training.

	CIFAR10			SVHN			ImageNet		
	undefended	madry	ours	undefended	madry	ours	undefended	madry	ours
max	349.9	1.8	1.6	267.9	15.3	3.4	0.48	0.004	0.084
mean	2.3	0.022	0.0098	0.77	0.12	0.022	0.0044	0.000038	0.00014

Table 4: Comparison of gradient magnitude, $\|\nabla_x L(x, y; \theta)\|_2^2$, of undefended model, madry's model, and our model, on three datasets (averaged over all test / validation images). The gradient is taken w.r.t. the original image range [0, 255], instead of [-1, 1], so the numbers are 127.5 times smaller than Table 2 in the paper. For CIFAR10 and SVHN, all models are run **three times** to average out randomness.

using PGD7-2 with budget 8 pixels. For ImageNet, madry's method is trained using PGD10-3 with budget 16 pixels. From the table, we see that adversarially trained models, including madry's and ours, leads to much smaller gradient magnitude (one or two order-of-magnitude), compared to undefended models. This correlates with our hypothesis that there may be a link between small gradient magnitude and adversarial robustness. Besides, we see that madry's and ours are comparable (particularly on CIFAR10 and SVHN). Note that although gradient magnitude confidently distinguish undefended and adversarially trained models, it is not a precise indicator of robustness between adversrially trained models. Currently, only from the gradient magnitude, we cannot confidently tell which one is more robust. So we have to compare and report their accuracy. Finding precise indicator for adversarial robustness is an active and unsolved research topic.

Acc.(%)	clean	FGSM	CE20	CE100	CW100
R-MC-LA (main)	90.8	69.3	54.6	52.9	51.9
R-MC-LA+ (main)	91.0	70.3	**57.5**	**55.2**	**53.8**
R-MC-LA (sup)	90.2	70.9	53.2	51.1	49.9
R-MC-LA+ (sup)	**91.5**	**71.4**	57.2	54.1	51.5

Table 5: The classification accuracy of the proposed R-MC-LA models under various white-box attacks on CIFAR10. The source models are trained using two solutions for generating the adversarial labels. We use $\beta = 9$ and $\epsilon_x = 8$ during training and in evaluation.

5. Another Solution to Generating Adversarial Labels

In this section, we provide another heuristic solution to the problem of generating adversarial labels

$$\max_{\|y'-y\|_\infty \le \epsilon_y} L(x, y'; \theta). \tag{1}$$

Here the original groundtruth y is a one-hot vector, i.e., $y_c = 1$ and $y_k = 0, k \neq c$.

In the main paper, the heuristic is to distribute the ϵ_y to non-groundtruth classes while keeping the share of the MC class very small. Specifically, the share is proportional to the gradient of each class subtracted by the minimal gradient (which corresponds to the MC class). Here, we propose another simpler heuristic, which is that the share is directly proportional to the respective gradient. We can then easily obtain the formula

$$y'_k = \frac{\epsilon_y v_k}{\sum_{k \neq c} v_k}, k \neq c. \tag{2}$$

Note that we use v_k to denote $\nabla_{y_k} L(x, y; \theta)$ for short. By using the following condition

$$y_c \geq \beta \max_{k \neq c} y_{k'}, \tag{3}$$

we can solve for the largest budget ϵ_y

$$\epsilon_y \leq \frac{1}{1 + \frac{\beta v_{max}}{\sum_{k \neq c} v_k}}. \tag{4}$$

Note that this solution is an exact application of gradient ascent

$$y'_k = y_k + \alpha \nabla_{y_k} L(x, y; \theta), k \neq c, \tag{5}$$

where

$$\alpha = \frac{1}{\sum_{k \neq c} v_k + \beta v_{max}}. \tag{6}$$

We favor the solution used in the main paper over this solution (2) for two reasons. Firstly, from the optimization point of view, the solution in the main paper leads to a higher (better) objective value for the maximization problem (1), because it distributes more shares to the classes with larger gradient. Secondly, the solution in the main paper leads to a smaller y'_{MC} (proof given below). Note that the adversarial image used in training is generated by the MC targeted attack. Using a smaller y'_{MC} will suppress the network to predict large probability on the MC class, thus better focusing on predicting large probability on the groundtruth class. The results achieved by these two solutions are shown in Table 5, where "main" refers to using the solution in the main paper, and "sup" refers to using the solution (2) in the supplementary material. We can see that "main" is slightly better than "sup" against multi-step PGD attacks.

Lastly we provide the proof. From the solution in the main paper, we have

$$y'_{MC,main} = \frac{\gamma}{\sum_{k \neq c} v_k - (n-1)(v_{MC} - \gamma) + \beta(v_{LL} - v_{MC} + \gamma)}. \tag{7}$$

From the solution (2), we have

$$y'_{MC,sup} = \frac{v_{MC}}{\sum_{k \neq c} v_k + \beta v_{LL}}. \tag{8}$$

The sufficient and necessary condition of

$$y'_{MC,main} < y'_{MC,sup} \tag{9}$$

is

$$(n-1)v_{MC} + \beta v_{MC} < \sum_{k \neq c} v_k + \beta v_{LL}, \tag{10}$$

which is obviously true. This is because the left is smaller than the right on both the first term and the second term respectively.

References

[1] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *IEEE Symposium on Security and Privacy*, 2017. 1

[2] Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Xiaolin Hu, Jianguo Li, and Jun Zhu. Boosting adversarial attacks with momentum. In *Computer Vision and Pattern Recognition*, 2018. 1

[3] Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, 2009. 1

[4] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. Deepfool: a simple and accurate method to fool deep neural networks. In *Computer Vision and Pattern Recognition*, 2016. 1

[5] Ludwig Schmidt, Shibani Santurkar, Dimitris Tsipras, Kunal Talwar, and Aleksander Madry. Adversarially robust generalization requires more data. In *Advances in Neural Information Processing Systems*, 2018. 1

[6] Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric P Xing, Laurent El Ghaoui, and Michael I Jordan. Theoretically principled trade-off between robustness and accuracy. In *International Conference on Machine learning*, 2019. 1