

# Transferability Reduced Smooth (TRS) Ensemble Training for Adversarial Transferability Attacks

**Thomas Boudras, Vivien Conti, and Jules Merigot**

**Université Paris Dauphine**

December 5, 2023

# Introduction

- ▶ Trained our model adversarially on FGSM, PGD  $\ell_2$  norm and  $\ell_\infty$  norm attacks, with fine-tuning
- ▶ Employed Transferability Reduced Smooth (TRS) ensemble training to combine these trainings to build a more robust model

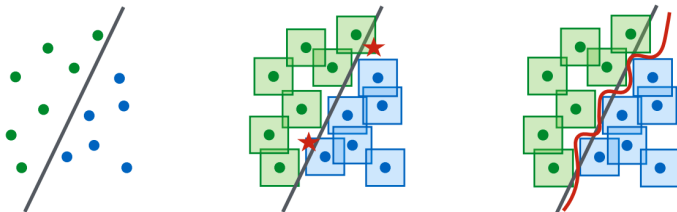


Figure 1: Illustration of standard vs. adversarial decision boundaries. [1]

# Hyper-parameters for Adversarial Training

- ◀ Best adversarially trained PGD  $\ell_\infty$  model:
  - Dynamic  $\epsilon$  from 0.03 to 0.08
  - Dynamic  $\alpha$  from 0.01 to 0.03
  - Learning rate scheduler for 50% decrease every epoch starting at  $\text{lr} = 0.01$
- ◀ Testing results:

Defensive Method	Natural Acc	PGD $\ell_\infty$ Acc	PGD $\ell_2$ Acc
PGD $\ell_\infty$ Adversarial Training	41.25	37.33	41.71
TRS Ensemble Training	-	-	-

Table 1: Results of our defensive methods.

# Ensemble Robustness via Transferability Minimization

- ◀ **Goal:** Enforce the smoothness of models to improve robustness **AND** reduce the loss gradient similarity between models to introduce global model orthogonality.

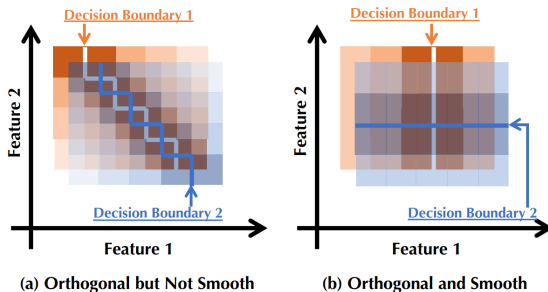


Figure 2: An illustration of the relationship between adversarial transferability, gradient orthogonality, and model smoothness [2]

# TRS Ensemble Training

Reduce adversarial transferability among base models  $\mathcal{F}$  and  $\mathcal{G}$  by enforcing model smoothness and low loss gradient similarity **at the same time**.

- ◀ Use regularization for model smoothness and thus robustness :

$$\mathcal{L}_{\text{smooth}}(\mathcal{F}, \mathcal{G}, x, \delta) = \max_{\|\hat{x} - x\|_{\infty} \leq \delta} (\|\nabla_{\hat{x}} \ell_{\mathcal{F}}\|_2 + \|\nabla_{\hat{x}} \ell_{\mathcal{G}}\|_2)$$

- ◀ Decrease the model loss gradient similarity by minimizing cosine similarity between loss gradient vectors  $\nabla_x \ell_{\mathcal{F}}$  and  $\nabla_x \ell_{\mathcal{G}}$  :

$$\mathcal{L}_{\text{sim}} = \left| \frac{(\nabla_x \ell_{\mathcal{F}})^{\top} (\nabla_x \ell_{\mathcal{G}})}{\|\nabla_x \ell_{\mathcal{F}}\|_2 \cdot \|\nabla_x \ell_{\mathcal{G}}\|_2} \right|$$

The optimal case implies orthogonal loss gradient vectors, which ensures models learn different patterns from the data.

# TRS Regularizer & Training

- ◀ **TRS Regularizer** : Increase model smoothness and diversity in learning.

$$\mathcal{L}_{\text{TRS}}(\mathcal{F}, \mathcal{G}, x, \delta) = \lambda_a \cdot \mathcal{L}_{\text{sim}} + \lambda_b \cdot \mathcal{L}_{\text{smooth}}$$

- ◀ **TRS Trainer** : Combination of the Ensemble Cross-Entropy (ECE) loss and the TRS regularizer.

$$\mathcal{L}_{\text{train}} = \frac{1}{N} \sum_{i=1}^N \mathcal{L}_{\text{CE}}(\mathcal{F}_i(x), y) + \frac{2}{N(N-1)} \sum_{i=1}^N \sum_{j=i+1}^N \mathcal{L}_{\text{TRS}}(\mathcal{F}_i, \mathcal{F}_j, x, \delta)$$

The training process focuses on making each model accurate (via ECE loss) and also on ensuring that the ensemble as a whole is robust and diverse in its learning approach (via TRS regularizer).

# Conclusion & Next Steps

TRS ensures that each model is both effective on its own and contributes uniquely to the overall performance of the ensemble.

TRS ensemble training is a lightweight yet effective way to reduce adversarial transferability, and leads to more robust models.

Next steps:

- ◀ Finalize ensemble code with our adversarially trained models
- ◀ Doing adversarial training with several attack types
- ◀ *Maybe:* Test with different model architectures to improve accuracy

# Publication and References

- 1 A. Madry, A. Makelov, L. Schmidt, D. Tsipras, A. Vladu. (2019). Towards Deep Learning Models Resistant to Adversarial Attacks. MIT. arXiv:1706.06083 [stat.ML]
- 2 Z. Yang, L. Li, X. Xu, S. Zuo, Q. Chen, B. Rubinstein, P. Zhou, C. Zhang, B. Li. (2021). TRS: Transferability Reduced Ensemble via Promoting Gradient Diversity and Model Smoothness. Proceedings of the 35th Conference on Neural Information Processing Systems (NeurIPS 2021). arXiv:2104.00671 [cs.LG]