1 Code

In this project, I implement the ZOO attacks in Matlab to evade a DNN model trained on the MNIST dataset(source code: Github).
The key idea of the ZOO attacks is to estimate the gradient of a black-box model. The approximate gradient is computed with the zeroth-order optimation method instead of using the actual backpropagation on the targeted model.
First, for a targeted attack, the loss function is

$$f(x,t) = max\{max_{i\neq t} log[F(x)]_i - log[F(x)]_t, 0\} \tag{1}$$

Minimizing the loss function requires to attains the highest confidence score for the targeted class t.
Then, ZOO attacks use the symmetric difference quotient to estimate the gradient on the loss function.

$$g_i = \frac{\partial f(x)}{\partial x_i} = \frac{f(x+he_i) - f(x-he_i)}{2h} \tag{2}$$

To reduce the computational complexity, I implemented two algorithms ZOO-ADAM and ZOO-Newton proposed by [**1**]. Both of them first randomly pick a batch of coordinates. Then ZOO-ADAM method calculates the first-order of the loss function to compute the best coordinate update. Meanwhile, the ZOO-Newton method calculates the gradient and Hessian for x_i to update the coordinate.

2 Result

2.1 Attack Rates

It takes a long time to calculate the adversarial samples on the whole test dataset(about one day for 300 adversarial samples on my computer). To evaluate the performance of the attack methods, I first selected 1000 images that could be correctly classified as their true labels and randomly select a target label. **The success attack rate of both algorithms are 100%.**
Then I evaluated the attack rate for different target labels. I selected 200 images and generated 9 adversarial samples for each image with target classes that are different from their true labels. Table **??** shows the attack success rate with those two attack methods. Both of them can reach a 100% attack rate with all target classes.

Target Label	ZOO-ADAM	ZOO-Newton
0	100%	100%
1	100%	100%
2	100%	100%
3	100%	100%
4	100%	100%
5	100%	100%
6	100%	100%
7	100%	100%
8	100%	100%
9	100%	100%

Tabelle 1: Attack Rate

In practice, the ZOO-ADAM method is faster than the ZOO-Newton method. **For ZOO-ADAM attack the average number of classifier queries to achieve a successful attack is 883.8. For the ZOO-Newton attack, the average number of queries is 1471.1.** Each query randomly selects 128 coordinates to update.
As shown in Fig.**??**, the human can see there are some noises on the adversarial samples but we still can identify true labels for those generated samples. **Generally, the adversarial samples are imperceptible.**

2.2 Defense

One way to defend against this attack is to train another model to differentiate the adversarial samples. I trained neural networks using the same architecture as the classification model. 70% generated adversarial

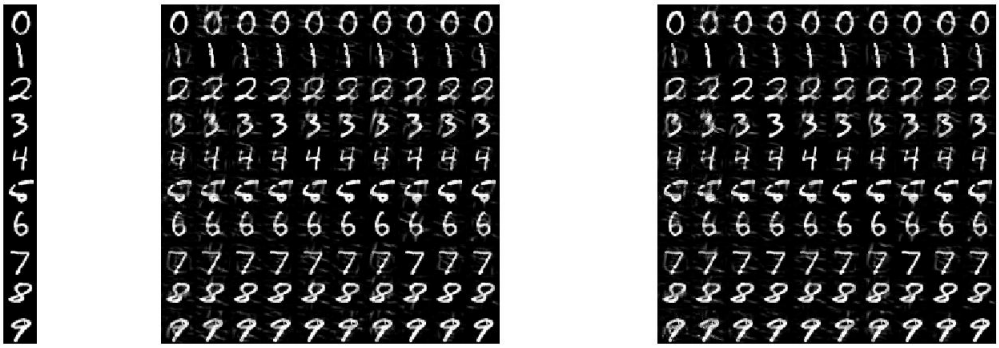

Abbildung 1: Successful adversarial examples in MNIST. Each row displays adversarial examples from the sampled images. Each column indexes the targeted class(0-9) for attack.

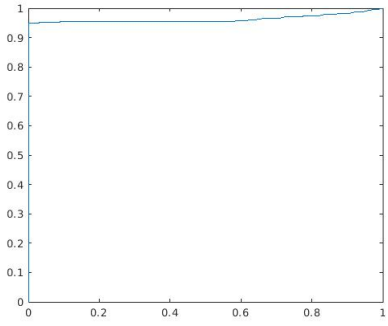

Abbildung 2: Performance of the defense method for the ZOO-ADAM attack(neural networks)

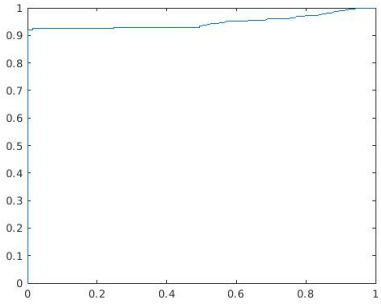

Abbildung 3: Performance of the defense method for the ZOO-Newton attack(neural networks)

examples and their corresponding original images are used for training and others are used to test the model. Figure **??** is the performance of the detection model. **The AUC of the model for the ZOO-ADAM attack on the test dataset is 95.36%. For the ZOO-Newton attack, it is 94.74%.**

For the MNIST dataset, I found the generate samples have an obvious characteristic. There are more pixels with the gray color in adversarial samples. Fig. **??** is the frequency of 256 different intensities in 200 original samples and 200 adversarial samples. If we filter out the white(256) and black(0), the differences between the generated samples and the original samples are more significant as shown in Fig. **??**.

So we could calculate the ratio of "gray color"(intensities from 2 to 90) in one unknown image to detect the attack. First, we use 70% generated adversarial examples and their corresponding original images to define the threshold that would be classified as attacks. Figure **??** is the ratio of gray color in original images and adversarial images. Based on the statical analysis, we simply define the following decision rules: if the ratio is more than 9%, we can classify the images as attacks with 100%; if the ratio is more than 6% and less than 9%, the probability the image is not an adversarial example is 70%; if it is more than 2% and less than 6%, the probability the image is not an adversarial example is 90%; otherwise, the probability the image is not an

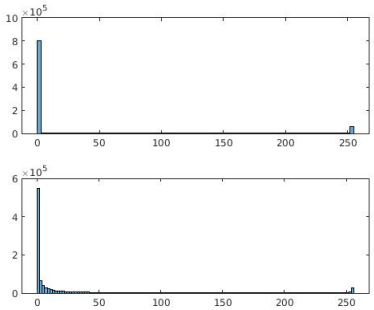

Abbildung 4: The frequency of different intensities (from 0 to 255)

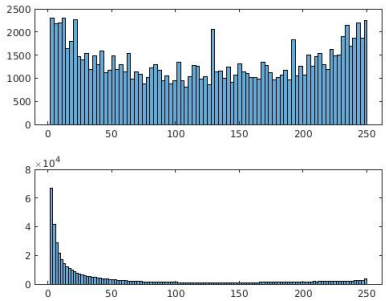

Abbildung 5: The frequency of different intensities (from 2 to 250)

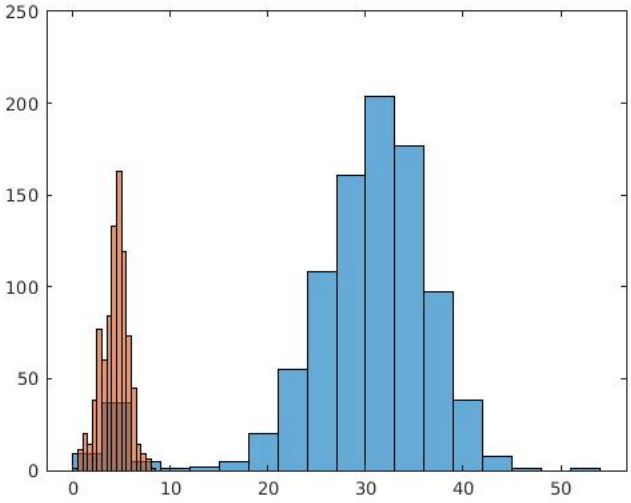

Abbildung 6: The ratio of gray color in original images and adversarial images

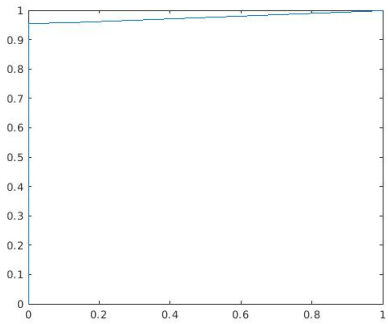

Abbildung 7: Performance of the defense method for the ZOO-ADAM attack(gray color)

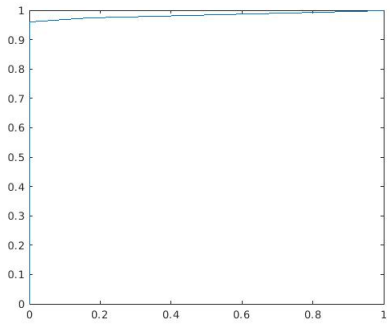

Abbildung 8: Performance of the defense method for the ZOO-Newton attack(gray color)

adversarial example is 70%.
Then we use these simple rules to calculate the AUC on the test dataset. Figure **??** is the performance of this detection method. And **the AUC for the ZOO-ADAM attack is 97.66%; for the ZOO-Newton attack it is 98.36%**.

Literatur

[1] Chen, P. Y., Zhang, H., Sharma, Y., Yi, J., & Hsieh, C. J. (2017). ZOO: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models. ArXiv.