Supplementary Material for
Investigating Catastrophic Overfitting in Fast Adversarial Training: A Self-fitting Perspective

A. Experiment details.

FAT settings. We train ResNet18 on Cifar10 with the FGSM-AT method [3] for 100 epochs in Pytorch [1]. We set $\epsilon = 8/255$ and $\epsilon = 16/255$ and use a SGD [2] optimizer with 0.1 learning rate. The learning rate decays with a factor of 0.1 at the 80th and 90th epochs. To better study CO, we use zero initialization to generate adversarial samples, and weight decay is set to 0 to reproduce CO stably. The batch size is 128. Images are padded with 4 pixels and randomly cropped and flipped horizontally.

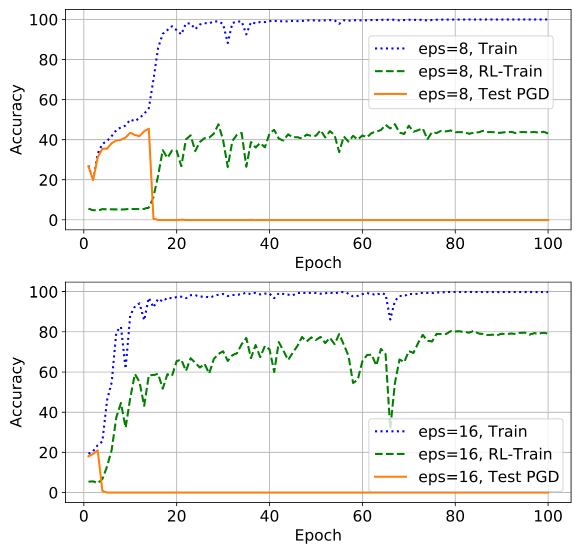

Figure 1. FGSM-AT training with different ϵ on Cifar10 using WideResNet28-10. Catastrophic overfitting happens at 15th epoch for $\epsilon = 8/255$ and 4th epoch for $\epsilon = 16/255$.

PGD-AT details in further discussion. There is only a little difference between the settings of PGD-AT and FAT. PGD-AT uses a smaller step size and more iterations with $\epsilon = 16/255$. The learning rate decays at the 75th and 90th epochs. The robust accuracy during training of different settings is shown in Fig. 3.

B. Experiments on WideResnet28-10.

This section reports experiments on WideResNet28-10 about self-fitting. Compared to ResNet18, WideResNet28-10 has more parameters and therefore has a stronger learning capability.

Training curve. Fig. 1 shows the training curve of WideResNet28-10 on Cifar10 with FGSM-AT method. The training setting also follows Appendix A. Catastrophic overfitting happens earlier than ResNet18. After CO, the random-label FGSM accuracy also increases quickly with training accuracy, suggesting that self-information dominates the classification.

Probability changes with attack step size's increase. Fig. 2 visualizes that when the step size of FGSM perturbation gradually increases, how output probability of the network in the corresponding classes. The model is trained on Cifar10 using WideResNet28-10. When the step size increases, the probability firstly decreases, meaning that the perturbation can fool the network, then increases, meaning that the network can recognize the self-information in the perturbation when the step size is large enough.

Table 1. Accuracy drop of different networks trained on Cifar10 using WideResNet28-10. Compared to the network without CO, the network with CO has a large drop in FGSM accuracy while little change in clean accuracy.

	CLEAN	FGSM	PGD
W/O CO, NOT PRUNED	76.4%	51.2%	45.4%
W/O CO, PRUNED	-19.9%	-19.5%	-16.2%
WITH CO, NOT PRUNED	79.2%	99.5%	0.0%
WITH CO, PRUNED	-10.4%	-92.4%	0.1%

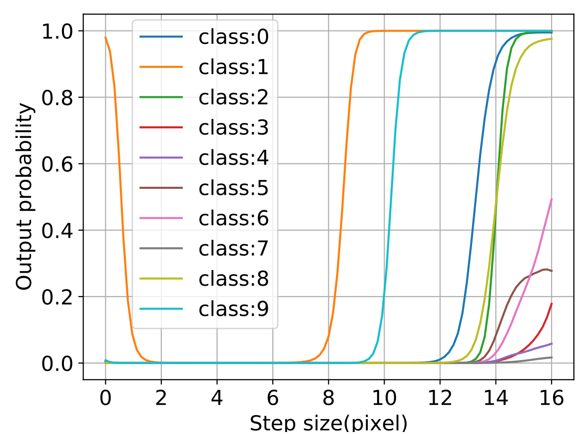

Figure 2. Visualization of the probability of the network in the corresponding classes when the step size of FGSM perturbation gradually increases. The original class is class 1. The network is trained with $\epsilon = 16/255$.

Channel variance in descending order. Fig. 4 shows

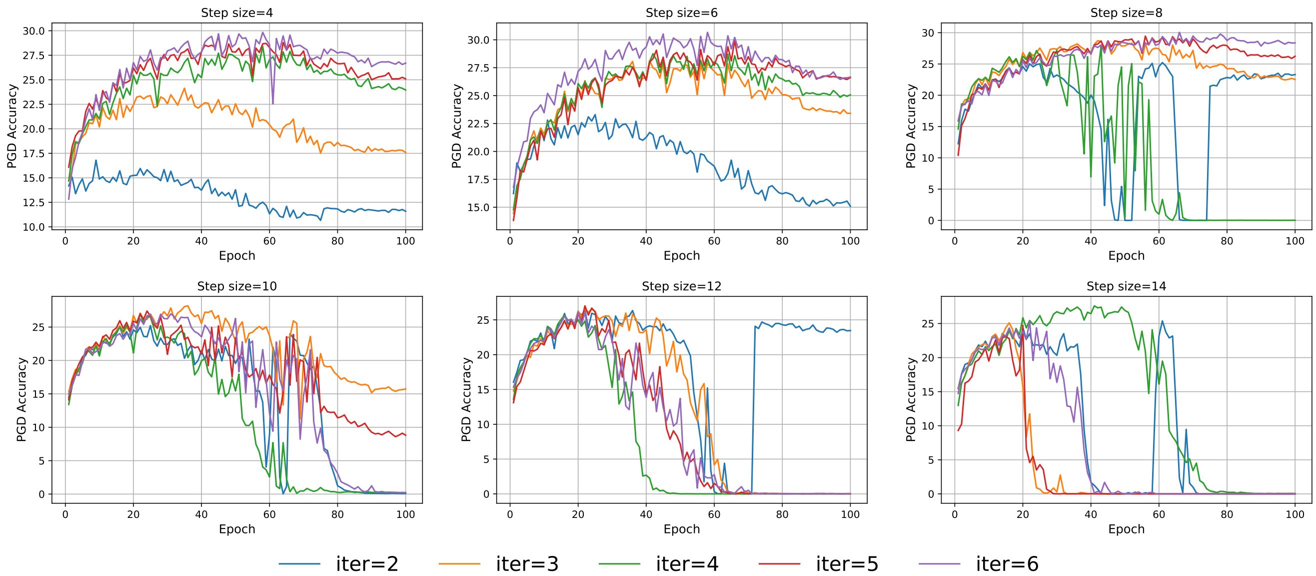

Figure 3. Training curves of multi-step AT with different iterations and step sizes. PGD accuracy is calculated using a PGD20 attack with 3 random starts.

the variance values in descending order of networks with and without the CO on WideResNet28-10. The features after the first layer of WideResNet28-10 have only 16 channels. After CO, some channels become dominant to recognize self-information, thus having a larger variance. While some channels for data-information become unimportant and "dead".

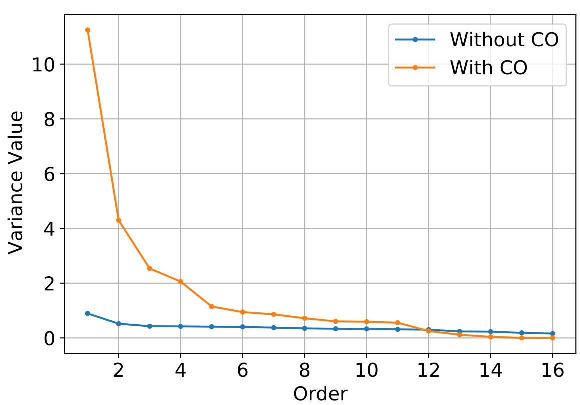

Figure 4. The variance values in descending order of networks with and without the CO on WideResNet28-10. The network with CO has a larger maximum variance value and more zero variance channels.

Accuracy of pruned network. Tab. 1 shows the accuracy change of different setting after pruning, which is for WideResNet28-10 trained on Cifar10. Only one channel of the first layer with the highest variance is pruned. The network without CO has a similar drop in all accuracy after pruning. In contrast, after pruning, the network with CO has a drop of 92.4% in FGSM accuracy, while the clean accuracy only decreases by 10.4%.

References

[1] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 8024–8035, 2019. 1

[2] Ning Qian. On the momentum term in gradient descent learning algorithms. *Neural networks*, 12(1):145–151, 1999. 1

[3] Eric Wong, Leslie Rice, and J. Zico Kolter. Fast is better than free: Revisiting adversarial training. In *International Conference on Learning Representations (ICLR)*, 2020. 1