Robustness of Deep Learning Against Adversarial Examples
Geomatics Seminar Report

Bingxin Ke
Supervisor: Andres Rodriguez, Prof.Dr. Schindler Konrad

December 2020, Zurich

1 Introduction

The security aspect of machine learning, especially in computer vision and natural language processing is increasingly important. There are a series of method to ”fool” a machine learning model. Adversarial example is one of the most frequent used methods.

The adversarial example for a given labeled data (x, y) is an adversarially chosen input data point x' that causes a classifier c to output a different label y' while the difference(perturbation) between x and x' is small, or in other words, ”imperceptible”. Adversarial Example is a general concept for many kinds of classifiers. In this report, we will focus on the deep network for image data. Let’s take a look at the example in the experiment of Goodfellow et al. (2014):

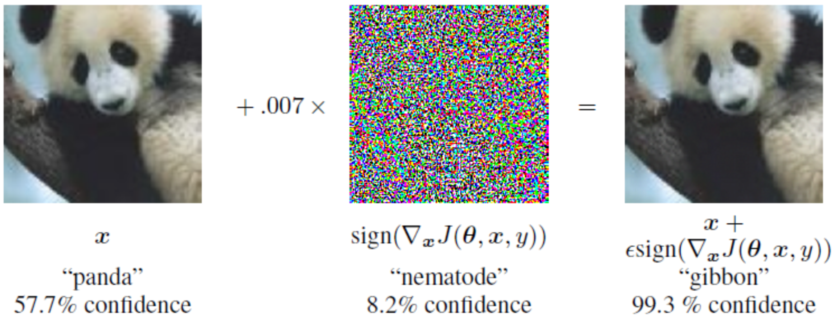

+ .007 ×
=
$\boldsymbol{x}$
"panda"
57.7% confidence
$\text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$
"nematode"
8.2% confidence
$\boldsymbol{x} + \epsilon\text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$
"gibbon"
99.3 % confidence

Figure 1 A demonstration of fast adversarial example generation made by Goodfellow et al. (2014)

In this experiment, they generate an optimal max-norm constrained perturbation using fast gradient sign method. By adding this perturbation, which is imperceptible to human, they change the classification of GoogLeNet (Szegedy et al., 2015).

Szegedy et al. (2013) have an intriguing finding that several machine learning models are vulnerable to adversarial examples, even including those stat-of-the-art neural networks.

2 How do Adversarial Examples Work

Goodfellow et al. (2014) propose that the primary cause of neural networks’ vulnerability to adversarial perturbation is their linear nature. They also showed that linear behavior in high-dimensional space is sufficient to cause adversarial examples.

2.1 Linear Explanation

Goodfellow et al. (2014) give a linear explanation of the adversarial example principle. They took 8-bit digital images as an example. The precision of features is 1/255, which means they discard all information below 1/255 of the dynamic range. (Here it's 1/255 instead of 1/256 because in an 8-bit pixel dynamic range is divided into 255 parts by 256 possible values.)

For a well-separated classifier, we expect it to assign the same class to x and $\tilde{x} = x + \eta$ when $\|\eta\|_\infty < \epsilon$, where ϵ is small enough to be discarded by digital devices in this problem.

How would the classifier response actually to this perturbation? We take dot product between the weight vector ω and the adversarial example $\tilde{x}$:

$$\omega^T \tilde{x} = \omega^T x + \omega^T \eta$$

The activation grow caused by adversarial perturbation is $\omega^T \eta$. If we purposely maximize the increase by assigning $\eta = sign(\omega)$. Then the activation will grow by $\epsilon \cdot m \cdot n$, where m is the average magnitude of elements of weight vector, n is the dimensionality of ω. Activation growth $\omega^T \eta$ will grow with n. Thus, for high dimensional problems, we can make many infinitesimal changes into the input and add up to one large change to the output. From this result, we should be aware that even simple model (linear regression) might be vulnerable to adversarial attack, which means adversarial examples are not something special for deep network, it could be something dangerous to most kinds of machine learning models.

3 How to Generate Adversarial Examples

3.1 Fast Gradient Sign Method (FGSM)

Goodfellow et al. (2014) proposed "fast gradient sign method" of generating adversarial examples. This is considered a one-step method(Madry et al., 2017).

Let $J(\theta, x, y)$ be the cost function of a neural network, where θ denotes the parameters of the model, x denotes the input and y denotes the targets (labels). We can linearize J around θ and get an optimal max-norm constrained perturbation:

$$\eta = \epsilon \cdot \mathrm{sign}(\bigtriangledown_x J(\theta, x, y)),$$

noting that ϵ is the precision of features. They also note that using backpropagation can efficiently compute this gradient. In this experiment, they found that FGSM reliably causes misclassification of a wide variety of models. Figure 1 shows an example on GoogLeNet Szegedy et al. (2013) with $\epsilon = .007$. They successfully change the classification of the image. I put their results of other experiment settings into Table 1 below:

Table 1 Experiment Result of FGSM carried out by Goodfellow et al. (2014)

Classifier	Data set	ϵ	Error Rate	Average confidence
shallow softmax	MNIST	.25	99.9%	79.3%
maxout	MNIST	.25	89.4%	97.6%
convolutional maxout	CIFAR-10	.1	87.15%	96.6%

3.2 Projected Gradient Descent (PGD)

Since the work of Tramèr et al. (2017) shows that FGSM has some important shortcomings, Madry et al. (2017) use a multi-step method, projected gradient descent, as a more powerful adversary:

$$x^{t+1} = \Pi_{x+\mathcal{S}} \left(x^t + \alpha \operatorname{sign} \left(\nabla_x L(\theta, x, y) \right) \right).$$

Madry et al. (2017) carried out experiments and showed that for both normally trained networks and adversarially trained networks, the local maxima found by PGD have similar loss values. And they conjecture that PGD maxima is the local maximum of all first-order adversaries. Thus, if we train a network to be robust against PGD adversaries, it becomes robust against a wide range of other first-order attacks.

3.3 Other Methods

Goodfellow et al. (2014) mentioned that other simple methods are also possible to generate adversarial examples, e.g. rotating x by a small angle in the direction of the gradient.

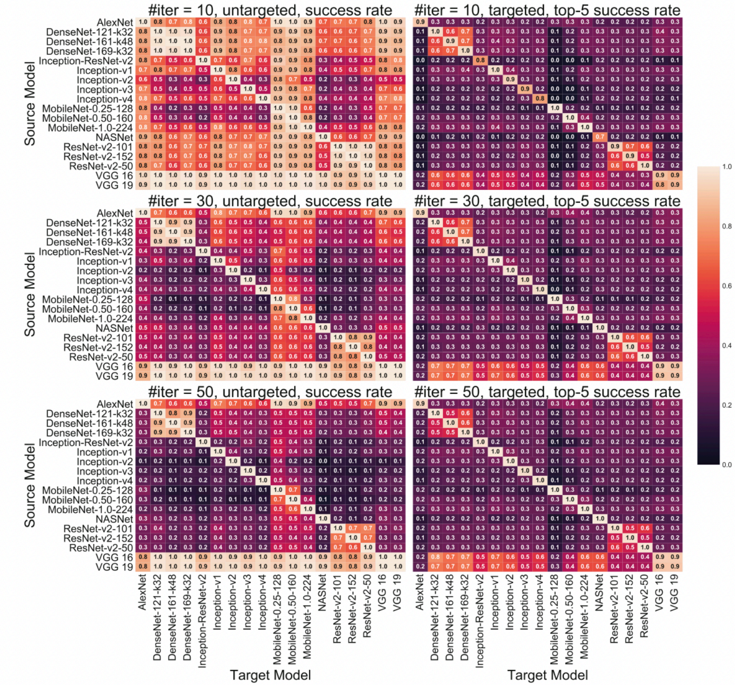


Figure 2 Transferability of I-FGSM attack over 18 ImageNet models, $\epsilon = 0.3$. Su et al.

3.4 Transferred Adversarial Examples

Attacks are not limited to the model it trained on, it can also be used to attack other models. This is called transfer attack, In the experiment carried out by Su et al.. From the result, we can find that the adversarial examples generated by the VGG family can transfer very well to all the other 17 models, while most adversarial examples of other models can only transfer within the same model family.

4 How to Improve Robustness

4.1 Saddle Point Problem

Madry et al. (2017) summarize the attack-defense problem of adversarial samples as a saddle point problem: ([2]-2.1)

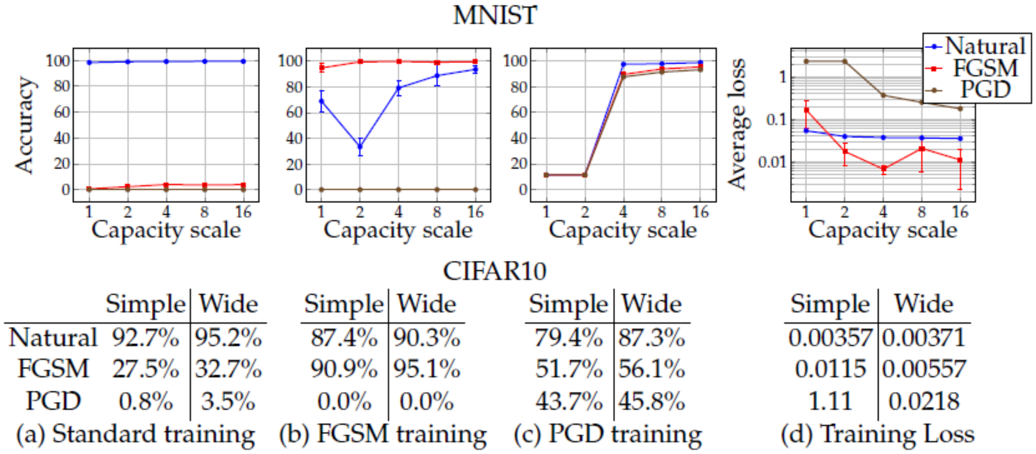
$$\min_{\theta} \rho(\theta), \qquad \text{where} \qquad \rho(\theta) = E_{(x,y)\sim\mathcal{D}} \left[\max_{\delta \in \mathcal{S}} L(\theta, x+\delta, y) \right]$$

1. Inner maximization - to find an adversarial version of a given data point x that achieves a high loss
2. Outer minimization - to find model parameters so that the "adversarial loss" given by the inner attack problem is minimized

4.2 Model Capacity

Capacity is an interesting property of a deep network. Larger capacity of neural network can always deposit more information to improve the discrimination ability of the model (Wang et al., 2017).

Madry et al. (2017) explored the robustness of networks with different capacity. The result shows that capacity is crucial for the ability to successfully train against adversaries. The result in Figure 3 is the result of network with different capacity(x axis) trained on (a)natural examples, (b)with FGSM-made adversarial examples, (c) with PGD-made adversarial examples. It shows that: firstly, without adversarial training, improvement on capacity improves the robustness against FGSM attack, secondly models trained with FGSM examples are not robust against PGD attack, thirdly the robustness of PGD training increases with model capacity. Su et al. also concluded that for a similar network architecture, increasing network depth slightly improves robustness in l_∞ distortion metric.



	Simple	Wide	Simple	Wide	Simple	Wide	Simple	Wide
Natural	92.7%	95.2%	87.4%	90.3%	79.4%	87.3%	0.00357	0.00371
FGSM	27.5%	32.7%	90.9%	95.1%	51.7%	56.1%	0.0115	0.00557
PGD	0.8%	3.5%	0.0%	0.0%	43.7%	45.8%	1.11	0.0218

(a) Standard training (b) FGSM training (c) PGD training (d) Training Loss

Figure 3 The effect of network capacity on the performance of the network (Madry et al., 2017)

4.3 Model Architecture

Model architecture is also a crucial aspect we should take into account. Goodfellow et al. (2014) think that RBF networks are naturally immune to adversarial examples, since they default to predicting the class is absent or have low confidence when they are ”fooled”. Su et al. found that each family of networks exhibits a similar level of robustness, despite different depths and model sizes.

4.4 Adversarial Training

Madry et al. (2017) propose that there are two steps to reliably train models that are robust to adversarial attacks. Firstly, to specify an attack model by a precise definition. Next step is to perturb the input. This is consistent with a widely used method, training on a mixture of adversarial and clean examples, in other words, augment the dataset with adversarial examples. Szegedy et al. (2013) also propose that training on adversarial examples can regularize the model. From the experiment (Figure 2) we can find that networks of a same family have a similar level of robustness, despite different depths and model sizes. And, it's also mentioned by Goodfellow et al. (2014) that RBF networks are naturally immune to adversarial examples, since they default to predicting the the class is absent or have low confidence when they are ”fooled”. But we can't say which model is the best or has the best robustness. It really depends on the application, data set and many other factors.

4.4.1 Using FGSM Attacking

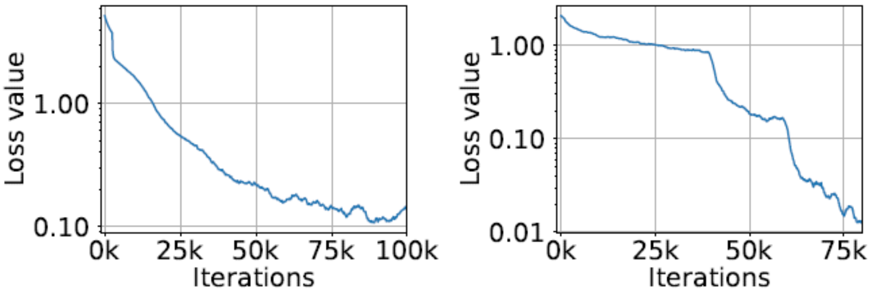
Goodfellow et al. (2014) show that adversarial training with examples produced by FGSM can bring a better regularization benefit than using dropout alone. And training with an adversarial objective function based on the fast gradient sign method was an effective regularizer:

$$\tilde{J}(\boldsymbol{\theta}, \boldsymbol{x}, y) = \alpha J(\boldsymbol{\theta}, \boldsymbol{x}, y) + (1 - \alpha) J\left(\boldsymbol{\theta}, \boldsymbol{x} + \epsilon \operatorname{sign}\left(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y)\right)\right.$$

They used this approach to continually train the network on updating adversarial examples, which resist the current version of the model. These data may not occur naturally but expose weakness of model. This is a process of mutual improvement. The adversarial keep finding blind points of model while model improving its robustness against these adversarial examples.

4.4.2 Using PGD Attacking

Madry et al. (2017) replace the inputs by their corresponding adversarial perturbations and normally train the network on the input that perturbed using projected gradient descent (PGD), which is also known as iterative fast gradient sign method (I-FGSM). They use Stochastic Gradient Descent (SGD) to minimize th loss function, i.e. to solve the outer optimization problem of the saddle point formulation. Their experiments (as shown in Figure 4) illustrate that by applying SGD using the loss gradient at adversarial examples, we can consistently reduce the loss of the saddle point problem during training, thus producing an increasingly robust classifier.



(a) MNIST (b) CIFAR10

Figure 4 Cross-entropy loss on adversarial examples during training (Madry et al., 2017)

4.5 Trade-off between Robustness and Accuracy

Su et al. empirically found the relation between robustness and accuracy of different ImageNet models - the distortion scales linearly with the logarithm of classification error. Zhang et al. (2019) theoretically decompose the prediction error for adversarial examples (robust error) as the sum of the natural classification error and boundary error:

$$\mathcal{R}_{rob}(f) = \mathcal{R}_{nat}(f) + \mathcal{R}_{bdy}(f)$$

5 Outlook

Though we already have a primary understanding of adversarial examples and have some attack methods. The theory behind this is still unclear, so that we can't fully produce a model against all white-box attack. On the other hand, because of the effeteness of transfer attack, the robustness against black-box attack is also what we should explore in the future.

References

I. J. Goodfellow, J. Shlens, and C. Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

D. Su, H. Zhang, H. Chen, J. Yi, P.-Y. Chen, and Y. Gao. Is robustness the cost of accuracy?–a comprehensive study on the robustness of 18 deep image classification models. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 631–648.

C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

F. Tramèr, N. Papernot, I. Goodfellow, D. Boneh, and P. McDaniel. The space of transferable adversarial examples. *arXiv preprint arXiv:1704.03453*, 2017.

A. Wang, H. Zhou, W. Xu, and X. Chen. Deep neural network capacity. *arXiv preprint arXiv:1708.05029*, 2017.

H. Zhang, Y. Yu, J. Jiao, E. P. Xing, L. E. Ghaoui, and M. I. Jordan. Theoretically principled trade-off between robustness and accuracy. *arXiv preprint arXiv:1901.08573*, 2019.