# Mathematical Analysis of Adversarial Attacks[*]

Zehao Dou[†] and Stanley J. Osher and Bao Wang[‡]

**Abstract.** In this paper, we analyze the efficacy of the fast gradient sign method (FGSM) and the Carlini-Wagner's $L_2$ (CW-L2) attack. We prove that, within a specific regime, the untargeted FGSM can fool any convolutional neural nets (CNN) with ReLU activation; the targeted FGSM can mislead any CNN with ReLU activation to classify any given image into any prescribed class. For a particular two-layer neural nets, a linear layer followed by the softmax output activation, we show that the CW-L2 attack increases the ratio of the classification probability between the target and the ground truth classes. Moreover, we provide numerical results to verify our theoretical results.

**Key words.** Adversarial Attack, Deep Learning.

**AMS subject classifications.** 41A63, 65D17, 68T05

## 1. Introduction.

The adversarial vulnerability [19] of deep neural nets (DNN) threatens their applicability in security critical tasks, e.g., autonomous cars [1], robotics [7], DNN-based malware detection systems [15, 6]. Since the pioneering work by Szegedy et al. [19], many advanced adversarial attack schemes have been devised to generate imperceptible perturbations to sufficiently fool the DNN [5, 14, 4]. Not only are adversarial attacks successful in white-box attacks, i.e., when the adversary has access to the DNN parameters, but they are also successful in black-box attacks, i.e., it has no access to the parameters. Black-box attacks are successful because one can perturb an image so it misclassifies on one DNN, and the same perturbed image also has a significant chance to be misclassified by another DNN; this is known as transferability of adversarial examples [17]. Due to this transferability, it is very easy to attack neural nets in a black-box fashion [3]. In fact, there exist universal perturbations that can imperceptibly perturb any image and cause misclassification for any given network [13]. There is much recent research on designing advanced adversarial attacks and defending against adversarial perturbation.

Defensive distillation was recently proposed to increase the stability of DNN [16], and a related approach [20] cleverly modifies the training data to increase robustness against black-box attacks, and adversarial attacks in general. To counter the adversarial perturbations, Guo et al. [8], proposed to use image transformation, e.g., bit-depth reduction, JPEG compression, TVM, and image quilting. Adversarial training is another family of defense methods to improve the stability of DNN [5]. In particular, the projected gradient descent (PGD) adversarial training achieves state-of-the-art guaranteed resistance to the first order attack [12]. Generative adversarial nets (GANs) are also employed for adversarial defense [18]. In [2], the authors proposed an approximated gradient to attack the defenses that is based on the obfuscated gradient. Wang et al. [22, 21], introduce a data dependent activation to defend against

[*] Submitted to the editors DATE.

[†] School of Mathematical Science, Peking University, Beijing, China (zehaodou@pku.edu.cn).

[‡] Department of Mathematics, UCLA, Los Angeles, CA, 90095-1555 (sjo@math.ucla.edu, wangbaonj@gmail.com).

adversarial attacks, joint with total variation minimization, training data augmentation, and the PGD adversarial training, state-of-the-art adversarial defense results are achieved. More recently, motivated by the Feynman-Kac formalism, Wang et al. [24], proposed a novel neural nets ensemble algorithm which significantly improves the guaranteed robustness towards the first order adversarial attack.

In this paper, we analyze the efficacy of the fast gradient sign method (FGSM) [5, 11] and the Carlini-Wagner's $L_2$ (CW-L2) attack [4]. FGSM belongs to the fixed perturbation attack, while CW-L2 attack belongs to the zero-confidence attack. For FGSM, we consider both the targeted and the untargeted attacks. We prove that, within a specific regime, the untargeted FGSM can fool any convolutional neural nets (CNN) with ReLU activation; the targeted FGSM can mislead any CNN with ReLU activation to classify any given image into any prescribed class. For a two-layer neural nets, a linear layer followed by the softmax output activation, we show that the CW-L2 attack increases the ratio of the classification probability between the target and ground truth classes. Our theoretical results give guidance on applying different attacks to attack neural nets, especially, the targeted ones.

This paper is structured in the following way: In section 2, we give a review of the well known adversarial attack schemes and briefly discuss the mathematical principle behind them. We analyze the untargeted FGSM, the targeted FGSM, and the CW-L2 attacks, respectively, in sections 3, 4, 5. We verify our theoretical results numerically in section 6. The paper ends up with concluding remarks.

## 2. Adversarial Attacks.

We denote the classifier defined by the DNN with softmax output activation as $\tilde{y} = f(\theta, x)$ for a given image-label pair $(x, y)$. FGSM finds the adversarial image $x'$ by maximizing the loss $L(x', y) \doteq L(f(\theta, x'), y)$, subject to the $l_\infty$ perturbation constraint $||x' - x||_\infty \leq \epsilon$ with $\epsilon$ be the attack strength. Under the first order approximation i.e., $L(x', y) \approx L(x, y) + \nabla_x L(x, y)^T \cdot (x' - x)$, we have

$$x' = x + \epsilon \operatorname{sign} \cdot (\nabla_x L(x, y)) . \tag{2.1}$$

IFGSM iterates FGSM to generate enhanced attacks, i.e.,

$$x^{(m)} = x^{(m-1)} + \epsilon \cdot \operatorname{sign}\left(\nabla_x L(x^{(m-1)}, y)\right), \tag{2.2}$$

where $m = 1, \cdots, M$, $x^{(0)} = x$ and $x' = x^{(M)}$, with $M$ being the number of iterations.

In practice, we apply the following clipped IFGSM

$$x^{(m)} = \mathrm{Clip}_{x,\alpha}\left\{x^{(m-1)} + \epsilon \cdot \operatorname{sign}\left(\nabla_x L(x^{(m-1)}, y)\right)\right\}, \tag{2.3}$$

where $\alpha$ is an additional parameter to be specified in the experiments.

*Remark* 2.1. The above FGSM or IFGSM attack fools DNN to mis-classify the image $x$. To mislead the classification result falls into any given class $t$, with one-hot label $e_t$, we apply the following targeted FGSM schemes

- Targted FGSM

$$x' = x - \epsilon \operatorname{sign} \cdot (\nabla_x L(x, e_t)) . \tag{2.4}$$

- Targted IFGSM

$$x^{(m)} = x^{(m-1)} - \epsilon \cdot \text{sign}\left(\nabla_x L(x^{(m-1)}, e_t)\right), \tag{2.5}$$

where $m = 1, \cdots, M$, $x^{(0)} = x$ and $x' = x^{(M)}$, with $M$ being the number of iterations. In Eqs. (2.4, 2.5), $L(x, e_t)$ is the loss between predicted label of the adversarial image and the targeted label $e_t$.

Furthermore, we consider the following zero-confidence attack. For a given image-label pair $(x, y)$, and $\forall t \neq y$, CW-L2 searches the adversarial image that will be classified to class $t$ by solving the optimization problem:

$$\min_{\delta} ||\delta||_2^2, \qquad \text{subject to } f(x+\delta) = t,\ x+\delta \in [0,1]^n, \tag{2.6}$$

where $\delta$ is the adversarial perturbation (for simplicity, we ignore the dependence of $\theta$ in $f$).

The equality constraint in Eq. (2.6) is hard to handle, so Carlini et al. consider the surrogate

$$g(x) = \max\left(\max_{i\neq t}(Z(x)_i) - Z(x)_t, 0\right), \tag{2.7}$$

where $Z(x)$ is the logit vector for an input $x$, i.e., output of the neural nets before the softmax layer. $Z(x)_i$ is the logit value corresponding to class $i$. It is easy to see that $f(x+\delta) = t$ is equivalent to $g(x+\delta) \leq 0$. Therefore, the problem in Eq. (2.6) can be reformulated as

$$\min_{\delta} ||\delta||_2^2 + c \cdot g(x+\delta) \qquad \text{subject to } x+\delta \in [0,1]^n, \tag{2.8}$$

where $c \geq 0$ is the Lagrangian multiplier.

By letting $\delta = \frac{1}{2}(\tanh(w)+1) - x$, Eq. (2.8) can be written as an unconstrained problem. Moreover, Carlini et al. introduce the confidence parameter $\kappa$ into the above formulation. Above all, CW-L2 attacks seek the adversarial image by solving the following problem

$$\begin{aligned} &\min_{w} ||\frac{1}{2}(\tanh(w)+1) - x||_2^2 + c \cdot \\ &\max\left\{-\kappa, \max_{i\neq t}(Z(\frac{1}{2}(\tanh(w))+1)_i) - Z(\frac{1}{2}(\tanh(w))+1)_t\right\}. \end{aligned} \tag{2.9}$$

This unconstrained problem can be solved efficiently by the Adam optimizer [10].

In the case of CW-L2 attack, we introduce different levels of adversarial attack by setting the adversarial image to

$$x' = x + \epsilon\left(x^{\text{adv}} - x\right), \tag{2.10}$$

where $x^{\text{adv}}$ is the solution to Eq. (2.9).

Figure 1 depicts three randomly selected images (horse, automobile, airplane) from the CIFAR10 dataset, their adversarials by using different attacks on ResNet56. All attacks fool the classifiers completely on these images. Figure 1 (a) shows that the perturbations resulted from FGSM and IFGSM with $\epsilon = 0.02$, 10 iterations with $\alpha = 0.1$ for IFGSM, is almost imperceptible. For CW-L2, we set the parameters $c = 10$ and $\kappa = 0$, and run 10 iterations of Adam optimizer with learning rate 0.01. Figure 1 (b) shows the corresponding images of (a) with a stronger attack, $\epsilon = 0.08$. With a larger $\epsilon$, the adversarial images become more noisy.

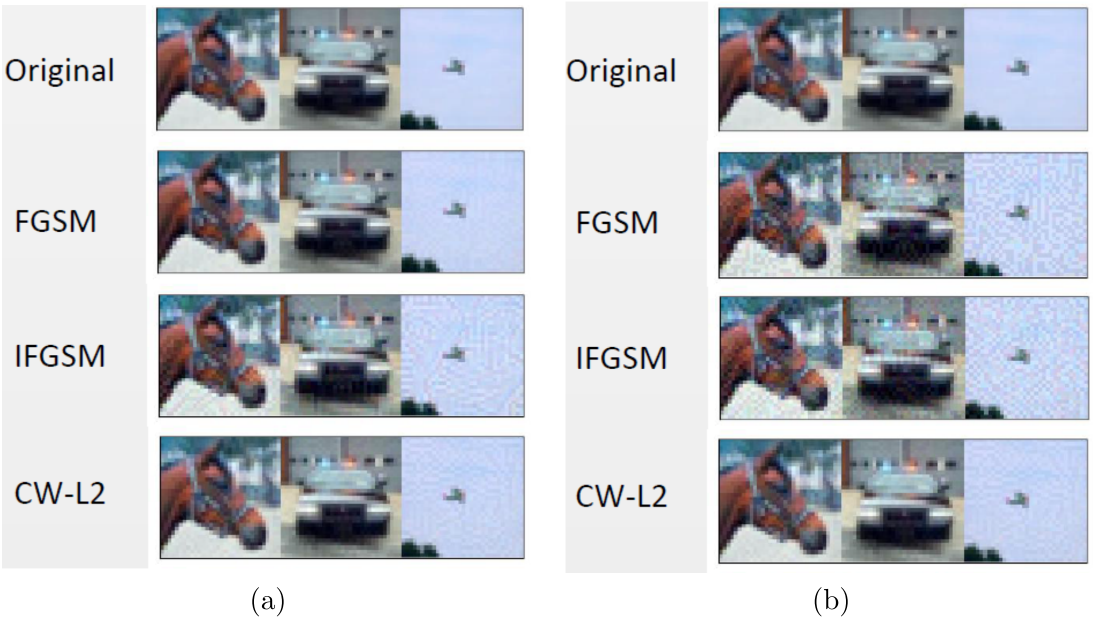


**Figure 1.** *Samples from CIFAR10. Panel (a): from the top to the last rows show the original, adversarial images by attacking ResNet56 with FGSM, IFGSM, CW-L2 ($\epsilon = 0.02$). Panel (b) corresponding to those in panel (a) with $\epsilon = 0.08$.*

## 3. Analysis of the Untargeted FGSM.

**3.1. Case 1. A linear layer followed by a softmax output layer.** For an input image $x \in R^n$ and its corresponding one-hot label vector $y \in R^k$. We consider the simple neural nets

$$\hat{y} = \text{softmax}(Wx), \tag{3.1}$$

and consider the cross entropy loss

$$L(x, y) = \text{crossentropy}(\hat{y}, y) = -\sum_{j=1}^{k} y_j \cdot \ln \hat{y}_j = -\ln \hat{y}_s,$$

where $W \in R^{k\times n}$, $s$ is the index of 1 in the one-hot vector $y$ i.e., $y_s = 1$ and $y_i = 0 \quad \forall i \neq s$.

Theorem 3.1. *For the neural nets defined in Eq. (3.1) and any input-output pair $(x, y)$. Let $x'$ be the adversarial image of $x$ resulting from FGSM attack, i.e.,*

$$x' = x + \epsilon \cdot sign(\nabla_x L(x, y)).$$

*Then, for $\forall \epsilon > 0$, we have:*

$$L(x, y) \leqslant L(x', y).$$

*Proof.* For any given $x$, suppose it belongs to class $s$. The loss can be expressed as:

$$\begin{aligned}L(x,y) &= \text{crossentropy}(\text{softmax}(Wx),y)\\ &= -\ln(\text{softmax}(Wx)_s)\\ &= -\ln(\frac{\exp(Wx)_s}{\exp(Wx)_1+\exp(Wx)_2+\cdots+\exp(Wx)_k})\end{aligned} \tag{3.2}$$

Here,

$$\text{softmax}(Wx)_i = \frac{\exp(Wx)_i}{\sum_{j=1}^k \exp(Wx)_j}.$$

It is easy to get the exact expression of $x'$, in fact,

$$\begin{aligned}x'_i &= x_i + \epsilon \cdot \text{sign}(\frac{\partial}{\partial x_i}L(x,y))\\ &= x_i + \epsilon \cdot \alpha_i.\end{aligned} \tag{3.3}$$

Here:

$$\begin{aligned}\alpha_i &= \text{sign}(\frac{\partial}{\partial x_i}L(x,y))\\ &= \text{sign}(-\frac{\sum_{j=1}^k \exp(Wx)_j}{\exp(Wx)_s}\cdot\\ &\quad \frac{\exp(Wx)_s w_{si}(\sum_{j=1}^k \exp(Wx)_j) - \exp(Wx)_s \sum_{j=1}^k \exp(Wx)_j w_{ji}}{(\sum_{j=1}^k \exp(Wx)_j)^2})\\ &= -\text{sign}(w_{si}(\sum_{j=1}^k \exp(Wx)_j) - \sum_{j=1}^k \exp(Wx)_j w_{ji})\\ &= \text{sign}(\sum_{j=1}^k \exp(Wx)_j w_{ji} - (\sum_{j=1}^k \exp(Wx)_j) w_{si}).\end{aligned} \tag{3.4}$$

In order to prove $L(x,y) \leqslant L(x',y)$, we only need to show

$$\begin{aligned}&\frac{\exp(Wx)_s}{\sum_{j=1}^k \exp(Wx)_j} \geqslant \frac{\exp(Wx')_s}{\sum_{j=1}^k \exp(Wx')_j}\\ \Leftrightarrow &\frac{\exp(Wx')_s}{\exp(Wx)_s} \leqslant \frac{\exp(Wx')_1+\exp(Wx')_2+\cdots+\exp(Wx')_k}{\sum_{j=1}^k \exp(Wx)_j}\\ \Leftrightarrow &\frac{\exp(Wx')_s}{\exp(Wx)_s} \leqslant \sum_{j=1}^k \text{softmax}(Wx)_j \cdot \frac{\exp(Wx')_j}{\exp(Wx)_j}\\ \Leftrightarrow &\exp(\epsilon W\alpha)_s \leqslant \sum_{j=1}^k \text{softmax}(Wx)_j \cdot \exp(\epsilon W\alpha)_j.\end{aligned} \tag{3.5}$$

Here, $\alpha = (\alpha_1, \alpha_2, \cdots, \alpha_k)^T$. Since $\text{softmax}(Wx)_i, 1 \leqslant i \leqslant k$, are $k$ non-negative real numbers sum to 1. By the Jensen's inequality, we can get the following lower bound for the right hand side of Eq. (3.5)

$$\begin{aligned}&\text{softmax}(Wx)_1 \cdot \exp(\epsilon W\alpha)_1 + \cdots + \text{softmax}(Wx)_k \cdot \exp(\epsilon W\alpha)_k \\ &\geqslant \exp(\sum_{j=1}^{k} \epsilon \cdot \text{softmax}(Wx)_j (W\alpha)_j).\end{aligned} \tag{3.6}$$

So far, in order to get Eq. (3.5) proved, we only have to prove the following inequality

$$\sum_{j=1}^{k} \text{softmax}(Wx)_j (W\alpha)_j \geqslant (W\alpha)_s. \tag{3.7}$$

This is equivalent to :

$$\sum_{j=1}^{k} \exp(Wx)_j (W\alpha)_j \geqslant (\exp(Wx)_1 + \cdots + \exp(Wx)_k)(W\alpha)_s. \tag{3.8}$$

In fact, we have

$$\begin{aligned}&\sum_{j=1}^{k} \exp(Wx)_j (W\alpha)_j - (\exp(Wx)_1 + \cdots + \exp(Wx)_k)(W\alpha)_s \\ =&\sum_{j=1}^{k} \exp(Wx)_j (w_{j1}\alpha_1 + \cdots + w_{jn}\alpha_n) - (\sum_{j=1}^{k} \exp(Wx)_j)(w_{s1}\alpha_1 + \cdots + w_{sn}\alpha_n) \\ =&\sum_{t=1}^{n} \alpha_t (\sum_{j=1}^{k} \exp(Wx)_j w_{jt} - (\sum_{j=1}^{k} \exp(Wx)_j) w_{st}) \\ \geqslant& 0.\end{aligned} \tag{3.9}$$

The last step uses Eq. (3.4) and the fact that, $x \cdot sign(x) \geqslant 0$. Now the theorem is proved.■

### 3.2. Case 2. Two linear layers followed by softmax output layers, with ReLU Activation in the middle.

$$\hat{y} = \text{softmax}(V\sigma(Wx)), \tag{3.10}$$

again, we consider the cross entropy loss:

$$L(x, y) = \text{crossentropy}(\hat{y}, y) = -\sum_{j=1}^{k} y_j \cdot \ln \hat{y}_j = -\ln \hat{y}_s,$$

where $W \in R^{l \times n}, V \in R^{k \times l}, x \in R^n, y \in R^k$. $\sigma$ is the ReLU activation.

Theorem 3.2. *For the neural nets defined in Eq. (3.10) and any input-output pair $(x, y)$. Let $x'$ be the adversarial of $x$ by applying FGSM attack to Eq. (3.10), i.e.,*

$$x' = x + \epsilon \cdot \mathrm{sign}(\nabla_x L(x, y)).$$

*Suppose every element of $Wx$ is non-zero, if $\epsilon < \frac{|Wx|_{min}}{\|W\|_\infty}$, then we have:*

$$L(x, y) \leqslant L(x', y).$$

*Here, $|Wx|_{min}$ is the smallest element among the absolute values of $Wx$, $\|W\|_\infty$ is the infinity norm of matrix $W$, i.e.,*

$$||W||_\infty = \max_i(|w_{i1}| + |w_{i2}| + \cdots + |w_{in}|).$$

*Proof.* Let $T \doteq V\sigma W$, and $\hat{y} = \mathrm{softmax}(Tx)$. First, we introduce a simple lemma.

Lemma 3.3. *For $j = 1, 2, \cdots, l$, $\mathrm{sign}(Wx)_j = sign(Wx')_j$.*

*Proof.* Let $x' = x + \delta$, every element of $\delta$ is one of $\epsilon, -\epsilon$, $0$. Since

$$(Wx')_j = (Wx)_j + (W\delta)_j,$$

and :

$$\begin{aligned} |(W\delta)_j| &= |\sum_{i=1}^{n} w_{ji}\delta_i| \leqslant \sum_{i=1}^{n} |w_{ji}| \cdot |\delta_i| \\ &\leqslant \sum_{i=1}^{n} \epsilon |w_{ji}| = \epsilon \cdot \sum_{i=1}^{n} |w_{ji}| \\ &\leqslant \epsilon \|W\|_\infty < |Wx|_{min} \leqslant |(Wx)_j|. \end{aligned} \tag{3.11}$$

Therefore: $(Wx)_j$ and $(Wx')_j$ have the same sign. ■

We go back to proof the Theorem. (3.2). Let us define the following index set

$$A \doteq \{i : (Wx)_i > 0\} = \{i : (Wx')_i > 0\}.$$

Then we can express the operator $T$ as:

$$\begin{aligned} (Tx)_j &= \sum_{t=1}^{l} v_{jt}\sigma(w_{t1}x_1 + w_{t2}x_2 + \cdots + w_{tn}x_n) \\ &= \sum_{t \in A} v_{jt}(w_{t1}x_1 + w_{t2}x_2 + \cdots + w_{tn}x_n), \end{aligned} \tag{3.12}$$

So, the operator $T$ is a linear operator.

The loss function can be rewritten as

$$\begin{aligned} L(x, y) &= \mathrm{crossentropy}(\mathrm{softmax}(Tx), y) \\ &= -\ln(\mathrm{softmax}(Tx)_s) \\ &= -\ln(\frac{\exp(Tx)_s}{\exp(Tx)_1 + \exp(Tx)_2 + \cdots + \exp(Tx)_k}). \end{aligned} \tag{3.13}$$

Therefore, this case can be reduced to Case 1, where our model consists of a linear layer followed by softmax output layer. Similar to the previous case, every element of the adversarial image can be written as:

$$\begin{aligned} x_i' &= x_i + \epsilon \cdot \operatorname{sign}(\frac{\partial}{\partial x_i} L(x,y)) \\ &= x_i + \epsilon \cdot \alpha_i, \end{aligned} \tag{3.14}$$

Replacing $w_{ji}$ with $\sum_{a \in A} v_{ja} w_{ai}$ in Equation 3.4:

$$\begin{aligned} \alpha_i &= \operatorname{sign}(\frac{\partial}{\partial x_i} L(x,y)) \\ &= \operatorname{sign}(\sum_{j=1}^{k} \exp(Tx)_j (\sum_{a \in A} v_{ja} w_{ai}) - (\sum_{j=1}^{k} \exp(Tx)_j) \sum_{a \in A} v_{sa} w_{ai}). \end{aligned} \tag{3.15}$$

Completely similar to Case 1, after using Jensen's Inequality, we can get this theorem proved.

Remark: The usage of the upper bound of $\epsilon$ is to make the operator $T = V\sigma W$ a locally linear transformation. So that the Case 2 is reduced to Case 1 which is previously proved. ■

### 3.3. Case 3. Multiple linear and softmax output layers, with all ReLU activations.

From Case 2, we note that when $\epsilon$ is small enough, everything inside the ReLU activation does not change the sign after the FGSM attack. Under this condition, the part before the softmax output layer can be treated as a linear function. Therefore, according to our proof in Case 1 where the neural nets consists of only a single linear layer before the softmax, the theorem in Case 2 is undoubtedly correct. Therefore, we can generalize Case 2 to the neural nets consists of multiple linear and a softmax output layers, with all activations between linear layers set be ReLU.

Denote our training neural network as:

$$\hat{y} = softmax(W_L \sigma W_{L-1} \sigma \cdots \sigma(W_1 x)) \tag{3.16}$$

and we denote:

$$Tmp_1 = W_1 x, \quad Tmp_2 = W_2 \sigma(Tmp_1), \quad \cdots \quad , \quad Tmp_L = W_L \sigma(Tmp_{L-1})$$

Here, $L$ is the layer number of our neural network. $W_i \quad (1 \leqslant i \leqslant L)$ are matrices and $\sigma$ stands for the ReLU activation function. Besides, $Tmp_i \quad (1 \leqslant i \leqslant L)$ are the intermediate results in the neural network.

Therefore we give our general theorem.

Theorem 3.4. *For the training neural network 3.16 and any input and output $x, y$. Let $x'$ is the attacking result of the original input $x$ with FGSM:*

$$x' = x + \epsilon \cdot sign(\nabla_x L(x,y,W))$$

*Then, assume every element of $Tmp_i$ $(1 \leqslant i \leqslant L)$ is non-zero and $\epsilon$ is sufficiently small so that every element in all $Tmp_i$ do not change their sign after the $\epsilon$-FGSM attack, then we have:*

$$L(x, y) \leqslant L(x', y)$$

*In other words, as long as $\epsilon$ is sufficiently small, after using FGSM attack to replace the original input $x$ with $x'$, the loss function will surely increase, no matter what $x, y, W_i$ $(1 \leqslant i \leqslant L)$ are.*

The proof of the theorem is similar to the one in Case 2. Since every element in all $Tmp_i$ do not change their sign, therefore $W_L\sigma \cdots \sigma(W_1x)$ can be seen as a linear function during this attack. In this condition, this problem is equivalent to Case 1.

Finally, we give an upper bound of $\epsilon$ to satisfy the condition of the theorem above.

$$\epsilon < \min_{1\leqslant j\leqslant L} \frac{|Tmp_j|_{min}}{\|W_j\|_\infty}$$

**3.4. Remark on the convolutional layer.** In all the cases above, there is an assumption that all the layers are linear. We can also generalize this to convolutional layers. Note for a convolutional layer $h$ and an input matrix $X$, when we flatten $X$ and the result $h(X)$ to a column vector, $h$ is also linear. Therefore the convolutional layers can be regarded as a linear layer when we flatten all the input and intermediate matrices.

Overall, when the neural nets consists of linear or convolutional layers, with a softmax output layer and ReLU activation, then the efficacy of FGSM attack can be guaranteed as long as the $\epsilon$ is sufficiently small.

**4. Targeted Fast Gradient Sign Method.** In this section, we consider the efficacy of the targeted adversarial attack with FGSM. Given any input $x \in R^n$ and its corresponding one-hot label vector $y \in R^k$, we want to attack it so that the new output falls into the $t$-th category. Considering the following targeted FGSM

$$x' = x - \epsilon \cdot (\nabla_x L(x, e_t)), \tag{4.1}$$

where $e_t$ is the one-hot vector of class $t$.

**4.1. Case 1. A linear layer followed by a softmax output layer.** Again, we first consider a very simple neural nets,

$$\hat{y} = \mathrm{softmax}(Wx) \tag{4.2}$$

with cross-entropy loss

$$L(x, y) = \mathrm{crossentropy}(\hat{y}, e_t) = -\sum_{j=1}^{k} y_j \cdot \ln(e_t)_j = -\ln \hat{y}_t$$

Theorem 4.1. *For the neural nets define by Eq. (4.2), any input-output pair $(x, y)$, and the target label $t$. Let $x'$ be the adversarial of $x$ resulting from the targeted FGSM attack, i.e.,*

$$x' = x - \epsilon \cdot \mathrm{sign}(\nabla_x L(x, e_t)).$$

*Under the assumption that $\nabla_x L(x, e_t)$ has no zero elements, for any:*

$$\epsilon < \min_i \frac{1}{\|W\|_\infty} \ln\left(1 + \frac{|\sum_{j=1}^k \exp(Wx)_j w_{ji} - w_{ti}(\sum_{j=1}^k \exp(Wx)_j)|}{\sum_{j=1}^k |w_{ji} - w_{ti}| \cdot \exp(Wx)_j}\right)$$

*we have:*

$$L(x, e_t) \geqslant L(x', e_t).$$

*Proof.* The loss associated with the target label $t$ for any given image-label pair $(x, y)$ is

$$\begin{aligned} L(x, e_t) &= \text{crossentropy}(\text{softmax}(Wx), e_t) \\ &= -\ln(\text{softmax}(Wx)_t) \\ &= -\ln(\frac{\exp(Wx)_t}{\exp(Wx)_1 + \exp(Wx)_2 + \cdots + \exp(Wx)_k}), \end{aligned} \tag{4.3}$$

where, $e_t$ is the one hot vector for $t$-th class.

In fact,

$$\begin{aligned} x'_i &= x_i - \epsilon \cdot \text{sign}(\frac{\partial}{\partial x_i} L(x, e_t)) \\ &= x_i - \epsilon \cdot \alpha_i, \end{aligned} \tag{4.4}$$

where,

$$\begin{aligned} \alpha_i &= \text{sign}(\frac{\partial}{\partial x_i} L(x, e_t)) \\ &= \text{sign}(-\frac{\sum_{j=1}^k \exp(Wx)_j}{\exp(Wx)_t} \cdot \\ &\quad \frac{\exp(Wx)_t w_{ti}(\sum_{j=1}^k \exp(Wx)_j) - \exp(Wx)_t \sum_{j=1}^k \exp(Wx)_j w_{ji}}{(\sum_{j=1}^k \exp(Wx)_j)^2}) \\ &= -\text{sign}(w_{ti}(\sum_{j=1}^k \exp(Wx)_j) - \sum_{j=1}^k \exp(Wx)_j w_{ji}) \\ &= \text{sign}(\sum_{j=1}^k \exp(Wx)_j w_{ji} - w_{ti}(\sum_{j=1}^k \exp(Wx)_j)). \end{aligned} \tag{4.5}$$

According to the assumption, the derivative above is nonzero. Therefore, if $\epsilon$ is sufficiently small, we have

$$\begin{aligned} &\text{sign}(\sum_{j=1}^k \exp(Wx)_j w_{ji} - w_{ti}(\sum_{j=1}^k \exp(Wx)_j)) \\ =&\text{sign}(\sum_{j=1}^k \exp(Wx')_j w_{ji} - w_{ti}(\sum_{j=1}^k \exp(Wx')_j)). \end{aligned} \tag{4.6}$$

Then, under this condition:

$$\alpha_i = \text{sign}(\sum_{j=1}^{k} \exp(Wx')_j w_{ji} - w_{ti}(\sum_{j=1}^{k} \exp(Wx')_j)). \tag{4.7}$$

We would like to give a upper bound of $\epsilon$ such that Eq. (4.6) holds. Actually, since every entry of $x$ and $x'$ has a difference at most $\epsilon$, then $|(Wx)_j - (Wx')_j| \leqslant \|W\|_\infty \epsilon$. Therefore:

$$|\exp(Wx)_j - \exp(Wx')_j| \leqslant \exp(Wx)_j(\exp(\|W\|_\infty \epsilon) - 1).$$

Let

$$A = \sum_{j=1}^{k} \exp(Wx)_j w_{ji} - w_{ti}(\sum_{j=1}^{k} \exp(Wx)_j),$$

$$B = \sum_{j=1}^{k} \exp(Wx')_j w_{ji} - w_{ti}(\sum_{j=1}^{k} \exp(Wx')_j).$$

So after the attack, the difference

$$\begin{aligned} |A - B| &\leqslant \sum_{j=1}^{k} |w_{ji} - w_{ti}| \cdot |\exp(Wx)_j - \exp(Wx')_j| \\ &\leqslant \sum_{j=1}^{k} |w_{ji} - w_{ti}| \cdot \exp(Wx)_j(\exp(\|W\|_\infty \epsilon) - 1) \\ &= (\exp(\|W\|_\infty \epsilon) - 1) \sum_{j=1}^{k} |w_{ji} - w_{ti}| \cdot \exp(Wx)_j. \end{aligned} \tag{4.8}$$

When $\epsilon$ satisfies

$$\begin{aligned} \epsilon &< \frac{1}{\|W\|_\infty} \ln{(1 + \frac{|A|}{\sum_{j=1}^{k} |w_{ji} - w_{ti}| \cdot \exp(Wx)_j})} \\ &= \frac{1}{\|W\|_\infty} \ln{(1 + \frac{|\sum_{j=1}^{k} \exp(Wx)_j w_{ji} - w_{ti}(\sum_{j=1}^{k} \exp(Wx)_j)|}{\sum_{j=1}^{k} |w_{ji} - w_{ti}| \cdot \exp(Wx)_j})}. \end{aligned} \tag{4.9}$$

In summary, when

$$\epsilon < \min_i \frac{1}{\|W\|_\infty} \ln{(1 + \frac{|\sum_{j=1}^{k} \exp(Wx)_j w_{ji} - w_{ti}(\sum_{j=1}^{k} \exp(Wx)_j)|}{\sum_{j=1}^{k} |w_{ji} - w_{ti}| \cdot \exp(Wx)_j})}$$

Equation 4.7 holds.

Now, we show that $L(x, e_t) \geqslant L(x', e_t)$

$$\begin{aligned} & \frac{\exp(Wx)_t}{\sum_{j=1}^k \exp(Wx)_j} \leqslant \frac{\exp(Wx')_t}{\sum_{j=1}^k \exp(Wx')_j} \\ \Leftrightarrow & \frac{\exp(Wx)_t}{\exp(Wx')_t} \leqslant \frac{\exp(Wx)_1 + \exp(Wx)_2 + \cdots + \exp(Wx)_k}{\sum_{j=1}^k \exp(Wx')_j} \\ \Leftrightarrow & \frac{\exp(Wx)_t}{\exp(Wx')_t} \leqslant \sum_{j=1}^k \text{softmax}(Wx')_j \cdot \frac{\exp(Wx)_j}{\exp(Wx')_j} \\ \Leftrightarrow & \exp(\epsilon W\alpha)_t \leqslant \sum_{j=1}^k \text{softmax}(Wx')_j \cdot \exp(\epsilon W\alpha)_j. \end{aligned} \tag{4.10}$$

This is exactly Equation 3.5, which has been previously proved. ■

### 4.2. Case 2. Two linear and softmax output layers, with ReLU activation.

In this part, we consider the neural nets with two linear and softmax output layers, with ReLU activation

$$\hat{y} = \text{softmax}(V\sigma(Wx)), \tag{4.11}$$

the loss is

$$L(x, e_t) = \text{crossentropy}(\hat{y}, e_t) = -\sum_{j=1}^k (e_t)_j \cdot \ln \hat{y}_j = -\ln \hat{y}_t.$$

Here: $W \in R^{l\times n}, V \in R^{k\times l}, x \in R^n, y \in R^k$. $\sigma$ is the ReLU activation function.

Theorem 4.2. *For the neural net defined in Eq. (4.11) and any input-output pair $(x, y)$. Let $x'$ be the adversarial of $x$ resulting from targeted FGSM attack*

$$x' = x - \epsilon \cdot sign(\nabla_x L(x, e_t)),$$

*Suppose the derivative above has no zero elements, every element of $Wx$ is non-zero, and $\epsilon$ is smaller than an upper bound which will be written at the end of the proof below, we have*

$$L(x, e_t) \geqslant L(x', e_t).$$

*Proof.* Let $T \doteq V\sigma W$, and $\hat{y} = \text{softmax}(Tx)$. According to Lemma 3.3,if:

$$\epsilon < \frac{|Wx|_{min}}{\|W\|_\infty},$$

then $\text{sign}(Wx)_j = \text{sign}(Wx')_j$.

Denote the index set

$$A = \{i : (Wx)_i > 0\} = \{i : (Wx')_i > 0\}.$$

Then we can express the operator $T$ as:

$$\begin{aligned}(Tx)_j &= \sum_{t=1}^{l} v_{jt}\sigma(w_{t1}x_1 + w_{t2}x_2 + \cdots + w_{tn}x_n) \\ &= \sum_{t\in A} v_{jt}(w_{t1}x_1 + w_{t2}x_2 + \cdots + w_{tn}x_n),\end{aligned} \tag{4.12}$$

so, similar to Case 3.2, the operator $T$ can be regarded as a locally linear operator. And we only need to replace the $w_{ji}$ in Case 4.1 with $\sum_{a\in A} v_{ja}w_{ai}$. So once $\epsilon$ is controlled by the upper bound $U = min(U_1, U_2)$, then the theorem is correct. Here:

$$U_1 = \frac{|Wx|_{min}}{\|W\|_\infty}$$

$$U_2 = \min_i \frac{1}{\|T\|_\infty} \ln{(1 + \frac{|\sum_{j=1}^k \exp(Tx)_j t_{ji} - t_{ti}(\sum_{j=1}^k \exp(Tx)_j)|}{\sum_{j=1}^k |t_{ji} - t_{ti}| \cdot \exp(Tx)_j})}$$

where:

$$t_{ji} = \sum_{a\in A} v_{ja}w_{ai}$$

■

### 4.3. Case 3. Multiple linear and softmax output layers, with all ReLU activations.

From Case 2, we note that when $\epsilon$ is small enough, we can guarantee that everything inside the ReLU activation does not change their sign after the $\epsilon$-FGSM attack. Under this condition, the part before the softmax can be treated as a linear function. Therefore, according to our proof in Case 1 where the neural nets consists of only a single linear layer before the softmax, the theorem in Case 2 remains correct. Therefore, we can generalize Case 2 to the neural nets consists of multiple linear and softmax output layers, with all activation functions between linear layers be ReLU. Consider the neural nets

$$\hat{y} = \text{softmax}(W_L\sigma W_{L-1}\sigma \cdots \sigma(W_1x)) \tag{4.13}$$

and we denote:

$$\text{Tmp}_1 \doteq W_1x, \quad \text{Tmp}_2 \doteq W_2\sigma(\text{Tmp}_1), \quad \cdots \quad , \quad \text{Tmp}_L = W_L\sigma(\text{Tmp}_{L-1})$$

Here, $L$ is the number of layers. $W_i \;\; (1 \leqslant i \leqslant L)$ are matrices and $\sigma$ stands for the ReLU activation. Moreover, $\text{Tmp}_i \;\; (1 \leqslant i \leqslant L)$ are the intermediate results in the neural network.

For the neural nets defined in Eq. (4.13), we give the following theorem

Theorem 4.3. *For the neural net defined by Eq. (4.13) and any input-output pair $(x, y)$. Let $x'$ be the adversarial of $x$ by the targeted FGSM, i.e.,*

$$x' = x - \epsilon \cdot sign(\nabla_x L(x, e_t)).$$

*Suppose the derivative above has no zero elements, and every element of* $\text{Tmp}_i$ $(1 \leqslant i \leqslant L)$ *is non-zero and $\epsilon$ is sufficiently small so that every element in all* $\text{Tmp}_i$ *do not change their sign after the $\epsilon$-targeted FGSM attack, then we have*

$$L(x, e_t) \geqslant L(x', e_t)$$

The proof of the theorem is similar to the one in Case 2. Since every element in all $\mathrm{Tmp}_i$ does not change their sign, therefore $W_L\sigma\cdots\sigma(W_1x)$ can be seen as a linear function during this attack. In this condition, this problem is equivalent to Case 1.

Remark: For the same reason as Section 3.4, this conclusion can be also generated to convolutional layers.

To sum up, when our training neural network consists of linear or convolution layers, with a softmax output layer and all acivations ReLU, then the efficacy of targeted FGSM adversarial attack can be guaranteed theoretically as long as the $\epsilon$ is sufficiently small.

## 5. CW-L2 Targeted Adversarial Attack.

**5.1. Models and attack.** In this section, we consider the simple neural nets which consists of a linear and softmax output layers, i.e.,

$$\hat{y} = \mathrm{softmax}(Wx). \tag{5.1}$$

We consider the simplified CW-L2 attack, in which we relax the constraint that pixel values are between 0 and 1, in this case, the perturbation $\delta$ is the solution of the optimization problem

$$\arg\min_{\delta} \|\delta\|_2^2 + c\cdot g(x+\delta), \tag{5.2}$$

where,

$$g(x) = \max\left(\max_{i\neq t}(Z(x)_i) - Z(x)_t, 0\right)$$

and $Z(x)$ is the logit vector of the input $x$, which in our one-layer network, means that:

$$Z(x) = Wx \in R^k,$$

$c \geqslant 0$ is the Lagrangian multiplier.

**5.2. Task1. Increasing relative probability of the target label.**

Theorem 5.1. *For the CW-L2 attack, when the Lagrangian multiplier:*

$$c < \min_{j\neq y} \frac{((Wx)_y - (Wx)_j)^2}{\|W_{y,:} - W_{j,:}\|_2^2 \cdot ((Wx)_y - (Wx)_t)},$$

*where, $W_{i,:}$ is the $i$-th row vector of $W \in R^{k\times n}$, $1 \leqslant i \leqslant k$, and $y$ is the label of the original input $x$, i.e., $(Wx)_y$ is the largest among all the $(Wx)_i$, $(1 \leqslant i \leqslant k)$. Then,*

$$\frac{P(f(x)=t)}{P(f(x)=y)} \leqslant \frac{P(f(x+\delta)=t)}{P(f(x+\delta)=y)}$$

*In other words, the attack increases the ratio between probability of the $t$-th and $y$-th labels.*

*Proof.* Let $x' \doteq x+\delta$. We introduce a lemma first and then consider two different cases.

Lemma 5.2. *If $(Wx')_y \leqslant (Wx')_i$, then we have*

$$\|x'-x\|_2^2 > c\cdot((Wx)_y - (Wx)_t)$$

*Proof.* In fact, if $(Wx')_y \leqslant (Wx')_i$, then

$$\begin{aligned} &(Wx)_y + (W\delta)_y \leqslant (Wx)_i + (W\delta)_i \\ \Rightarrow &(W_{y,:} - W_{i,:})^T \delta \leqslant -((Wx)_y - (Wx)_i) \\ \Rightarrow &|(W_{y,:} - W_{i,:})^T \delta| \geqslant (Wx)_y - (Wx)_i \end{aligned} \tag{5.3}$$

According to the Cauchy-Schwarz Inequality, we have

$$|(W_{y,:} - W_{i,:})^T \delta|^2 \leqslant \|W_{y,:} - W_{i,:}\|_2^2 \cdot \|\delta\|_2^2$$

Therefore

$$\begin{aligned} \|\delta\|_2^2 &\geqslant \frac{((Wx)_y - (Wx)_i)^2}{\|W_{y,:} - W_{i,:}\|_2^2} \\ &> c \cdot ((Wx)_y - (Wx)_t) \end{aligned} \tag{5.4}$$

Till now, the lemma has been proved. ■

Let us go back to the theorem, we consider the following two cases.

Case 1. $(Wx')_y$ is the largest among all the $(Wx')_i$, $(1 \leqslant i \leqslant k)$.

$$\begin{aligned} L(x') &= \|\delta\|_2^2 + c \cdot g(x') \\ &= \|\delta\|_2^2 + c \cdot \max(\max_{i \neq t}(Wx')_i - (Wx')_t, 0) \\ &= \|\delta\|_2^2 + c \cdot ((Wx')_y - (Wx')_t) \\ &= \sum_{j=1}^{n} \delta_j^2 + c \cdot \sum_{j=1}^{n} (w_{yj} - w_{tj})(x_j + \delta_j) \end{aligned} \tag{5.5}$$

In order to minimize $L(x')$, it is easy to find that:

$$\delta_j = -\frac{c}{2}(w_{yj} - w_{tj}), \quad 1 \leqslant j \leqslant n$$

On the other hand, if the equation above holds, then:

$$\begin{aligned} \|\delta\|_2^2 &= \frac{c^2}{4} \|W_{y,:} - W_{t,:}\|_2^2 \\ &< c^2 \cdot \|W_{y,:} - W_{t,:}\|_2^2 \\ &< c((Wx)_y - (Wx)_t) \end{aligned} \tag{5.6}$$

*(The last line is because of the upper bound of $c$. Let $j = t$, we know that:*

$$\begin{aligned} c &< \frac{((Wx)_y - (Wx)_t)^2}{\|W_{y,:} - W_{i,:}\|_2^2 \cdot ((Wx)_y - (Wx)_t)} \\ &= \frac{(Wx)_y - (Wx)_t}{\|W_{y,:} - W_{i,:}\|_2^2} \end{aligned} \tag{5.7}$$

*That makes the inequality above proved.)*

Then, according to Lemma. 5.2, $(Wx')_y > (Wx')_i$ holds for all $i \neq y$, which is exactly the condition of Case 1. So under Case 1, the minimal value of $L$ is:

$$\begin{aligned} L^* &= c \cdot \sum_{j=1}^{n} (w_{yj} - w_{tj}) x_j - \frac{c^2}{4} \sum_{j=1}^{n} (w_{yj} - w_{tj})^2 \\ &= c \cdot ((Wx)_y - (Wx)_j) - \frac{c^2}{4} \sum_{j=1}^{n} (w_{yj} - w_{tj})^2 \end{aligned} \tag{5.8}$$

Case 2. $(Wx')_y$ is not the largest among all the $(Wx')_i \quad (1 \leqslant i \leqslant k)$. Then, according to the lemma, we know that:

$$\|\delta\|_2^2 > c((Wx)_y - (Wx)_t)$$

Hence,

$$L(x') \geqslant \|\delta\|_2^2 > c \cdot ((Wx)_y - (Wx)_t) \geqslant L^*.$$

Therefore, in this Case 2. The $x'$ is not the solution of our optimization problem.

To sum up, the accurate solution of the optimization problem is:

$$\delta_j = -\frac{c}{2}(w_{yj} - w_{tj})$$

Then:

$$\frac{P(f(x) = t)}{P(f(x) = y)} = \frac{\text{softmax}(Wx)_t}{\text{softmax}(Wx)_y} = \exp((Wx)_t - (Wx)_y)$$

and the same reason applies:

$$\frac{P(f(x') = t)}{P(f(x') = y)} = \exp((Wx')_t - (Wx')_y)$$

So, we only have to prove the following inequality:

$$\begin{aligned} &(Wx)_t - (Wx)_y \leqslant (Wx')_t - (Wx')_y \\ \Leftrightarrow &\sum_{j=1}^{n} (w_{tj} - w_{yj})(x'_j - x_j) \geqslant 0 \\ \Leftrightarrow &\sum_{j=1}^{n} (w_{tj} - w_{yj})\delta_j \geqslant 0 \end{aligned} \tag{5.9}$$

Since $\delta_j = -\frac{c}{2}(w_{yj} - w_{tj})$, the inequality above is obvious. ■

**5.3. Task2. Analysis of irrelevant labels.** In this part, we argue that for any third label $k \neq y, t$, the relative probability may either increase or decrease. For $c$ satisfying the condition in Theorem. 5.1, we can use the same way to prove that

$$\begin{aligned} &\frac{P(f(x) = k)}{P(f(x) = y)} < \frac{P(f(x') = k)}{P(f(x') = y)} \\ \Leftrightarrow &(Wx)_k - (Wx)_y < (Wx')_k - (Wx')_y \\ \Leftrightarrow &(W_{k,:} - W_{y,:}) \cdot (x' - x) > 0 \end{aligned} \tag{5.10}$$

Since $x' - x = -\frac{c}{2}(W_{y,:} - W_{t,:})^T$, the inequality above is equivalent to:

$$(W_{k,:} - W_{y,:}) \cdot (W_{y,:} - W_{t,:}) < 0.$$

And it is obvious that the inequality may be either right or wrong. For example, consider a 3D example, let $W_{y,:} = (1, 1, 1), W_{t,:} = (1, 0, 1)$. Then, if $W_{k,:} = (0, 2, 0)$, the inequality is wrong; if $W_{k,:} = (0, -1, 0)$, it is right.

## 6. Numerical Results.

We verify the above theoretical results by applying the aforementioned adversarial attacks to attack the ResNet56. We train ResNet56 on the CIFAR10 follow the standard procedure used by [9].

### 6.1. Untargeted FGSM Attack.

We first consider single and multiple iterations of the untergated FGSM attack. In Section. 3, we proved that for small attack strength $\epsilon$, the attack will fool the neural nets, this is validated by the numerical results shown in Fig. 2 (a), when $\epsilon < 0.1$, as $\epsilon$ increases, the classification accuracy decays monotonically. Moreover, Fig. 2 (b) shows that for a fixed $\epsilon$, as the number of iteration increases, the success rate of adversarial attack increases. This shows that IFGSM is a stronger attack than single step FGSM.

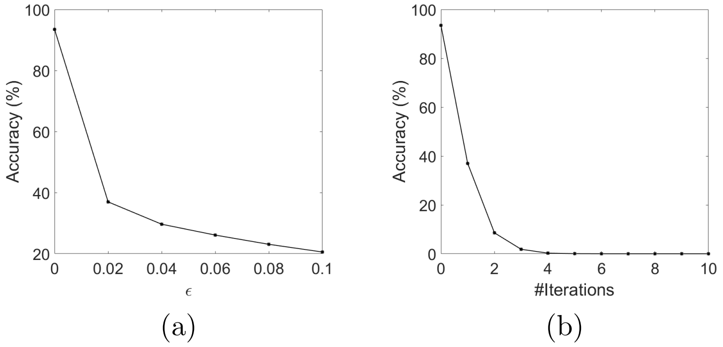


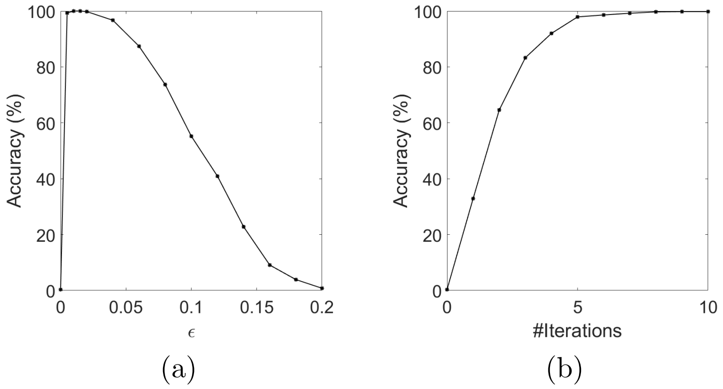
**Figure 2.** *Testing accuracy under the untargted FGSM attack for ResNet56 on CIFAR10 benchmark. (a): attack strength $\epsilon$ v.s. accuracy with only one iteration. (b): number of iterations v.s. accuracy with $\epsilon = 0.02$.*

### 6.2. Targeted FGSM Attack.

In this part, we verify the efficacy of the targeted FGSM attack numerically. We apply IFGSM to attack the cat (labeled 4 in CIFAR10) to dog (labeled 6 in CIFAR10). In all the experiments, we set $\alpha = 0.1$. Theoretically, in Section. 4, we showed that for any CNN with a softmax output activation and ReLU activation, within the regime of small $\epsilon$, targeted FGSM will fooled neural nets to classify cat to dog. Numerically, again, we consider the ResNet56. Figure. 3 (a) shows that for 10 iterations attack, when $\epsilon$ is sufficiently small ($\leq 0.015$), as $\epsilon$ increases, the success rate raises. Once $\epsilon > 0.015$, the success rate decays as $\epsilon$ increases. Furthermore, we consider impact of the number of iterations in the targeted adversarial attack. As shown in Figure. 3 (b), the success rate increases monotonically as the number of iterations increases.

### 6.3. CW-L2 Attack.

We proved, in Theorem. 5.1, that when CW-L2 attack is applied to fool the neural nets to classify an image to a target class. The rate of the classification probability between the target and the ground-truth labels increases. We continue to attack



(a) (b)

**Figure 3.** *Success rate on fooling ResNet56 to classify cat to dog under the targted FGSM attack for CIFAR10 benchmark. (a): attack strength $\epsilon$ v.s. success rate with 10 iteration. (b): number of iterations v.s. accuracy with $\epsilon = 0.02$.*

ResNet56 to mis-classify cat to dog. We apply 10 iterations of Adam optimizer with, $c = 10$. $\kappa = 0$, and learning rate to be 0.01 to optimize the CW-L2 attack objective (Eq. (2.9)). In Fig. 4, we depict the averaged probability of ResNet56 to classify the cat images before and after the adversarial attack over 1000 images. 750 images successfully attacked to the dog class. From Fig. 4, we see that the CW-L2 attack shift the probability density peak from class 4 (cat) to class 6 (dog). It is also interesting to notice that the probability of the images been classified to class 9 has a probability 0.0048 before attack, while it decreases to 0.004 after the adversarial attack. This further validates the theoretical conclusion we achieved in Section. 5.

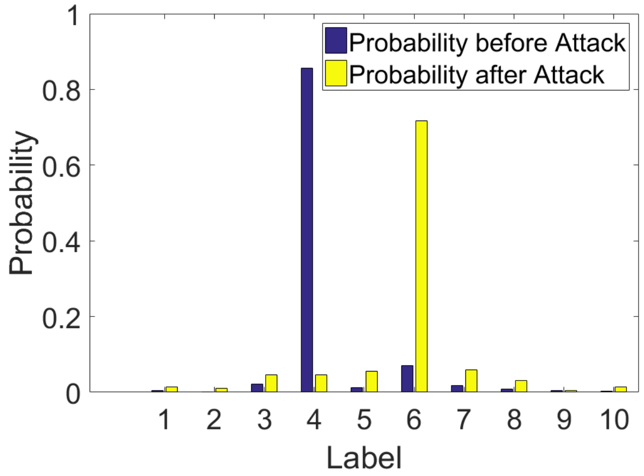


**Figure 4.** *The probability distribution of ResNet56 in classifying cat (labeled with 4) into each classes. Blue bars: before attack. Yellow bars: after CW-L2 attack.*

**7. Concluding Remarks.** In this paper we proved that the untargeted FGSM can fool any convolutional neural nets (CNN) with ReLU activation for small attack strength; within a specific regime, the targeted FGSM can mislead any CNN with ReLU activation to classify any given image into any prescribed class. For a two-layer neural nets, a linear layer followed by the softmax output activation, we show that the CW-L2 attack increases the ratio of the classification probability between the target and ground truth labels. A large amount of numerical results conform our theoretical results. Quantifying the relation between the attack strength $\epsilon$ and the loss is under our further exploration. Analyzing the influence of the recently proposed Laplacian smoothing gradient descent [23] on improving adversarial robustness is also under our investigation.

**8. Acknowledgments.** This material is based on research sponsored by the Air Force Research Laboratory: DARPA under agreement number FA8750-18-2-0066, MURI under the grant number FA9550-18-1-0502, and FA9550-18-1-0167. And by the U.S. Department of Energy, Office of Science, and by National Science Foundation, under grant numbers DOE-SC0013838 and DMS-1554564, (STROBE). And by the NSF DMS-1737770 and the Simons foundation. And Office of Naval Research: ONR:N00014-18-1-2527. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon.

## REFERENCES

[1] N. Akhtar and A. Mian, *Threat of adversarial attacks on deep learning in computer vision: A survey*, arXiv preprint arXiv:1801.00553, (2018).

[2] A. Athalye, N. Carlini, and D. Wagner, *Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples*, International Conference on Machine Learning, (2018).

[3] W. Brendel, J. Rauber, and M. Bethge, *Decision-based adversarial attacks: Reliable attacks against black-box machine learning models*, arXiv preprint arXiv:1712.04248, (2017).

[4] N. Carlini and D. Wagner, *Towards evaluating the robustness of neural networks*, IEEE European Symposium on Security and Privacy, (2016), pp. 39–57.

[5] I. J. Goodfellow, J. Shlens, and C. Szegedy, *Explaining and harnessing adversarial examples*, arXiv preprint arXiv:1412.6275, (2014).

[6] K. Grosse, N. Papernot, P. Manoharan, M. Backes, and P. McDaniel, *Adversarial perturbations against deep neural networks for malware classification*, arXiv preprint arXiv:1606.04435, (2016).

[7] A. Guisti, J. Guzzi, D. Ciresan, F. He, J. Rodriguez, F. Fontana, M. Faessler, C. Forster, J. Schmidhuber, G. D. Carlo, and et al, *A machine learning approach to visual perception of forecast trails for mobile robots*, IEEE Robotics and Automation Letters, (2016), pp. 661–667.

[8] C. Guo, M. Rana, M. Cisse, and L. van der Maaten, *Countering adversarial images using input transformations*, in International Conference on Learning Representations, 2018, https://openreview.net/forum?id=SyJ7ClWCb.

[9] K. He, X. Zhang, S. Ren, and J. Sun., *Deep residual learning for image recognition*, in CVPR, 2016, pp. 770–778.

[10] D. Kingma and J. Ba, *Adam: A method for stochastic optimization*, arXiv preprint arXiv:1412.6980, (2014).

[11] A. Kurakin, I. J. Goodfellow, and S. Bengio, *Adversarial examples in the physical world*, arXiv preprint arXiv:1607.02533, (2016).

[12] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, *Towards deep learning models resistant to adversarial attacks*, in International Conference on Learning Representations, 2018, https://openreview.net/forum?id=rJzIBfZAb.

[13] S.-M. Moosavi-Dezfooli, A. Fawzi, O. Fawzi, and P. Frossard, *Universal adversarial perturbations*, in The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 2017.

[14] N. Papernot, P. McDaniel, S. Jha, M. Fredrikson, Z. Celik, and A. Swami, *The limitations of*

*deep learning in adversarial settings*, IEEE European Symposium on Security and Privacy, (2016), pp. 372–387.

[15] N. Papernot, P. McDaniel, A. Sinha, and M. Wellman, *Sok: Towards the science of security and privacy in machien learning*, arXiv preprint arXiv:1611.03814, (2016).

[16] N. Papernot, P. McDaniel, X. Wu, S. Jha, and A. Swami, *Distillation as a defense to adversarial perturbations against deep neural networks*, IEEE European Symposium on Security and Privacy, (2016).

[17] N. Papernot, P. D. McDaniel, and I. J. Goodfellow, *Transferability in machine learning: from phenomena to black-box attacks using adversarial samples*, CoRR, abs/1605.07277 (2016), http://arxiv.org/abs/1605.07277, https://arxiv.org/abs/1605.07277.

[18] P. Samangouei, M. Kabkab, and R. Chellappa, *Defense-GAN: Protecting classifiers against adversarial attacks using generative models*, in International Conference on Learning Representations, 2018, https://openreview.net/forum?id=BkJ3ibb0-.

[19] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, and I. Goodfellow, *Intriguing properties of neural networks*, arXiv preprint arXiv:1312.6199, (2013).

[20] F. Tramr, A. Kurakin, N. Papernot, I. Goodfellow, D. Boneh, and P. McDaniel, *Ensemble adversarial training: Attacks and defenses*, in International Conference on Learning Representations, 2018, https://openreview.net/forum?id=rkZvSe-RZ.

[21] B. Wang, A. T. Lin, Z. Shi, W. Zhu, P. Yin, A. L. Bertozzi, and S. J. Osher, *Adversarial defense via data dependent activation function and total variation minimization.*, arXiv preprint arXiv:1809.08516, (2018).

[22] B. Wang, X. Luo, Z. Li, W. Zhu, Z. Shi, and S. Osher, *Deep neural nets with interpolating function as output activation.*, arXiv preprint arXiv:1802.00168, (2018).

[23] S. Osher, B. Wang, P. Yin, X. Luo, M. Pham and A. Lin, *Laplacian smoothing gradient descent.*, arXiv preprint arXiv:1806.06317, (2018).

[24] B. Wang, B. Yuan, S. Osher, and Z. Shi, *EnResNet: ResNet ensemble via the Feynman-Kac formalism.*, arXiv preprint arXiv:2480540, (2018).