Lab 3: Adversarial Attacks on Deep Neural Networks

Tiyas Dey (td2355)

11/18/2023

Link to Colab Notebook

1 Baseline

1.1 Model

The baseline deep neural network model used is implemented in PyTorch. The model consists of the following key components:

- Flattening Layer: Transforms each 28x28 pixel 2D image into a 1D tensor of 784 elements.
- Fully Connected Layers: Three fully connected (FC) layers are used, where:
  - The first FC layer has 512 neurons, transforming the input from 784 to 512 dimensions.
  - The second FC layer, also with 512 neurons, continues the transformation.
  - The third FC layer reduces the dimensionality from 512 to 128 neurons.
- ReLU Activations: Each FC layer is followed by a ReLU activation function, introducing non-linearity into the model.
- Output Layer: A final FC layer with 10 neurons corresponds to the 10 digit classes of MNIST, producing the model's output logits.
- The model uses a CrossEntropy Loss and Adam Optimizer

1.2 Dataset

The MNIST dataset from torchvision datasets is used and the training images are normalised so that the pixels lie between [0,1]. The training dataset is further transformed into a dataloader with a batch size of 64.

1.3 Evaluation

Initailly the model achieves an accuracy of around 96.62%

2 Adversarial Attacks

2.1 FGSM Based Untargetted Attacks

The untargetted attacks are carried out for $\epsilon = [0.1, 0.2, 0.25, 0.3, 0.4, 0.5]$ The results are:

- Epsilon: 0.1 : Fraction Successful Attacks = 1148 / 9662 = 0.11881598012833781
- Epsilon: 0.2 : Fraction Successful Attacks = 2048 / 9662 = 0.2119643966052577
- Epsilon: 0.25 : Fraction Successful Attacks = 2665 / 9662 = 0.2758228110122128
- Epsilon: 0.3 : Fraction Successful Attacks = 3292 / 9662 = 0.340716207824467
- Epsilon: 0.4 : Fraction Successful Attacks = 4611 / 9662 = 0.4772303870834196
- Epsilon: 0.5 : Fraction Successful Attacks = 5868 / 9662 = 0.6073276754295177

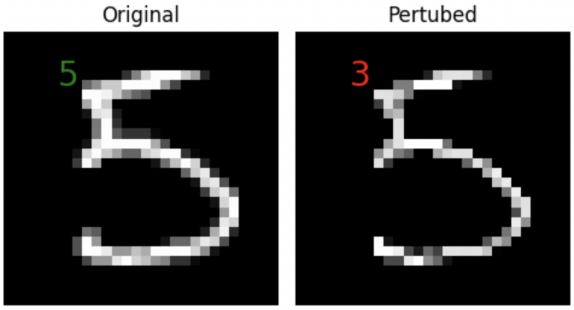

Figure 1: Untargetted Attacks

2.2 FGSM Based Targetted Attacks

The untargetted attacks are carried out for $\epsilon = [0.1, 0.2, 0.3, 0.4, 0.5]$ The results are:

- Epsilon: 0.1 : Attack Success Rate = 172/9662 = 0.01780169737114469
- Epsilon: 0.2 : Attack Success Rate = 423/9662 = 0.04377975574415235
- Epsilon: 0.3 : Attack Success Rate = 821/9662 = 0.08497205547505693
- Epsilon: 0.4 : Attack Success Rate = 1259/9662 = 0.13030428482715795
- Epsilon: 0.5: Attack Success Rate = 1683/9662 = 0.1741875388118402

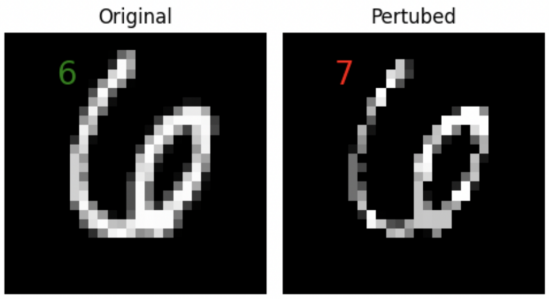

Figure 2: Targetted Attacks

3 Adversarial Retraining

3.1 Generating Pertubed Images with $\epsilon = 125/255$

- With $\epsilon = 125/255$, an attack accuracy of 1642/9662 = 0.16994411095011386 was achieved with FGSM targetted attacks.
- A custom dataset was created from these images and merged with the MNIST training dataset from torchvision, resulting it total 61642 images in training set.
- A new model was trained using the merged dataset

3.2 Robustness of Adversarially Trained DNN

With the new trained model, the attack success rates were as follows:

- Epsilon: 0.1: Fraction Successful Attacks = 4330 / 9695 = 0.44662197008767407
- Epsilon: 0.2: Fraction Successful Attacks = 4623 / 9695 = 0.4768437338834451
- Epsilon: 0.25: Fraction Successful Attacks = 4805 / 9695 = 0.4956162970603404

- Epsilon: 0.3: Fraction Successful Attacks = 4982 / 9695 = 0.5138731304796287
- Epsilon: 0.4: Fraction Successful Attacks = 5429 / 9695 = 0.5599793708096957
- Epsilon: 0.5: Fraction Successful Attacks = 5931 / 9695 = 0.6117586384734399

4 Conclusion

In this experiment, the untargetted attacks had a significant accuracy, however the targetted attacks had low accuracy and training the model with around 1600 adevrserially generated images with targetted attacks **did not improve the robustness of the model noticeably** to counter untargetted attacks.