Robust Neural Networks

Artur Dandolini Pescador
Caio Jordan Azevedo
Rafael Benatti

Adversarial Attacks

FGSM Attack

- Use the signed gradient to construct the output adversarial example.
- Attacker has access to the model gradients.

$$\mathbf{X}^{\text{adv}} = \mathbf{X} + \epsilon \, \text{sgn}(\nabla_X L(\mathbf{X}, y_{\text{true}}))$$

- Results:
  Normal training accuracy: 55%
  FGSM attack accuracy: 10%

Adversarial Attacks

FGSM Attack

$$\mathbf{X}^{\text{adv}} = \mathbf{X} + \epsilon \, \text{sgn}(\nabla_X L(\mathbf{X}, y_{\text{true}}))$$

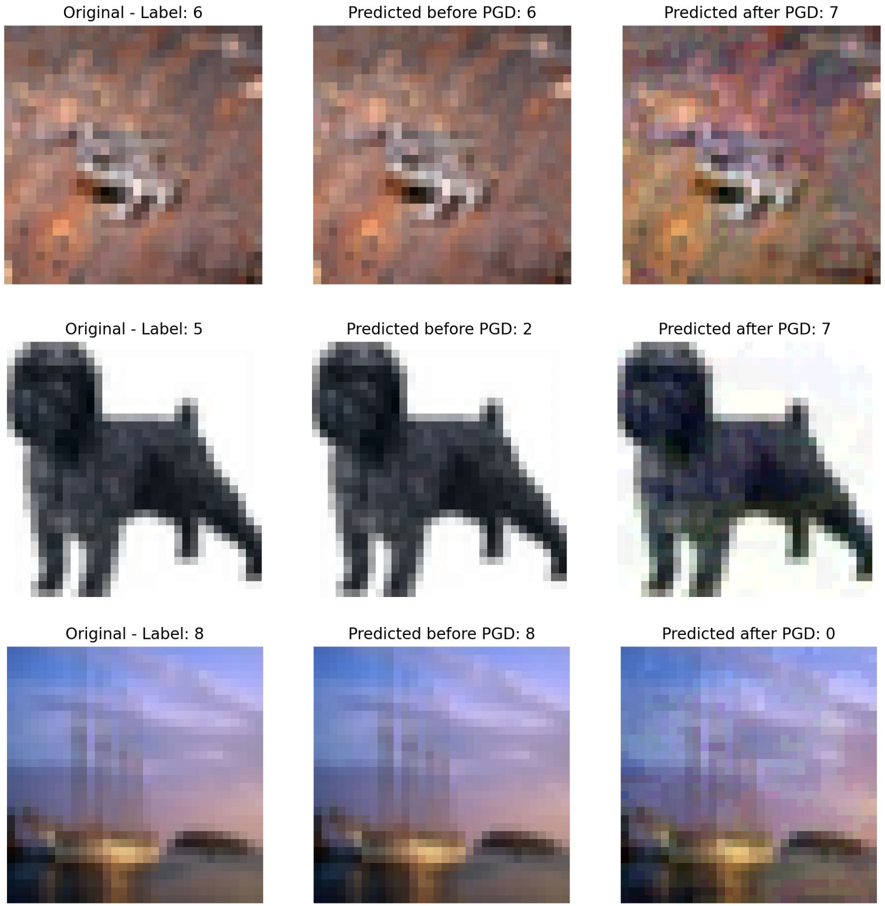
- Results:
  Normal training accuracy: 55%
  FGSM attack accuracy: 10%

Adversarial Attacks

PGD Attack

- Project Gradient Descent (PGD) attack.
- Iterative version of FGSM.
- Attacker has access to the model gradients.

- Results:
  Normal training accuracy: 55%
  PGD attack accuracy: 6.3%

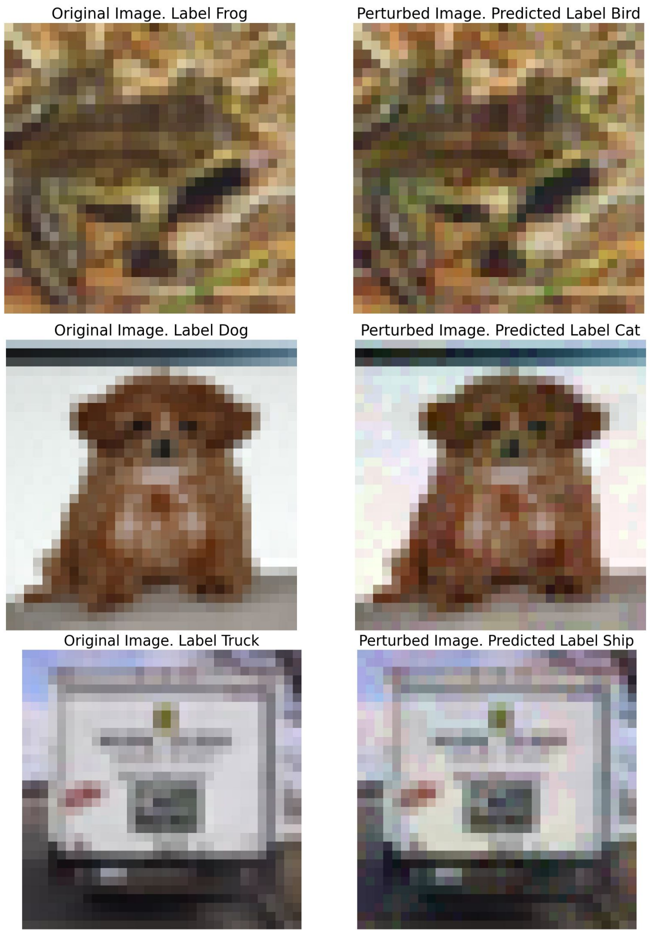


Adversarial Attacks

CW Attack

$$\text{minimize } \left\|\tfrac{1}{2}(\tanh(w) + 1) - x\right\|_2^2 + c \cdot f\left(\tfrac{1}{2}(\tanh(w) + 1)\right)$$

- Results:
  Normal training accuracy: 55%
  Accuracy for CW Adversarial Examples: 32%



Adversarial Training

- Adversarial training to produce robust models.

Adversarial Training	Training Accuracy (with attack)	Adversarial Training Accuracy
FGSM	10%	22,85%
PGD	6.3%	24,02%

Defense #1: Random Self-Ensemble (RSE)

- Defense based on **randomness** and **ensembling**
- Adds a (strong) noise layer before each convolution
- Compensates unstable performance with ensembling in inference
- Results:
  Normal training accuracy: 55% without x 52% with RSE
  FGSM attack accuracy: 10% without x 16% with RSE
  PGD attack accuracy: 6.3% without x 9.8% with RSE

Defense #2: Mixup Inference

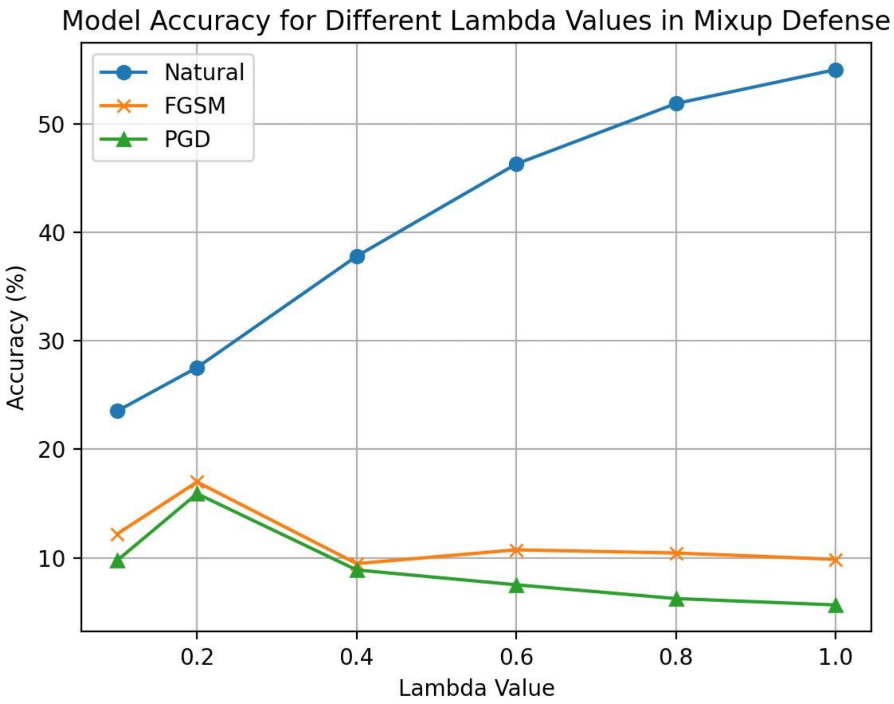
- Stochastic interpolation procedure used during inference to mitigate adversarial perturbation.
- For each input x, compute K interpolations with samples s_k,

$$\tilde{x}_k = \lambda x + (1 - \lambda) s_k$$

where λ is a fixed hyper-parameter (e.g. 0.6 - paper).

- Results:
Normal training accuracy: 55% without x 43% with Mixup Inference
FGSM attack accuracy: 10% without x 9.5% with Mixup Inference
PGD attack accuracy: 6.3% without x 7.3% with Mixup Inference

Defense #2: Mixup Inference



References

1. Goodfellow, I. J., Shlens, J., & Szegedy, C. (2014). Explaining and Harnessing Adversarial Examples. arXiv:1412.6572 (PGD)

2. Chang, T.-J., He, Y., & Li, P. (2018). Efficient Two-Step Adversarial Defense for Deep Neural Networks. arXiv:1810.03739 (FGSM)

3. Tianyu Pang, Kun Xu, Jun Zhu. Mixup Inference: Better Exploiting Mixup to Defend Adversarial Attacks. arXiv:1909.11515v2 (Mixup Inference)

4. Xuanqing Liu, Minhao Cheng, Huan Zhang, Cho-Jui Hsieh. Towards Robust Neural Networks via Random Self-ensemble. arXiv:1712.00673v2
5. Nicholas Carlini, David Wagner. Towards Evaluating the Robustness of Neural Networks. arXiv:1608.04644v2