EL-GY-9133 Machine Learning for Cyber-Security

Lab 1: Adversarial Attacks on Deep Neural Networks

Release Date: 10/17/2022; *Due Date*: Midnight, 10/31/2022

Overview

In this lab, you will investigate adversarial perturbation attacks on Deep Neural Networks using the MNIST digits dataset as a benchmark. You will then evaluate adversarial retraining as a defense against adversarial perturbations.

Dataset

The Fashion_MNIST dataset is a commonly used "toy" benchmarks for machine learning. It contains 28X28 grayscale images with a label from 10 classes, along with the associated labels. The dataset is available as part of the *tensorflow* package, which you will be using extensively in this lab.

What You Have to Do

The sample Google Colab notebook https://colab.research.google.com/github/tensorflow/docs/blob/master/site/en/tutorials/quickstart/beginner.ipynb#scrollTo=h3IKyzTCDNGo that implements a 2-layer DNN for MNIST digit classification. The DNN has a 784 (28x28) dimensional input, a 10-dimensional output (prediction probabilities for each of the 10 classes) and one hidden layer with 300 hidden neurons and ReLU activations. You will implement your attacks and defenses on this **baseline DNN**. Note that the baseline DNN first normalizes pixel values to lie between [0,1] by dividing each pixel by 255.

- **FGSM based untargeted attacks:** Your first goal is to implement FGSM based untargeted attacks using images from the *test* set on the baseline DNN. That is, your goal is to adversarially perturb each image in the test set using the following values of parameter ε = {25, 50, 75, 125}/255 **assuming pixel values are normalized in the range [0,1]**. ε is the L2 norm of the difference between the original and modified image.Report the success rate of your attack, i.e., the fraction of test images that were correctly classified by the baseline DNN that are mis-classified after adversarial perturbation for each ε

- **FGSM based targeted attacks:** Next, you will repeat Step 1 above, except this time perform ***targeted*** attacks where digit *i* is classified as (*i*+1)%10 on the baseline DNN. (Here, *i* refers to the true ground-truth label of the test images, and you can assume that the attacker has access to these labels.) As before, use the following values of the parameter *ε* = {25, 50, 75, 125}/255. Report the attack's success rate as a function of parameter ε, where success rate is defined as the fraction of test images that were correctly classified by the baseline DNN that are mis-classified after
- adversarialperturbations with label (*i*+1)%10.
- **Adversarial Retraining against Untargeted FGSM Attacks:** For this step, you can assume ε = 125/255 throughout. To defend against adversarial perturbations, the defender adversarially perturbs each image in her training set using the attacker's strategy in Step 1. She then appends the adversarially perturbed images to her training set, but using their *correct* labels. Then, the defender retrains the baseline DNN with a new training dataset containing both images from the original training dataset and the new adversarially perturbed images. We call the new DNN the **adversarially retrained DNN**.

  - Report the classification accuracy of the adversarially retrained DNN on the original test dataset that contains only clean inputs.
  - Is the adversarially retrained DNN robust against adversarial perturbations? Implement FGSM based untargeted attacks using images from the clean *test* set on the adversarially retrained DNN. Report the success rate of your attack.

- **Challenge:** Finally, you will all compete in an adversarial attack *challenge.* You are free to use any adversarial attack procedure to implement *untargeted* attacks on a 2-layer baseline DNN. You are welcome to implement IFGSM or PGD attacks for example. For this problem you will simply submit your adversarially perturbed test images from the Fashion_MNIST dataset. For each image, ε should be less than or equal to 125/255 assuming images are normalized to lie between [0,1]. We will evaluate your attacks on our own 2-layer DNN with the same architecture as in the tutorial link above --- however you do NOT have access to the weights/biases of our DNN. The expectation is that your attacks should leverage transferability of adversarial attacks (recall that we have seen that attacks on one DNN model transfer to others.)

What to Submit

- Colab Notebook and its pdf format.
- Please submit 1 Colab Notebook and 1 PDF file without zipping.
- For the Challenge, please submit a file containing your adversarially modified test images.