Programming Assignment 3
Adversarial Machine Learning with the MNIST

Ji In Kwak
Carnegie Mellon University
jiink@andrew.cmu.edu

November 21, 2022

1 White-Box Attack using the Fast Gradient Sign Attack Method (FGSM)

1.1 Task

The fast gradient sign attack method (FGSM) is one of the efficient and effective method for generating adversarial examples. For each pixel, it perturbates for $+\epsilon$ or $-\epsilon$ from the original pixel values and generates the images that are misclassified as the different label. As the value of increases, the adversarial images become more attackable. The Figure 1 shows the generated adversarial images from FGSM and we can see that the images are misclassified when the epsilon value is larger than zero. In Figure 2, we compared the accuracy of the classifier model when the epsilon increases. The accuracy was initially 98.1% but is decreased into 8.7%.

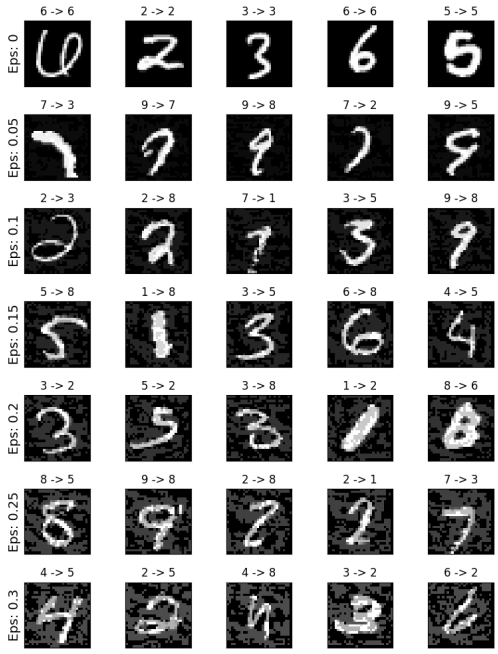

Figure 1: The examples of adversarial images generated using FGSM

1.2 Task

We can ensure that the pixel value perturbates at most ϵ, because after changing every pixel value by adding $+\epsilon$ or $-\epsilon$, we clamped the values between 0 and 1. For example, when the perturbated values $x' = x + \epsilon$ is larger than 1, the value becomes $x' = 1$ so that the perturbated value is smaller than .

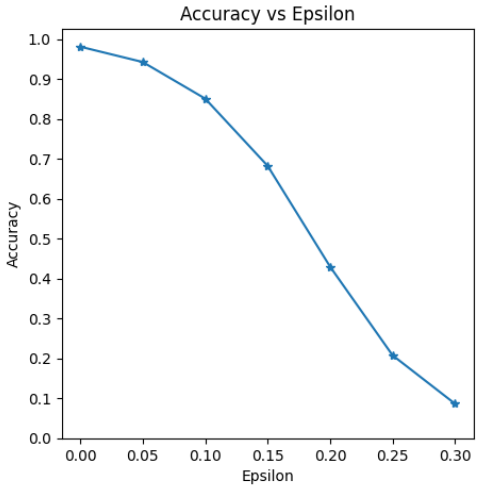

Figure 2: Model accuracy with adversarial images of different perturbations using FGSM

1.3 Task

$J(\theta, x_{orig}, y)$ is the loss of original image for label y and $J(\theta, x_{adv}, y)$ is a loss of adversarial image for label y. An adversarial image is generated to make the model not classify the image as y. Hence, it attacks in the way of increasing the loss of adversarial images$J(\theta, x_{adv}, y)$. Therefore, we can say that the $J(\theta, x_{adv}, y)$ is strictly larger than the $J(\theta, x_{orig}, y)$.

1.4 Task

This method is very similar to FGSM but there is a difference between them. In case of FGSM, the pixel values can only move with $+\epsilon$ or $-\epsilon$, but in projection methods, it can move with any values between $[-\epsilon, +\epsilon]$. After the perturbation, the clipping methods are the same. In projection method, there is a case of not moving any pixel values if the gradient is zero. Hence, we can say that the $J(\theta, x_{adv}, y)$ is larger or equal to the $J(\theta, x_{orig}, y)$.

2 White-Box Attack Using Projected Gradient Descent (PGD)

2.1 Task

PGD method is a different attack method from FGSM which operates with the number of iterations. Figure 3 shows the example images of PGD with different epsilon values and Figure 4 shows the accuracy according to the various epsilon values. We can see that the success rate of attack is very high from $\epsilon = 0.05$. This means we can make the adversarial images without visibility to humans. However, it does not mean that the model accuracy always decreases when the ϵ increases.

2.2 Task

In case of FGSM, it is quite fast than the PGD because there are no iteration steps. Just computing the sign of loss function one time is enough for FGSM. However, the success rate of attack is quite low when the epsilon value is too small. We figured out in task 1 that the accuracy of FGSM attack slowly decreases as the perturbation increases. However, in case of PGD, the attack is very efficient although the epsilon value is 0.05 which is very small. And the cons of PGD is that it takes time to calculate the 1000 iterations.

2.3 Task

To reduce the computing time, we can reduce the number of iterations in PGD. Then, I think it is necessary to increase the value of to make enough perturbations. Therefore, I experimented with 300 iterations and $\alpha = \frac{\epsilon}{100}$. Then, the result images are shown in the Figure 5 and 6. Since we reduced the iteration numbers from 1000, the computation time was reduced. However, the success rate was similar to the results in task 2.1.

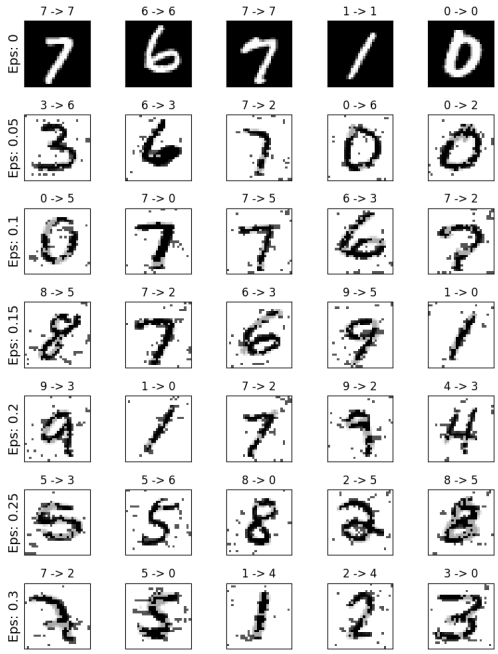

Figure 3: The examples of adversarial images generated using PGD

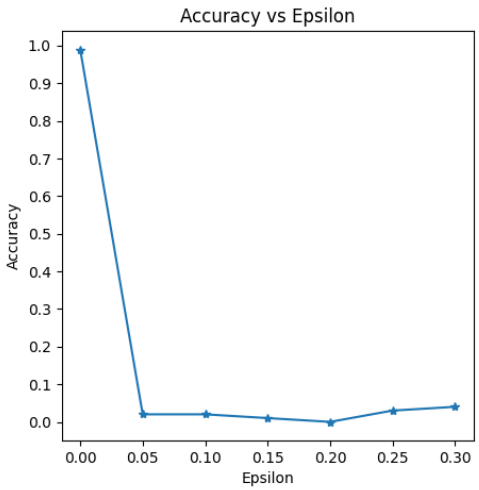

Figure 4: Model accuracy with adversarial images of different perturbations using PGD

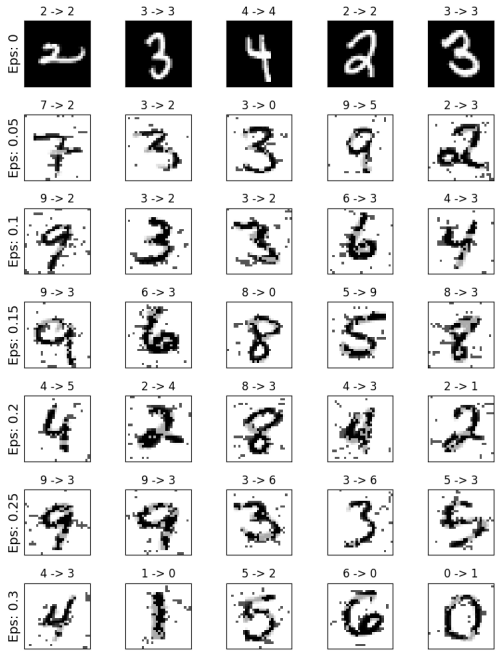

Figure 5: The examples of adversarial images generated using PGD of different parameters

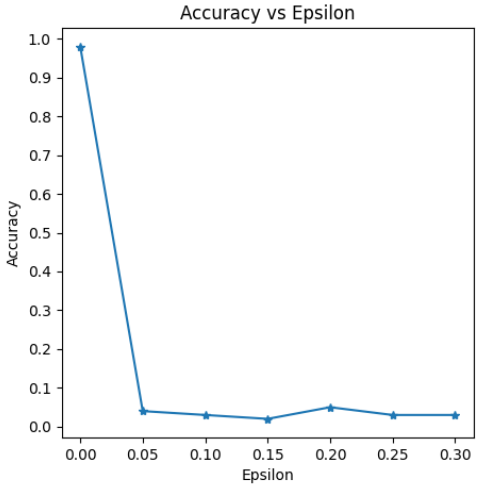

Figure 6: Model accuracy with adversarial images of different perturbations using PGD