WWW.DAUPHINE.PSL.EU

11.12.2024

Assigment 3

Training Robust Neural Networks

Marwa NAIR, Sabina ASKEROVA, Emmarius DELAR

Introduction

- **Challenge:** Neural networks are highly vulnerable to adversarial perturbations, making them unreliable in critical applications.

- **Goal of the Assignment:** Explore strategies to improve a NN-classifier robustness on the CIFAR-10 dataset.

I. Baselines

Baselines

- **Standard Training**: No defense mechanism, serves as a reference for comparison.

- **Adversarial Training (AT) using** :
  - FGSM [1]
  - PGD [2]

→ Two adversarially trained baseline models.

II - Randomization matters !

A - Game Theory and Nash Equilibrium

Adversary
Φ_ϵ

Defender
$\mathcal{H}$

	Φ1(H1)	Φ2(H2)
H1	(-1 ; 1)	(-1 ; 1)
H2	(1 ; -1)	(-1 ; 1)

Mathematical formulation :

$$\inf_{h \in \mathcal{H}} \quad \sup_{\phi \in \left(\mathcal{F}_{\mathcal{X}|\epsilon_2}\right)^2} \quad \mathcal{R}_{\text{adv}}(h, \phi)$$

B - Assumption on the setup

- Adversary is completely informed (weight, data distribution, etc..)

- Mass penalty

$$\Omega_{\text{mass}}(\phi) := E_{Y\sim\nu}\left[E_{X\sim\mu_Y}\left[1\{X \neq \phi_Y(X)\}\right]\right]$$

- Norm penalty

$$\Omega_{\text{norm}}(\phi) := E_{Y\sim\nu}\left[E_{X\sim\mu_Y}\left[\|X - \phi_Y(X)\|_2\right]\right]$$

C - Boosted Adversarial Training

- Boosting:
  - Minimize the risk of a class of functions (H, our set of classifiers).
- Adversarial Training:
  - FGSM & PGD attacks are used to generate adversarial data.
- Randomization:
  - Define a mixture of models.

D - Boosted Adversarial Training Algorithm

Algorithm 2 Boosted Adversarial Training [4]

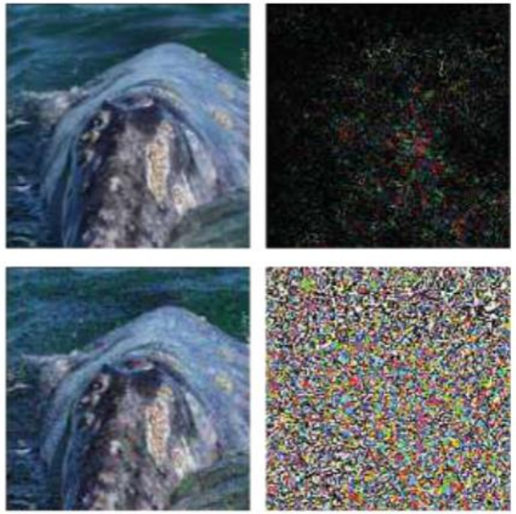
Input: D the training data set and α the weight parameter.
Create and adversarially train h_1 on D
Generate the adversarial data set $\tilde{D}$ against h_1
Create and naturally train h_2 on $\tilde{D}$
$q \leftarrow (1-\alpha, \alpha)$
$h \leftarrow (1-\alpha) * h_1 + (\alpha) * h_2$
Return h

DeepFool attack

III – DeepFool attack

A - Principle

- Finds minimal adversarial perturbation to change image classification
- Iteratively projects input across decision boundary
- Aims to use smallest possible modification to fool the classifier



Adversarial perturbations (DF and FGSM)

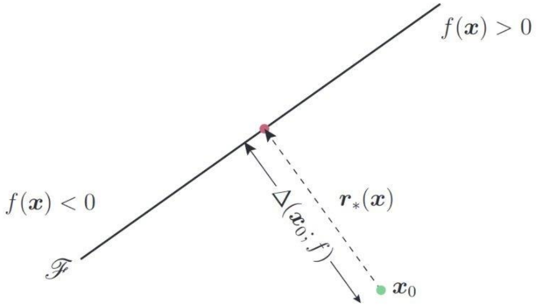
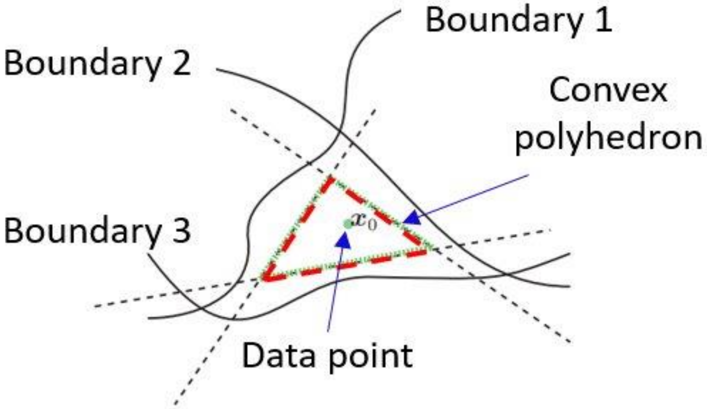


Figure 2: Adversarial examples for a linear binary classifier.



III – DeepFool attack

B - Limitations

- **Minimal perturbations might not sufficiently challenge network**
- **Computational complexity of iterative algorithm**
- **Subtle adversarial examples may not effectively expand model's decision boundaries**

DeepFool shows promise in theory, but practical implementation requires further refinement

III – DeepFool attack

C – Results

MODEL	Natural Accuracy	Accuracy for DeepFool attacks
default	**78.125**	18.55
BAT+PGD	66.503	**58.89**
BAT+FGSM	58.203	37.01
AT+DeepFool (overshoot 0.02)	75.407	37.5
AT+FGSM	72.143	35.742
AT+PGD	46.875	18.65

Results & Analysis

Results and Analysis

Model	Net Acc (%)	PGD ℓ_∞ (%)	PGD ℓ_2 (%)	Time (s)
Standard Training	76.25	6.03	25.15	153.46
AT + FGSM	63.75	15.92	17.85	444.51
AT + PGD	43.75	24.53	30.01	137.6
BAT + FGSM	63.75	50.03	57.79	247.3
BAT + PGD	65	49.83	57.05	780

References

- [1] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adver-sarial examples.
- [2] Aleksander Madry. Towards deep learning models resistant to adversarial attacks.
- [3] Fawzi A. Frossard P. Moosavi-Dezfooli, S. Deepfool: a simple and accurate method to fool deepneural networks.
- [4] Rafael Pinot, Raphael Ettedgui, Geovani Rizk, Yann Chevaleyre, and Jamal Atif. Randomization matters : how to defend against strong adversarial attacks

Thank you
for your attention!

UNIVERSITÉ PARIS DAUPHINE - PSL
Place du Maréchal de Lattre de Tassigny – 75775 Paris cedex 16