Is wider model more robust to adversarial attacks

Anonymous Author(s)
Affiliation
Address
email

Abstract

Deep neural networks are vulnerable to adversarial attacks, where the raw data is perturbed with human-imperceptible, carefully crafted noises. Previous works show that a neural network with a larger model capacity can achieve better robustness performance. However, how the neural network's width affects its robustness remains elusive. This paper investigates the relationship between the neural network's width and its adversarial robustness on three benchmark datasets. We observe that a wider model may not necessarily lead to better robustness. Furthermore, we identify three cases that affect the model's robustness: the overfitting effect, strong attacking effect, and large output space effect.

1 Introduction

Though deep neural networks have demonstrated high accuracy performance in various fields [1, 2], they are shown to be vulnerable to adversarial attacks. A human-imperceptible but carefully crafted noise can easily fool the neural networks to make wrong predictions with high confidence [3, 4]. However, sometimes neural networks are required to have robust performance over adversarial attacks for security concerns.

One commonly accepted point of view in adversarial attack is that adversarial training requires a neural network to have a larger capacity to achieve better robustness [5]. Madry [6] provides an intuitive explanation of the complexity of the decision boundary. It is the presence of possible adversarial examples that makes the decision boundary more complicated. A robust classification requires a neural network with a larger capacity to learn the more complicated decision boundary. Increasing the neural network's width is a common way to increase the model capacity. However, it remains elusive how the neural network's width affects its robustness to adversarial attacks.

This paper examines the relationship between neural network's width and robustness on three image classification datasets using three white-box attacking methods. Our experimental results suggest that wider models are not necessarily more robust to adversarial attacks. Moreover, we identify three scenarios where the increased network width does not lead to better robustness.

2 Related Works

2.1 Adversarial Robustness

There are extensive research work on analyzing the influencing factors of neural network's adversarial robustness. Guo. et al. [7] find that the model architecture is one of the crucial factors and propose a family of robust architectures (RobNets) that are more resilient to adversarial perturbations. Meanwhile, adversarial training strategy has been proved to be an effective way to the neural network's adversarial robustness [4, 8]. Moreover, Madry. et al. [6] observe that increasing the network's capacity can effectively increase the robustness under different training strategies against

Submitted to 34th Conference on Neural Information Processing Systems (NeurIPS 2020). Do not distribute.

perturbations from different attack methods. Our research fixes the model architecture and training scheme to focus on the relationship between adversarial robustness and model capacity.

2.2 Wide Neural Network

Wide neural networks have been used extensively due to their high accuracy performance [9]. Previous empirical results suggest that wide networks are essential to achieve high performance under adversarial training [10, 11]. In this project, we vary the neural networks' width to change the model capacity. Unlike what has been observed by Madry [6], we identify three cases where smaller capacity models outperform the larger capacity models.

3 Method

3.1 Adversarial Robustness

Let us consider a classification task with K classes using neural networks f of a fixed architecture. The neural network classifiers take the form $h_\theta(x) = \arg\max_{y\in[K]} f(x;\theta)_y$, where $f(x;\theta) \in \mathbb{R}^K$ is a probability vector of scores assigned to candidate labels y, given the example x and parameters θ. We measure the adversarial robustness of the neural network f by its accuracy under adversarial examples.

$$\mathcal{E}(\mathcal{D}_{\text{adv}}(\epsilon)) = \mathbb{E}_{(x,y)\subset\mathcal{D}_{\text{adv}}(\epsilon)}\left[\mathbb{I}(f(x;\theta)=y)\right], \tag{1}$$

where $\mathbb{I}$ is the indicator function, ϵ is the attack strength, and $\mathcal{D}_{\text{adv}}(\epsilon)$ is a test dataset consisting of adversarial examples.

3.2 Adversarial Examples

Given a test dataset $\mathcal{D}_{\text{test}}$, we construct the adversarial test dataset $\mathcal{D}_{\text{adv}}(\epsilon)$ by perturbing every data point in the test dataset $\mathcal{D}_{\text{test}}$ using the following white-box attack methods.

i) **FGSM** (Fast Gradient Sign Method) [12]: FGSM is a one-step attack method, which aims to fool the trained model by adding a small perturbation to the original input based on the loss gradients. Let us denote the cross-entropy loss as l_{ce} and let $\nabla_x l_{\text{ce}}(f(x;\theta),y)$ be the gradient with respect to the input x, the adversarial example has the form:

$$x_{\text{adv}} = x + \epsilon * \text{sign}\left(\nabla_x l_{\text{ce}}(f(x;\theta),y)\right) \tag{2}$$

ii) **PGD** (Projected Gradient Descent) [6]: PGD is a multi-step attack method, which adopts an iterative approach to find the perturbation that maximizes the loss of a model for a given input x. Starting from a random perturbation, PGD takes a gradient step towards the direction of the greatest loss and projects the adversarial example to a subspace of allowed adversarial examples. Let $\mathcal{S}$ be a function of ϵ and represent the set of allowed perturbation. Let $\Pi_{x+\mathcal{S}(\epsilon)}$ represent the projection onto the set of allowed adversarial examples and α be the step size of the gradient update. The update rule for the PGD can be expressed as:

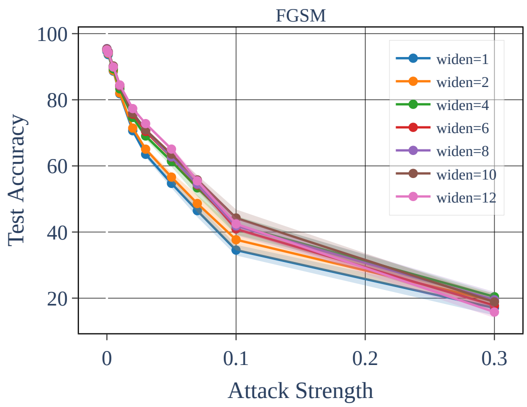
$$x^{t+1} = \Pi_{x+\mathcal{S}(\epsilon)}\left(x^t + \alpha * \text{sign}\left(\nabla_x l_{\text{ce}}(f(x;\theta),y)\right)\right) \tag{3}$$

iii) **GN** (Gaussian Noise)[13]: Gaussian Noise attack randomly samples a noise from Gaussian Distribution $\mathcal{N}(\mathbf{0},\sigma^2 I)$ and adds that to the clean data. Let the attack strength $\epsilon = \sigma$. The adversarial examples are sampled from the following distribution:

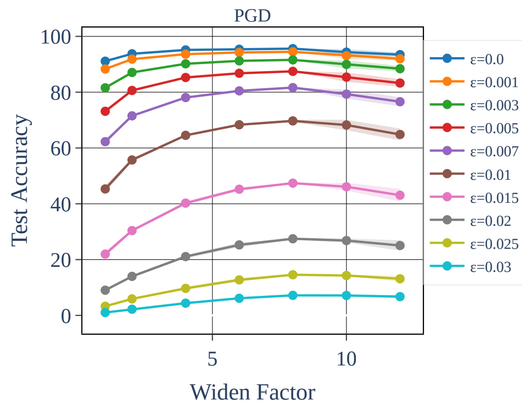
$$x_{\text{adv}} \sim \mathcal{N}(x, \epsilon^2 I) \tag{4}$$

3.3 Model

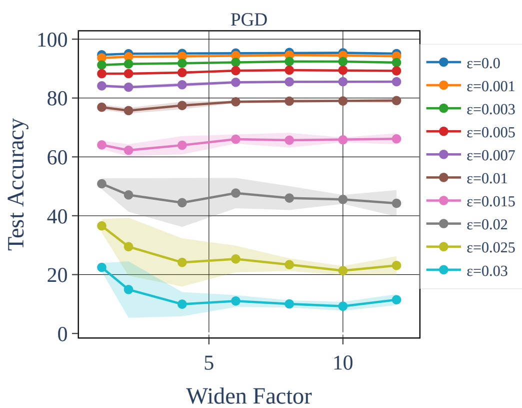
To investigate the relationship between adversarial robustness and model width, we fix the model architecture and training procedure. We use the wide residual networks (WRN) [9] with a fixed depth of 16 throughout our experiments and vary the widen-factor from [1, 2, 4, 6, 8, 10, 12]. We train the model parameters by minimizing the cross-entropy loss $\mathbb{E}\left[l_{\text{ce}}(f(x;\theta),y)\right]$ over all training examples, where $l_{\text{ce}} = \sum_{i=1}^K \mathbb{I}(y=i)\ln{}^1\!/_{p_i} = \ln{}^1\!/_{p_y}$. After training, we evaluate the models' robustness under the three adversarial examples with different attack strengths.



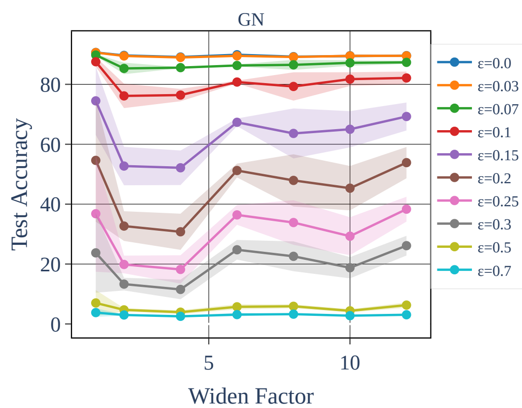
(a) Test accuracy on FashionMNIST under FGSM attack



(b) Test accuracy on CIFAR10 under PGD attack



(c) Test accuracy on FashionMNIST under PGD attack



(d) Test accuracy on EMNIST under GN attack

Figure 1: (a) Larger models are more robust to adversarial attacks. (b) Overfitting effect: there is a decrease in test accuracy as we increase the model size due to overfitting. (c) Strong attacking effect: overfitting becomes more severe as we increase the attack strength. (d) Large output space effect: the model with a larger output space is more sensitive to adversarial attacks.

4 Experimental Results

We evaluate the adversarial robustness of WRN on three image classification benchmark dataset, namely, EMNIST [14], FasionMNIST [15], and CIFAR10 [16]. For each dataset, we evaluate the models with seven different widths under three adversarial attack methods. For each attack method, we use nine different attack strengths. We provide more dataset, training, and evaluation details in Appendix A. We repeat all experiments three times with different random seeds and report the mean and standard deviation. We provide all training results in Appendix C.

4.1 Larger models lead to better adversarial robustness

In general, we observe that larger models are more robust to adversarial attacks across all strength levels on FashionMNIST and CIFAR10. As shown in Figure 1a, the models with larger widen-factors achieve higher test accuracies across all attack strengths, suggesting that larger models are more robust to adversarial attacks. These results agree with Madry's observation [6]. However, we also observe cases where the larger model fails to achieve high test accuracy, as explained in the following sections.

4.2 Overfitting effect

We observe that larger models may have worse adversarial robustness due to overfitting. Figure 1b shows that the model achieves a higher adversarial accuracy under the PGD attack as we increase the widen-factor up to 8, which agrees with the result in Section 4.1. However, the performance becomes worse as we keep increasing the widen-factor. It suggests that larger models can easily fit into noise when the number of parameters is beyond the model capacity sufficient for the task. It is because that the landscape of the loss surface becomes less smooth, and the model becomes more sensitive to perturbations. This observation agrees with the theoretical analysis based on local Lipschitzness in Boxi [17]. Moreover, we notice that the overfitting effect is more noticeable under strong attack strengths, which we explore in Section 4.3.

4.3 Strong attacking effect

We observe that the attack strength can affect the relationship between the model's adversarial robustness and model width. As shown in Figure 1b, when the attack strength is small, the robustness of WRN tends to increase as the widen-factor increases, which matches the general observation in Section 4.1. In contrast, for a large attack strength, the model achieves lower test accuracy as we increase the widen-factor. Moreover, as we increase the attack strength, we observe a sharper decrease in test accuracy for the model with a small width. It suggests that a strong attack strength can amplify the overfitting effect.

Figure 2 in the Appendix provides an intuitive explanation for the strong attacking effect based on the models' decision boundaries under different model capacities and attack strength levels. The model with a larger width can learn more complicated decisions boundaries as shown in Figure 2b and Figure 2d. With small attack strength, the complicated decision boundaries can handle noise correctly as the adversarial examples are close to the training examples. However, if the noise gets further away from the cluster mean, even a small change can result in drastic prediction changes. In contrast, Figure 2a and 2c show that the model with a smaller width is less sensitive to the adversarial noise under strong attack because of a simper decision boundary.

4.4 Large output space effect

Compared with other datasets, we find that models trained on EMNIST are more sensitive to adversarial attacks. As shown in Figure 1d, the change in test accuracy is more sensitive to the change in attack strength. Moreover, the model achieves lower accuracy under high attack strength. Notice EMNIST has 47 classes, yet FashionMNIST and CIFAR10 have only ten classes. In a larger output space, we believe that it is easier to generate a perturbation to cross the decision boundary, causing the model to be more susceptible to attacks. However, what surprises us is that both the small width model and large width model achieve better performance than the medium width model. It suggests that there may also be a double descent phenomenon for the adversarial attack [18].

5 Conclusion

This paper shows that a larger neural network can help with the model robustness performance in general. Nevertheless, the model's adversarial robustness still heavily depends on the dataset and the attacking methods. We identify three cases that affect the model robustness, summarized as the overfitting effect, strong attacking effect, and large output space effect. Our work suggests that it is not enough to increase the model width to get a more robust neural network. We may also need a stronger regularization to overcome the overfitting and make the decision boundary more smooth. Moreover, when we design a robust neural network, we need to consider both different attacking methods and different attack strengths.

References

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems* (F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, eds.), vol. 25, pp. 1097–1105, Curran Associates, Inc., 2012.

[2] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," 2019.

[3] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," 2015.

[4] A. Kurakin, I. Goodfellow, and S. Bengio, "Adversarial machine learning at scale," 2017.

[5] B. Wu, J. Chen, D. Cai, X. He, and Q. Gu, "Do wider neural networks really help adversarial robustness?," 2021.

[6] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," 2019.

[7] M. Guo, Y. Yang, R. Xu, Z. Liu, and D. Lin, "When nas meets robustness: In search of robust architectures against adversarial attacks," 2020.

[8] C. Xie, M. Tan, B. Gong, A. Yuille, and Q. V. Le, "Smooth adversarial training," 2020.

[9] S. Zagoruyko and N. Komodakis, "Wide residual networks," 2017.

[10] J. Zhang, X. Xu, B. Han, G. Niu, L. Cui, M. Sugiyama, and M. Kankanhalli, "Attacks which do not kill training make adversarial learning stronger," 2020.

[11] V. Feldman, K. Ligett, and S. Sabato, eds., *Algorithmic Learning Theory, 16-19 March 2021, Virtual Conference, Worldwide*, vol. 132 of *Proceedings of Machine Learning Research*, PMLR, 2021.

[12] X. Yuan, P. He, Q. Zhu, and X. Li, "Adversarial examples: Attacks and defenses for deep learning," 2018.

[13] N. Ford, J. Gilmer, N. Carlini, and D. Cubuk, "Adversarial examples are a natural consequence of test error in noise," 2019.

[14] G. Cohen, S. Afshar, J. Tapson, and A. van Schaik, "EMNIST: an extension of MNIST to handwritten letters," *CoRR*, vol. abs/1702.05373, 2017.

[15] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms," 2017.

[16] A. Krizhevsky, "Learning multiple layers of features from tiny images," 2009.

[17] B. Wu, J. Chen, D. Cai, X. He, and Q. Gu, "Do wider neural networks really help adversarial robustness?," 2020.

[18] P. Nakkiran, G. Kaplun, Y. Bansal, T. Yang, B. Barak, and I. Sutskever, "Deep double descent: Where bigger models and more data hurt," 2019.

[19] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems 32*, pp. 8024–8035, Curran Associates, Inc., 2019.

[20] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, "Random erasing data augmentation," in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2020.

[21] H. Kim, "Torchattacks: A pytorch repository for adversarial attacks," *arXiv preprint arXiv:2010.01950*, 2020.

A Experimental Setup

A.1 Dataset

i) **EMNIST** [14]: It is an extension of MNIST consists of 47 classes of 28x28 images of both digits and letters. We use the balanced EMNIST, where each class has 2400 training examples and 400 test examples.
ii) **Fashion MNIST** [15]: It is a dataset similar to MNIST consisting of 10 classes of 28x28 clothing images. It includes 60,000 training examples and 10,000 test examples.
iii) **CIFAR** [16]: It is a standard image dataset with two tasks: one coarse-grained over 10 classes (CIFAR10) and one fine-grained over 100 classes (CIFAR100). We evaluate our method on CIFAR10.

A.2 Training

For each dataset, we optimize the model using the stochastic gradient descent (SGD) optimizer with a batch size of 128 and a weight decay of 5e-4. We apply a step-wise learning rate decay schedule which multiplies the current learning rate by 0.1 at the specific epochs. For CIFAR10, we train the model for 300 epochs with an initial learning rate of 0.1 and decay the learning rate at [150, 225] epochs. For FashionMNIST and EMNIST, we train the model for 150 epochs with an initial learning rate of 0.03 and decay the learning rate at [75, 110] epochs. We repeat all experiments three times with different random seeds. All models are trained in Pytorch [19] based on the code from [20].

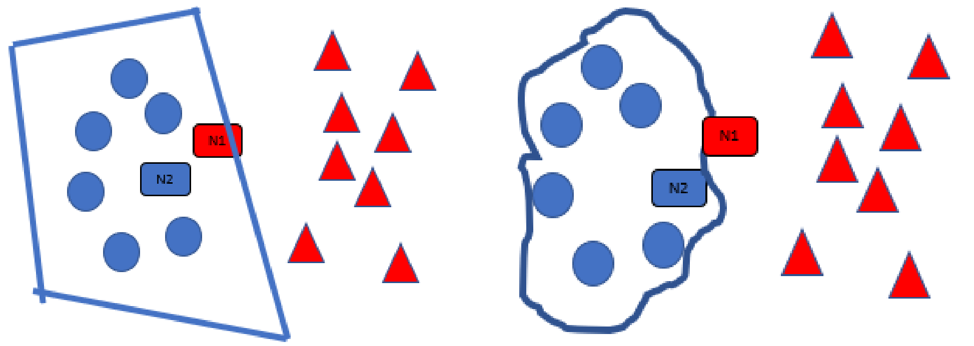
A.3 Evaluation

We evaluate the adversarial robustness of the trained model using three attack methods with nine different attack strengths based on code from [21]. The nine attack strengths cover the cases where the model is under mild attack and severe attack. We summarize them as follows.

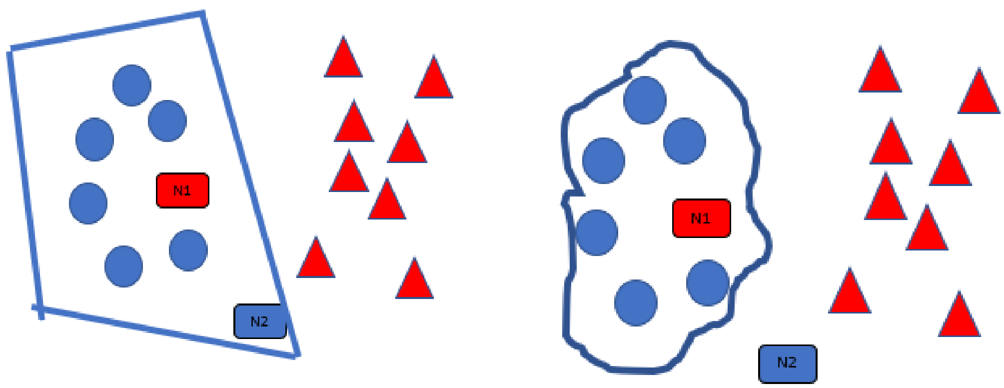
i) FGSM: we use attack strengths $\epsilon \in \{0.001, 0.003, 0.005, 0.007, 0.01, 0.015, 0.02, 0.025, 0.03\}$.
ii) PGD: we use attack strengths $\epsilon \in \{0.001, 0.005, 0.01, 0.02, 0.03, 0.05, 0.07, 0.1, 0.3\}$.
iii) GN: we use attack strengths $\epsilon \in \{0.001, 0.003, 0.005, 0.007, 0.01, 0.015, 0.02, 0.025, 0.03\}$.

Note: For PGD, we use step size $\alpha = 1/225$ and perform 40 iterative updates.

B Decision Boundary Visualization



(a) small width model with small attack strength (b) large width model with small attack strength

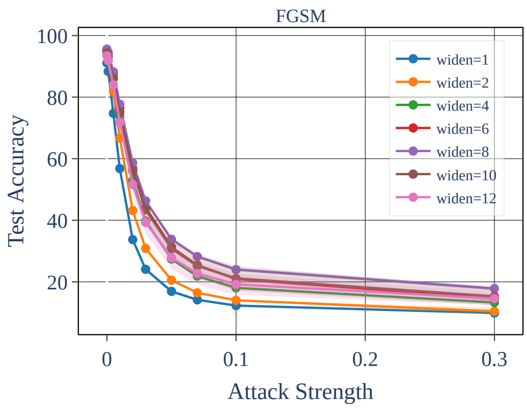


(c) small width model with large attack strength (d) large width model with large attack strength

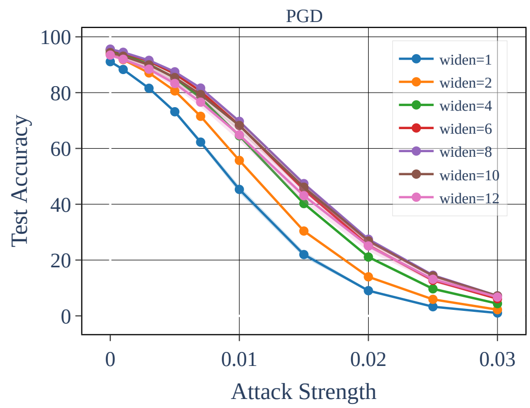
Figure 2: Visualization of decision boundaries of small and large models and adversary examples generated from small and large attack strength. Circles and triangles represent training examples for classification. The curves represent coarse and fine decision boundaries under small and large width models. N1 and N2 in (a) (b) represent the adversarial examples generated by small attack strength (small Euclidean distance to the cluster mean). N1 and N2 in (c) (d) represent the adversarial examples generated by large attack strength (large Euclidean distance to the cluster mean). Classification error of N1 appears in (a) and classification error of N2 appears in (d).

C Additional Training Results

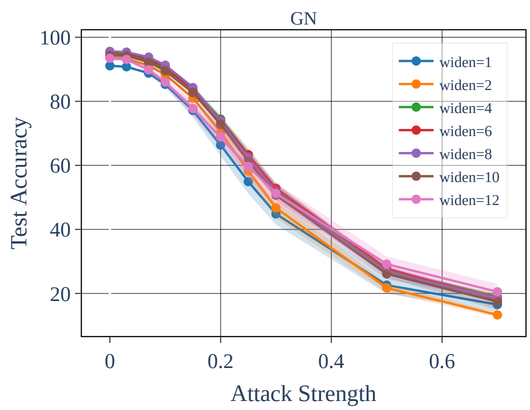
C.1 CIFAR10



(a) FGSM

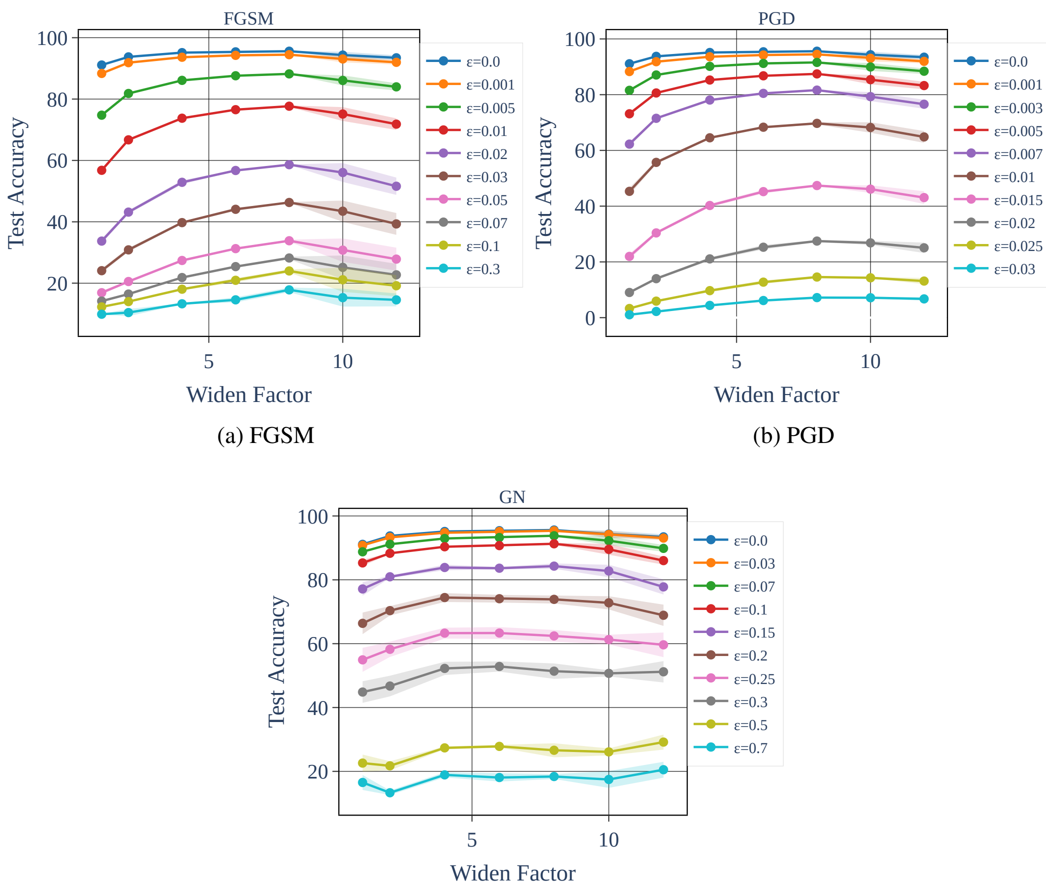


(b) PGD



(c) GN

Figure 3: The relationship between robustness and attack strengths under different attack methods using different width of WRN models trained on CIFAR10



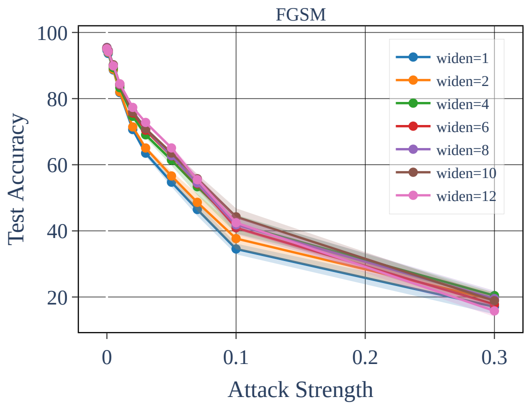
(a) FGSM

(b) PGD

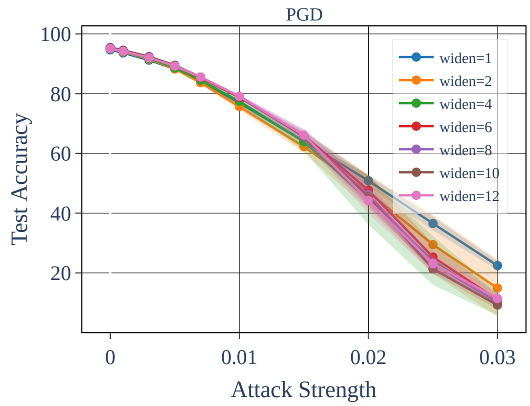
(c) GN

Figure 4: The relationship between robustness and WRN models width trained on CIFAR10 under different attack methods and strengths

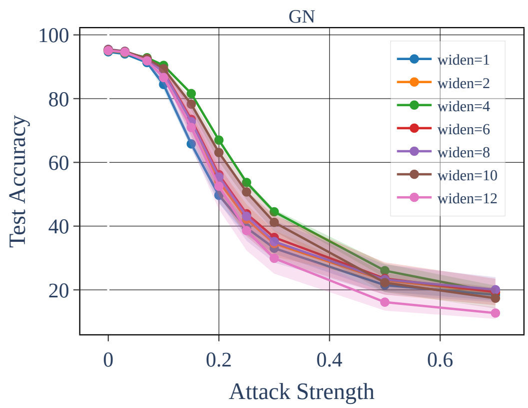
C.2 FASHIONMNIST



(a) FGSM

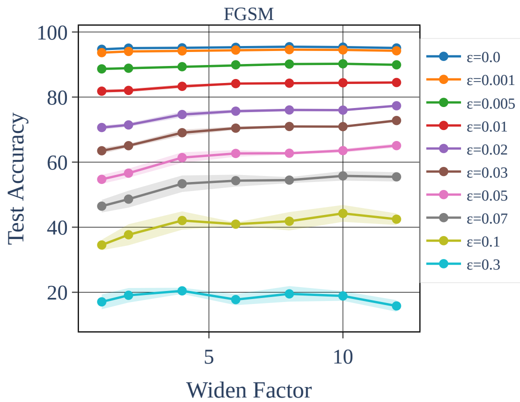


(b) PGD

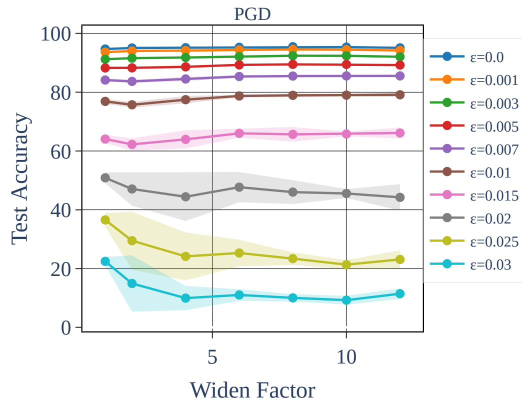


(c) GN

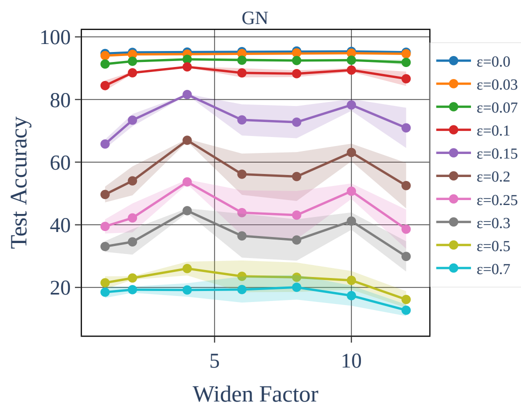
Figure 5: The relationship between robustness and attack strengths under different attack methods using different width of WRN models trained on FashionMNIST



(a) FGSM



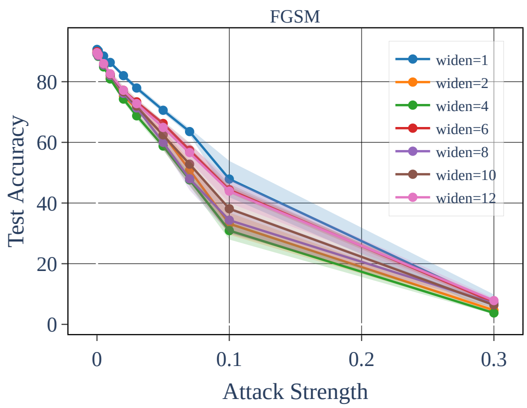
(b) PGD



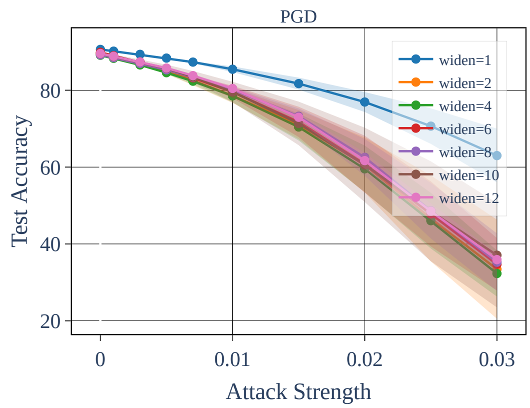
(c) GN

Figure 6: The relationship between robustness and WRN models width trained on FashionMNIST under different attack methods and strengths

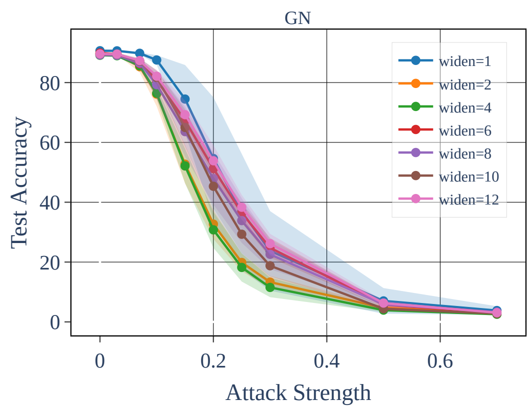
C.3 EMNIST



(a) FGSM

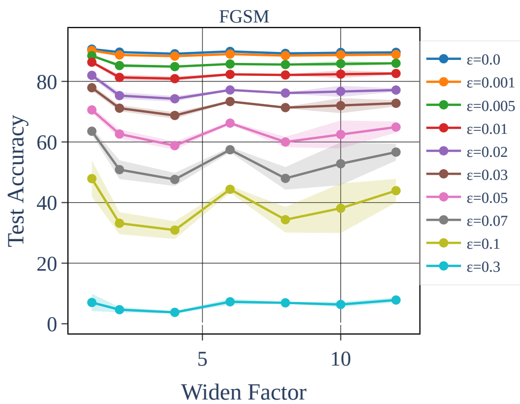


(b) PGD

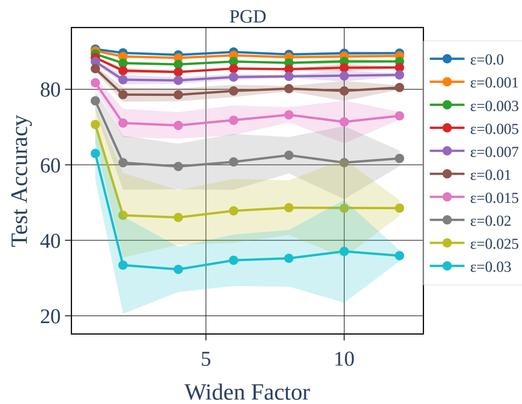


(c) GN

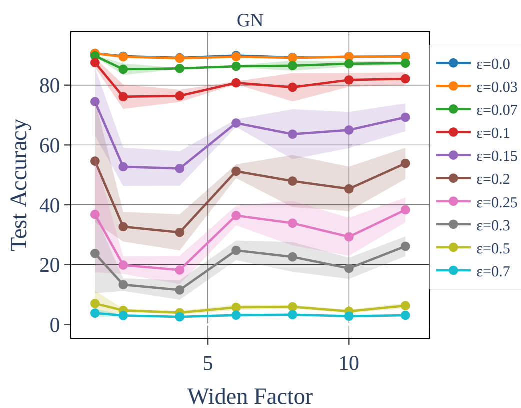
Figure 7: The relationship between robustness and attack strengths under different attack methods using different width of WRN models trained on EMNIST



(a) FGSM



(b) PGD



(c) GN

Figure 8: The relationship between robustness and WRN models width trained on EMNIST under different attack methods and strengths

D Contribution Table

name	contribution
Cong Yu Fang	33.3% (Implement the train and evaluation, run experiments, write the report)
Guanjie Wang	33.3% (Literature review, run experiments, write the report)
Yongchao Zhou	33.3% (Literature review, write the code for visualization, run experiments, write the report)