Project Report for Adversarial Attacks Against Neural Networks

Sishuo Chen 1700012765
School of Electronic Engineering and Computer Science, Peking University
chensishuo@pku.edu.cn

Abstract

This report is a description of my implementation for several classical adversarial attack methods against neural networks, as a practice project for the Intro to AI course in Peking University, spring 2021. For the white-box attack part, I implement FGSM (Goodfellow et al., 2015) and PGD (Madry et al., 2018). For the black-box attack part, I implement the transfer-based attack boosted by momentum(MI-FGSM) proposed by Dong et al. (2018) . In addition to reaching the required ASR on MNIST and CIFAR10 benchmarks, I also explore the effect of several important factors in adversarial attacks, including the step size and the number of iterations for PGD, the substitute-target model architecture difference and the decay parameter for momentum.

1 Introduction

Deep neural networks are vulnerable to adversarial examples. In this project, we are required to implement classical white-box and black-box attack approaches and use them to attack pretrained neural networks for image classification.

According to the instruction, the hyper-parameter limits and performance bars are listed below. For the MNIST task, the limit on the perturbation size is $\epsilon = 0.3$ (L_∞ metric) for both white-box and black-box attack, the inner iteration for PGD is 10, and the target model is a small CNN. For the CIFAR10 task, the limit on the perturbation size is $\epsilon = 0.031$ (L_∞ metric) for both white-box and black-box attack, the inner iteration for PGD is 10, and the target model is a preActResnet18(He et al., 2016). The requirement for attack success rate(ASR) is listed in Table 1.

Dataset	Attack	ASR bar
MNIST	FGSM	$\approx$ 65%-75%
	PGD	$\approx$ 98%
	Black-box	40%
CIFAR10	FGSM	$\approx$ 80%-90%
	PGD	$\approx$ 100%
	Black-box	70%

Table 1: ASR Performance Bar

2 White-box Attacks

2.1 Annotations

I implement three classical attack approaches and apply them to attack the pretrained models in a non-target way.

Fast Sign Gradient Method (FGSM) (Goodfellow et al., 2015) can be formulated as:

$$x^* = x + \epsilon \cdot \operatorname{sign}\left(\nabla_x L\left(f_\theta(x), y\right)\right) \tag{1}$$

Projected Gradient Descent(PGD)(Madry et al., 2018) is an iterative version of FGSM. It can be described as the following formula(α is the step size hyper parameter).

$$\boldsymbol{x}^{t+1} = \Pi_\epsilon\left(\boldsymbol{x}^t + \alpha \cdot \operatorname{sign}\left(\nabla_{\boldsymbol{x}} tL\left(f_\theta\left(\boldsymbol{x}^t\right), y\right)\right)\right) \tag{2}$$

MI-FGSM (Dong et al., 2018)is an improved version of PGD boosted by momentum. It can be described as the following formula(α is the step size hyper parameter and μ is the momentum decay hyper parameter).

$$\begin{aligned} \boldsymbol{x}^{t+1} &= \Pi_\epsilon\left(\boldsymbol{x}^t + \alpha \cdot \operatorname{sign}\left(\boldsymbol{g}_{t+1}\right)\right) \\ \boldsymbol{g}_{t+1} &= \mu \boldsymbol{g}_t + \frac{\nabla_{\boldsymbol{x}} tL\left(f_{\boldsymbol{\theta}}\left(\boldsymbol{x}^t\right), y\right)}{\left\|\nabla_x tL\left(f_{\boldsymbol{\theta}}\left(\boldsymbol{x}^t\right), y\right)\right\|_1} \end{aligned} \tag{3}$$

2.2 Experimental Setting

For the MNIST task, the limit for the perturbation size ϵ is 0.3, and the ϵ limit for the CIFAR10 task is 0.031. For both tasks, the inner iteration rounds for PGD and MI-FGSM T is 10. The step size hyper parameter α is 0.1 for MNIST and 0.01 for CIFAR10. The momentum decay μ for MI-FGSM is 1.0 following Dong et al. (2018).

2.3 Main Results

The attack success rate results are listed in Table 2. It's obvious that the iterative-way proposed in PGD improves ASR results, but the momentum trick helps little under the white-box setting.

Task	Method	ASR %
MNIST	FGSM	64.18
	PGD	99.63
	MI-FGSM	98.43
CIFAR10	FGSM	88.25
	PGD	100.00
	MI-FGSM	100.00

Table 2: Whitebox Attack Results

2.4 Ablation Study

2.4.1 the Step-size of PGD

In the baseline experiments above, the step size α for PGD is set to 0.1 for MNIST and 0.01 for CIFAR10. To explore the effect of α, I show the relationship between the success rate and α in Figure 1 and 2. For the MNIST task, I test α = 0.0125, 0.025, 0.05, 0.10, 0.20, 0.40, 0.80. For the CIFAR10 task, I test α = 0.00125, 0.0025, 0.005, 0.010, 0.020, 0.040, 0.080. ϵ and T are kept the same as baseline.

$\alpha^* = \epsilon/T$ is a meaningful threshold for α. When $\alpha \leq \alpha^*$, the exploration is limited in the ϵ ball naturally with no need for clamping. When $\alpha > \alpha^*$, perturbation for some pixels may surpass the ϵ limit and clamping is needed in the iteration progress. In the baseline experiments, I set $\alpha \approx 3\alpha^*$(α^* = 0.03 for MNIST and 0.0031 for CIFAR10). Experiments show that my choice reaches the best performance compared with other options for α. When α is too small, the actual perturbation is limited far below the requirement threshold, so ASR is low ; when α is too large, too many pixels reach the ϵ ball limit and then are clamped in the iterative progress, which also does harm to ASR.

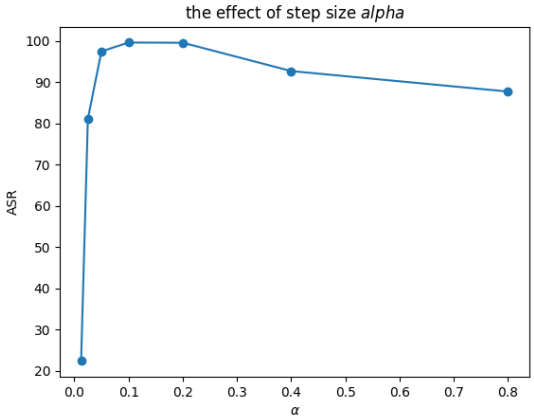

Figure 1: the relationship between the success rate and α(MNIST task)

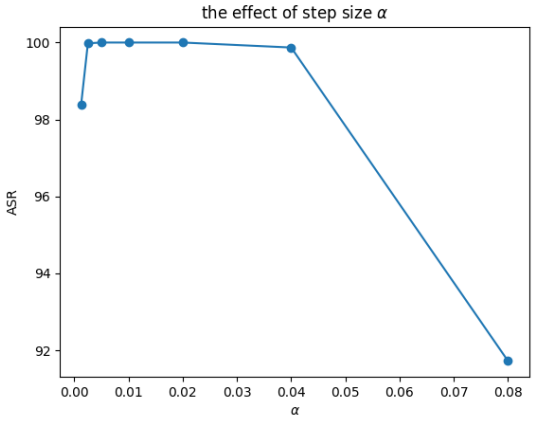

Figure 2: the relationship between the success rate and α(CIFAR10 task)

2.4.2 the Number of Iterations for PGD

In the baseline experiments, the number of inner iterations T for PGD is set to 10 for both tasks. To explore the effect of α, I show the relationship between the success rate and T in Figure 3. I test T in range $[1, 20]$, and ϵ, α are set the same as the baseline.

Results show that the ASR result is positively correlated with T. At the point around T=10, ASR reaches the highest point and almost keep the same when Tcontinues to grow. It's possibly because that when T is big enough, the perturbation reaches the surface of the ϵ ball and changes little in later iterations.

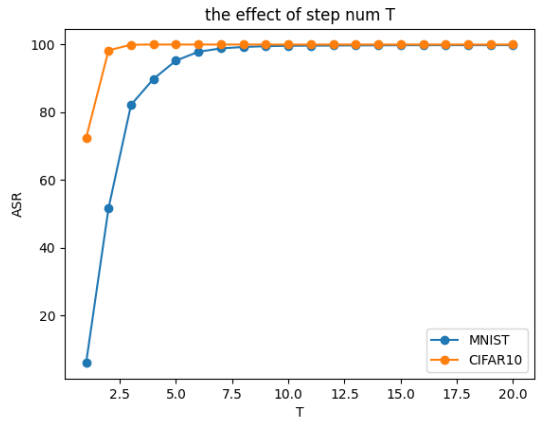

Figure 3: The relationship between the success rate and iterations)

3 Black-box Attacks

3.1 Experimental Setting

For simplicity, I only implement the transfer-based black-box attack approach. It means that the gradients will be gained by a substitute model trained on the same training dataset by myself. Except that, the settings is the same with that in 2.2. In the baseline experiment, the substitute model architectures are the same with those of the target models.

3.2 Main Results

The ASR results of the black-box baseline are listed in Table 3. Under the black-box setting, both PGD and MI-FGSM show significant preformance drop compared tp the white-box setting, but MI-FGSM surpassed PGD, showing the transferability gain provided by momentum.

Task	Method	ASR %
MNIST	FGSM	64.48
	PGD	97.33
	MI-FGSM	97.75
CIFAR10	FGSM	58.01
	PGD	86.25
	MI-FGSM	89.43

Table 3: Blackbox Attack Results(with the substitute models sharing the same architectures with the targets retrained by me)

3.3 Ablation Study

3.3.1 Substitute-target Model Architecture Difference

In the baseline experiments, it's supposed that the target model architecture is known and the same models can be trained. It's an unpractical condition in actual applications. To explore the effect of the architecture of the substitute model, I change the architecture of the substitute model and retrain them to apply attacks. For MNIST, I choose LeNet (Lecun et al., 1998). For CIFAR10, I choosed the VGG11 (Simonyan and Zisserman, 2015) and the after-activation version of Resnet18 (He et al., 2016). Hyper parameters setting is the same as baseline.

Task	Method	Substitute Model Architecure		
		small CNN	Lenet	
MNIST	FGSM	64.48	40.64	
	PGD	97.33	48.65	
	MI-FGSM	97.75	57.42	
		preActResnet18	Resnet18	VGG11
CIFAR10	FGSM	58.01	52.26	46.29
	PGD	86.25	72.59	60.36
	MI-FGSM	89.43	78.66	62.50

Table 4: ASR with different substitute models

The results are listed in Table 4. It's evident that all three attack approaches show performance drop when the substitute model architecture is different from the target model to attack. MI-FGSM shows the best transfer ability, proving the effect of the momentum trick.

3.3.2 Momentum Decay

The momentum decay hyper parameter μ is the key factor in MI-FGSM. When μ is larger, the more history-step influence will be gain by an update step. When $\mu = 0$, MI-FGSM degrades to PGD. In the baseline experiments, I set $\mu = 1.0$ following the baseline experiments in Madry et al. (2018). To explore the effect of μ, I show the relation of ASR and μ in Figure 4. Results show that the relation of ASR and μ is an inverted U-shape curve. Model architecture and ϵ, α are kept the same as baseline. The performance reaches the highest point at around $\mu = 0.6 - 0.8$ and drops a little when μ continues to grow.

4 Conclusion

In this project, I implement three classical adversarial approaches: FGSM, PGD and MI-FGSM. I apply them to attack pretrained models for image

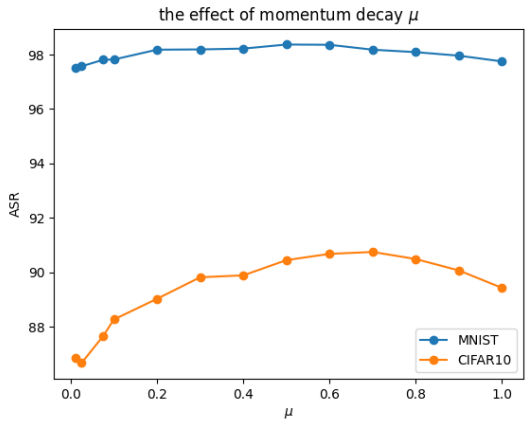

Figure 4: the relationship between the success rate and momentum decay μ)

classification under both white-box and black-box settings. In addition to the baseline implementation and experiments , I also discuss the effect of several important factors, including the step size and iteration rounds for PGD, the substitute model architecture in transfer-based attacks and the momentum decay for MI-FGSM.

References

Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. 2018. Boosting adversarial attacks with momentum. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 9185–9193. IEEE Computer Society.

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. 2015. Explaining and harnessing adversarial examples. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 770–778. IEEE Computer Society.

Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. 2018. Towards deep learning models resistant to adversarial attacks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Karen Simonyan and Andrew Zisserman. 2015. Very deep convolutional networks for large-scale image recognition. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.