# Programming-4 : Adversarial Examples for Neural Nets

Aman Raj, PID: A53247556

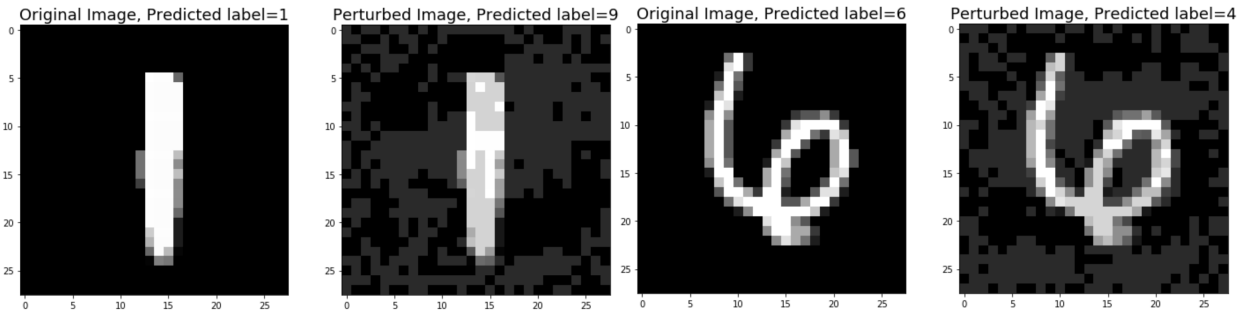
March 15, 2019

## 1 Fast Gradient Sign Method (FGSM)

We implemented Fast Gradient Sign Method to create adversarial examples for MNIST dataset. FGSM takes a trained neural network $f$ and changes the input $x$ by taking a small step $\Delta$ along which the cross entropy loss $L$ increases. Adversarial sample $\tilde{x}$ is generated by:

$$\tilde{x} = x + \epsilon \cdot \text{sign}(\Delta L(f, x, y))$$

Our Multi Layer Perceptron model is pre-trained on train split of MNIST and we report accuracy and errors on original and adversarial examples of test split only throughout this report. Accuracy on original test split is *97.6%*, accuracy on adversarial examples of test split for $\epsilon = 0.1$ is *1.4%*. Therefore, the error rate of neural network on original dataset and perturbed version is *2.4%* and *98.6%*($\epsilon = 0.1$) respectively. Figure 1, shows adversarial examples generated from some samples that have fooled the neural network leading to misclassification. In nutshell, FGSM is a white-box attack where we know the parameters of neural network and change the input such that it is misclassified.




(a) Sample from MNIST test split with label=1 (b) Sample from MNIST test split with label=6

Figure 1: Original and perturbed version of a sample and their predictions from neural network using FGSM

## 2 New Methods

We explored two different approaches to generate adversarial examples. Section 2.1 is an additive noise based method while Section 2.2 is gradient based method like FGSM which we explain in detail with pseudocode. It can be easily noted that in both methods measure of modification to input $x$ is atmost $\epsilon$.

### 2.1 Gaussian Noise Sign Method (GNSM)

We generate random Gaussian Noise with $\mu = 0$ and $\sigma = 4$ and update the input $x$ to generate adversarial example $\tilde{x}$ by: $\tilde{x} = x + \epsilon \cdot \text{sign}(\phi(\mu, \sigma))$. As we can see it's a strategy with randomness, hences we do several experiments for each $\epsilon$ and report the errors in Section 3.

### 2.2 Targeted Fast Gradient Sign Method (T-FGSM)

We propose an FGSM based method that compute a gradient step, in the direction of negative gradient with respect to a chosen target class. The notion is to do a small change to the input $x$ such that it is similar to samples of target class in image space and hence fooling the neural network to misclassify it as belonging to the target class. We call it Targeted Fast Gradient Sign Method, like FGSM this is also a white-box attack.

**Step 1** Take input *(x, y)* and perturbation $\epsilon$.

**Step 2** Do forward pass of neural network: *f(x)* to compute layer outputs ($z^i$'s) and activations ($h^i$'s).

**Step 3** Now randomly select a target label $y_t$ for adversarial attack such that $y_t \neq y$ and compute the gradient $\Delta L(f, x, y_t)$ using chain rule as we did in FGSM algorithm.

**Step 4** Generate adversarial example $\tilde{x}$ by:

$$\tilde{x} = x - \epsilon \cdot \text{sign}(\Delta L(f, x, y_t))$$

## 3 Experimental Results

The comparison of FGSM and proposed methods T-FGSM and GNSM on various values of $\epsilon$ are shown in Table 1. For Gaussian Noise Sign Method, we did several experiments namely $N = 50$ for each value of $\epsilon$ and report mean accuracy with margin of error where margin of error is calculated by:

$$error = \frac{Z_p * \sigma}{\sqrt{N}}$$

$Z_p = 1.96$ for 95% confidence level, $\sigma$ is standard deviation of sample space having $N$ samples. We also show error bars for 50-runs for $\epsilon = 0.20$ in Figure 2.

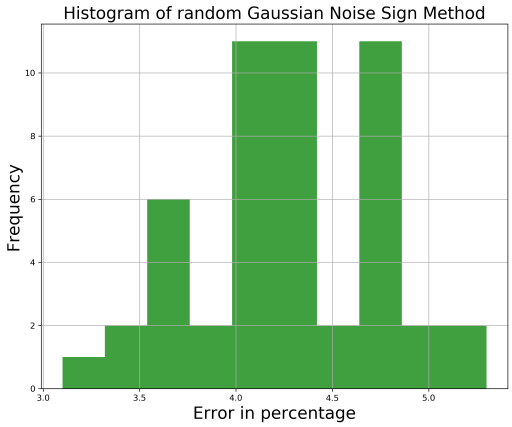



Figure 2: Error bars for 50-runs for $\epsilon = 0.20$ on test split of MNIST.

| Epsilon($\epsilon$) | Fast Gradient SM | Targeted Fast Gradient SM | Gaussian Noise SM |
|---|---|---|---|
| 0.05 | 33.5 | 60.40 | 97.50 ±0.03 |
| 0.10 | 1.40 | 11.50 | 97.20 ±0.07 |
| 0.15 | 0.20 | 1.30 | 96.67 ±0.08 |
| 0.20 | 0.00 | 0.20 | 95.71 ±0.13 |

Table 1: Accuracy of different algorihtms for generating adversarial examples on test split of MNIST.

# 4 Critical Evaluation

As we can observe from Table 1, neither Targeted-FGSM nor GNSM is a clear improvement over FGSM. However, T-FGSM is very close in performance to FGSM as the accuracy of the algorithm decreases with increasing $\epsilon$ finally reducing to zero at $\epsilon = 0.20$ which higlights with more perturbation the algorithm is getting better at fooling the network like FGSM. Gaussian Noise Sign Method is worst algorithm among all, highlighting the fact that we have to be very picky about the kind of perturbation to input $x$ to create adversarial example.

From the results in Table 1, it is clear that there is scope for improvement for proposed T-FGSM. It is to be noted that T-FGSM and FGSN both are *one-shot attacks* on neural network as we are taking single step in either positive or negative direction of the gradient. Next, we would like to try an iterative method that takes multiple steps in the gradient direction each of magnitude $\alpha = \epsilon/T$, where $T$ is number of steps. Generation of adversarial example $\tilde{x}$ for input $x$ using such iterative method over $T$ numver of updates can be given by:

$$\tilde{x}_0 = x$$

$$\tilde{x}_{t+1} = \tilde{x}_t + \alpha \cdot \text{sign}(\Delta L(f, \tilde{x}_t, y))$$