Adversarial Feature Desensitization

Pouya Bashivan[1,2], Reza Bayat[2], Adam Ibrahim[2], Kartik Ahuja[2], Mojtaba Faramarzi[2], Turaj Laleh[2], Blake Richards[1,2], Irina Rish[2]

[1] McGill University
[2] MILA, Université de Montréal

NeurIPS 2021

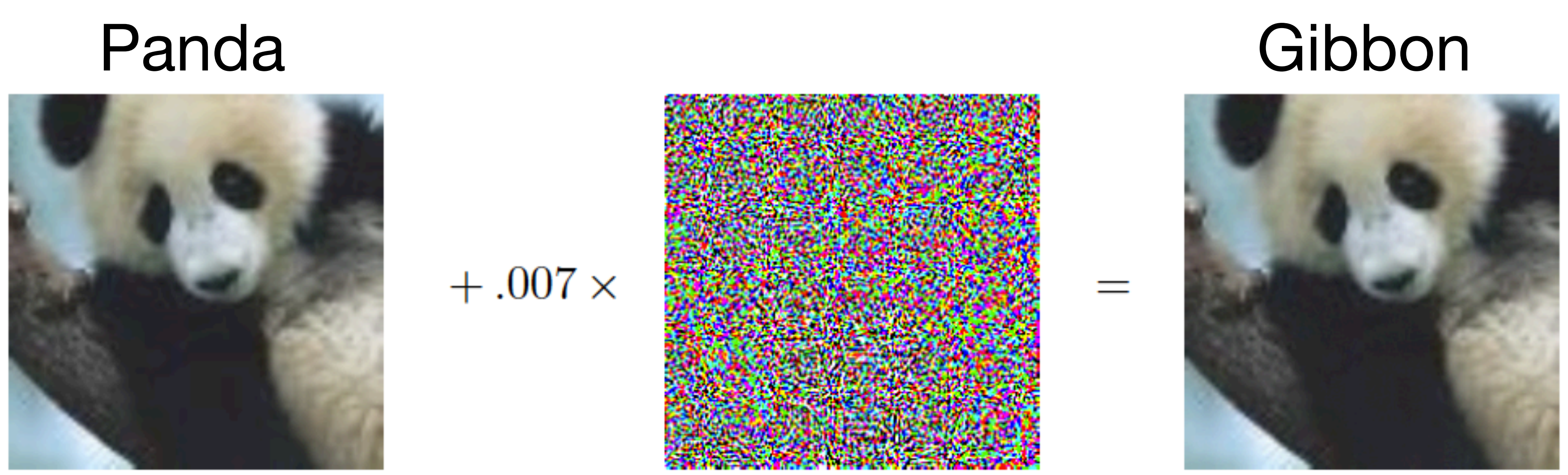
Université de Montréal

Mila

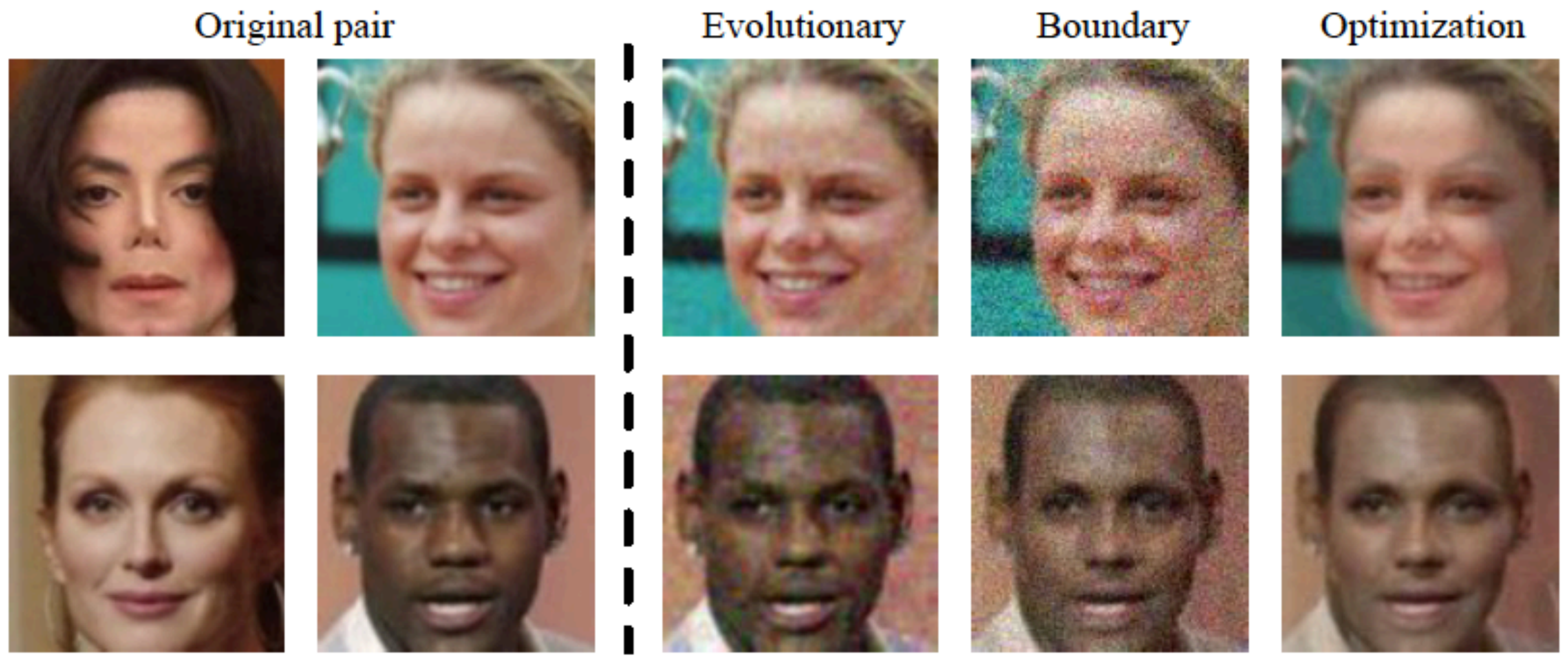
McGill

Motivation

1. **Common assumption**: train and test distributions come from the same distributions
2. *Adversarial attacks* intentionally violate this assumption.
3. This severely impacts the safety of ML-based systems in real world applications such as face recognition and autonomous driving.



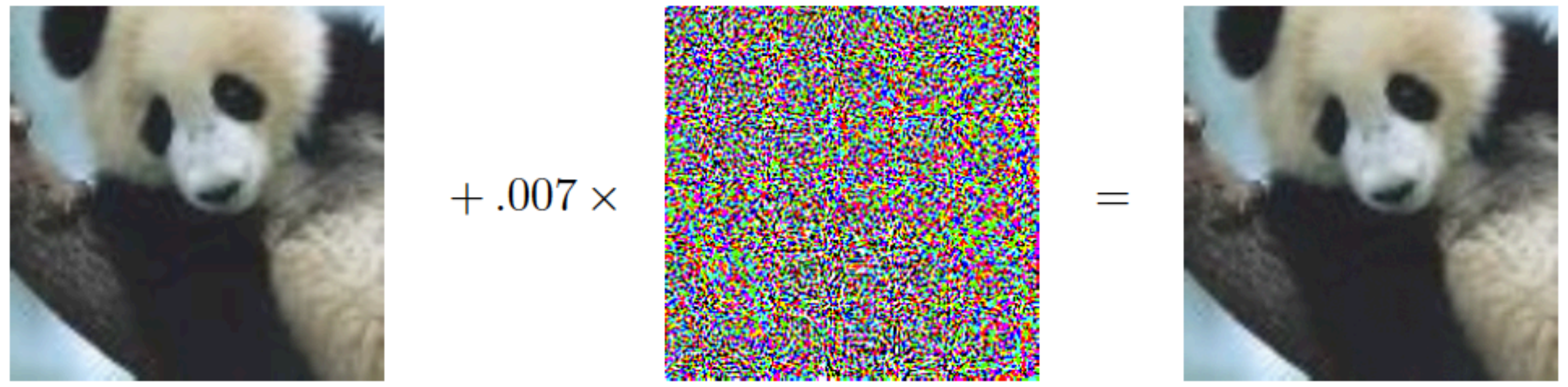
Goodfellow et al. ICLR 2015



Dong et al. CVPR 2019

Eykholt et al. CVPR 2018

What is an adversarial attack?



Goodfellow et al. ICLR 2015

- Assume
  - Feature learning function $F_\theta : \mathscr{X} \rightarrow \mathscr{Z}$, $\mathscr{X} \subseteq \mathbb{R}^n, \mathscr{Z} \subseteq \mathbb{R}^m$
  - Task classifier $C_\phi : \mathscr{Z} \rightarrow \mathscr{Y}, \mathscr{Y} = \{1,\ldots,K\}$
  - $\hat{y} = C_\phi(F_{theta}(x))$ is the predicted class for sample input x
- For $(x, y) \in \mathscr{X} \times \mathscr{Y}$, $\pi(x, \epsilon)$ is and attack function that generates perturbed samples $x' \in \mathscr{B}(x, \epsilon)$ within the ϵ-neighborhood of x by maximization the following objective:

$$\max_{t \in \mathscr{B}(x,\epsilon)} \mathscr{L}(C_\phi(F_\theta(t)), y)$$

Prior work

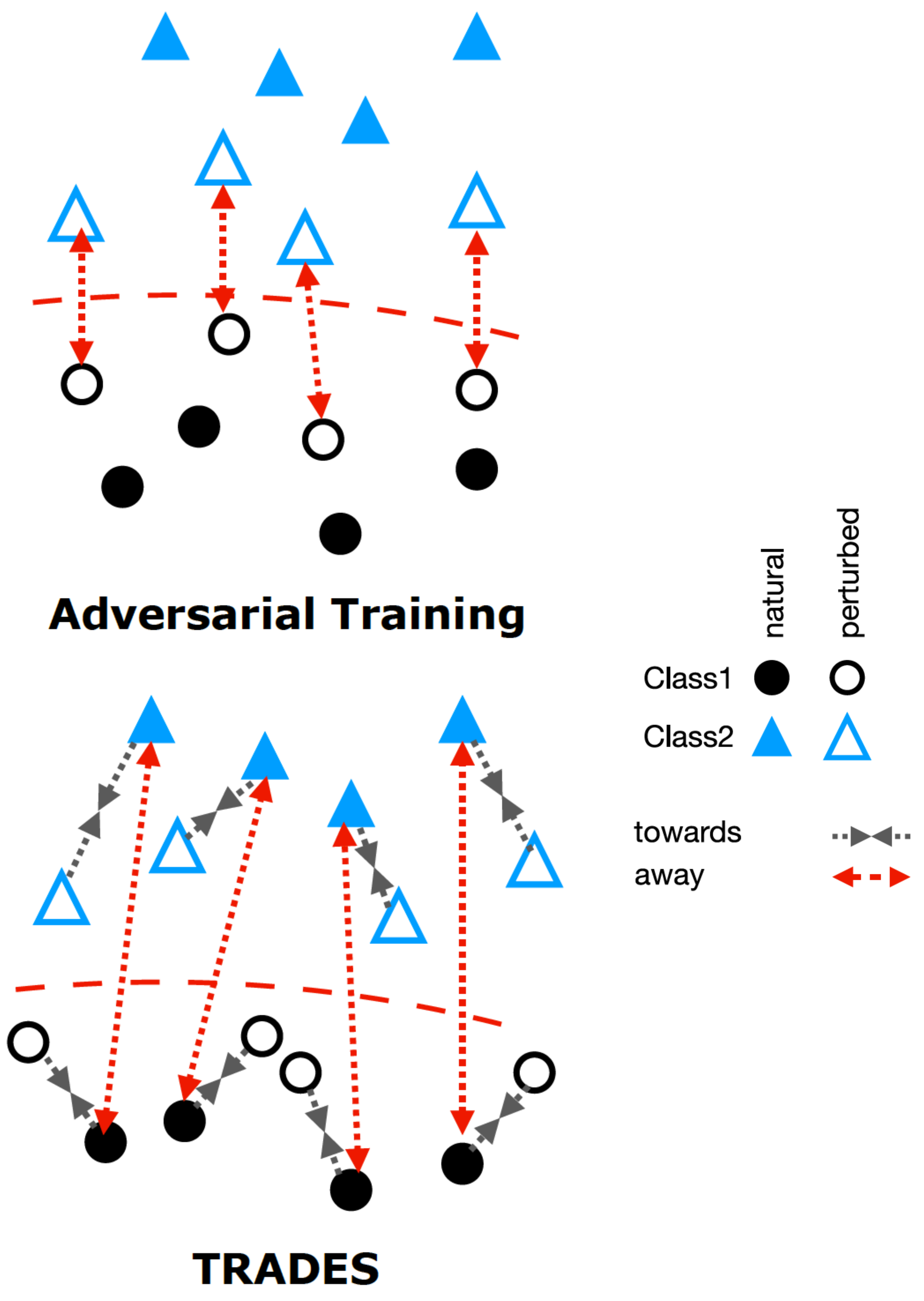
- ***Adversarial training (Madry et al. 2018)***
  train the model on examples that maximize the loss.

$$\min_{\theta} \rho(\theta), \quad \text{where} \quad \rho(\theta) = \mathbb{E}_{(x,y)\sim\mathcal{D}} \left[\max_{\delta \in \mathcal{S}} L(\theta, x+\delta, y) \right] .$$

- ***TRADES (Zhang et al. 2019)***
  pushes the decision boundary away from data.

$$\min_{f} \mathbb{E}\Big\{ \underbrace{\phi(f(\boldsymbol{X})Y)}_{\text{for accuracy}} + \underbrace{\max_{\boldsymbol{X}' \in \mathbb{B}(\boldsymbol{X},\epsilon)} \phi(f(\boldsymbol{X})f(\boldsymbol{X}')/\lambda)}_{\text{regularization for robustness}} \Big\}.$$

Robust performance remains susceptible to even slightly larger adversarial attacks or to other forms of attacks.



Method

- Our <u>proposal</u> is to ***view the adversarial robustness problem through the lens of domain adaptation*** *(Ben-David et al. 2007, 2010)***.**
- Domain adaptation theory answers "*Under what conditions can we adapt a classifier trained on the source domain for use in the target domain?" (Ben-David et al. 2007).*
- We consider the distributions of *Natural* and *Adversarial* examples as the source and target domains. Although here the ***target domain continuously evolves!***
- Our <u>goal</u> is to learn representations $z = F_\theta(x)$ that are invariant to the choice of domain (i.e. natural or adversarial).

Domain adaptation

- Assume
  - Feature learning function $F_\theta : \mathscr{X} \rightarrow \mathscr{Z}$, $\mathscr{X} \subseteq \mathbb{R}^n, \mathscr{Z} \subseteq \mathbb{R}^m$
  - Task classifier $C_\phi : \mathscr{Z} \rightarrow \mathscr{Y}, \mathscr{Y} = \{1,\ldots,K\}$
  - $\hat{y} = C_\phi(F_{theta}(x))$ is the predicted class for sample input x
  - Distributions of *Natural* and *Adversarial* examples are input domains $\mathscr{D}_\mathscr{X}$ and $\mathscr{D}'_\mathscr{X}$, their induced feature distributions are $\mathscr{D}_\mathscr{Z}$ and $\mathscr{D}'_\mathscr{Z}$.
  - $\epsilon_\mathscr{Z}$ and $\epsilon'_\mathscr{Z}$ are classification errors over $\mathscr{D}_\mathscr{Z}$ and $\mathscr{D}'_\mathscr{Z}$.

$$\epsilon'_\mathscr{Z}(h) \leq \epsilon_\mathscr{Z}(h) + \frac{1}{2} d_{\mathscr{H}\Delta\mathscr{H}}(\mathscr{D}_\mathscr{Z}, \mathscr{D}'_\mathscr{Z}) + c$$

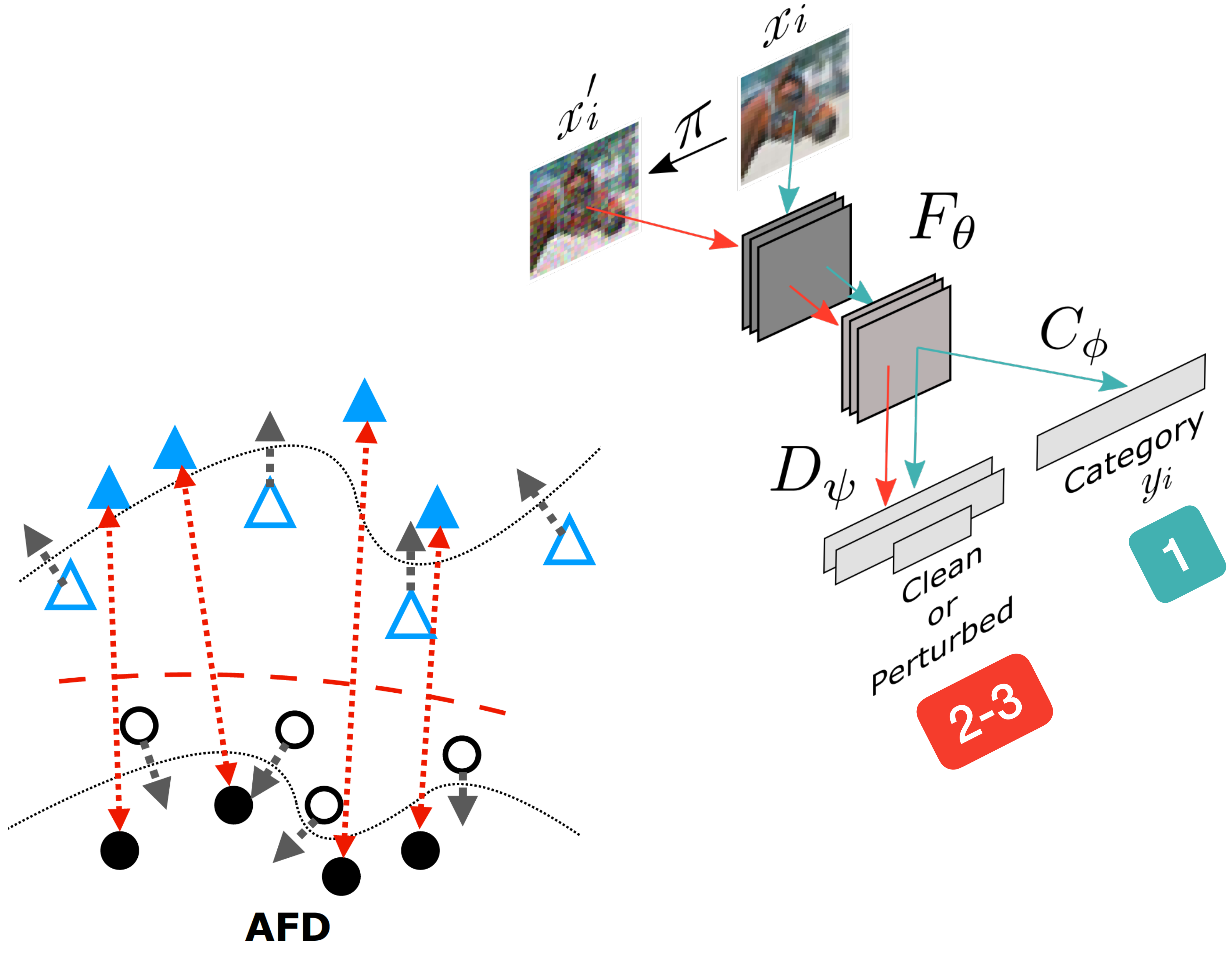
Ben-David et al. 2007, 2010

Method - Adversarial Feature Desensitization

$$\epsilon'_{\mathcal{Z}}(h) \leq \underbrace{\epsilon_{\mathcal{Z}}(h)}_{1} + \underbrace{\frac{1}{2}d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}_{\mathcal{Z}}, \mathcal{D}'_{\mathcal{Z}})}_{2\text{-}3} + c$$

- We minimize the adversarial error by
  1. Update parameters θ and ϕ to minimize the natural classification loss.
  2. Update parameters ψ to minimize the domain classification loss.
  3. Update parameters θ to maximize the domain classification loss.

This procedure implicitly "desensitizes" the learned features to adversarial perturbations.



Similar to Ganin et al. 2015

Results - robust classification on typical attacks

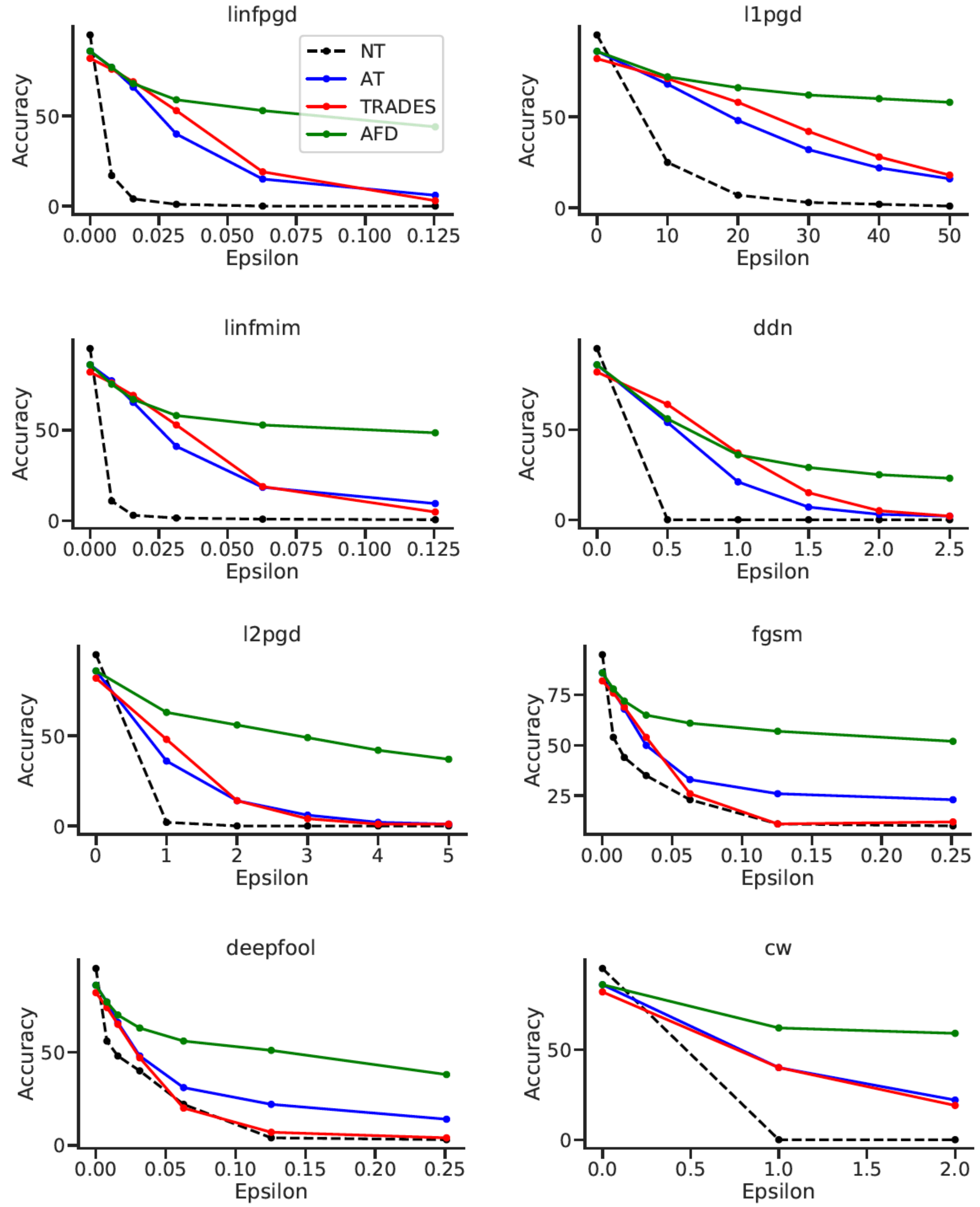
MNIST: $\epsilon = 0.3$ **CIFAR**: $\epsilon = 0.3$ **Tiny-Imagenet**: $\epsilon = 0.3$

Method	Dataset	Network	Clean	PGD_∞ (WB)	$\text{C\&W}_2$ (WB)	AA_∞ (WB)	PGD_∞ (BB)	$\text{C\&W}_2$ (BB)
NT†	MNIST	RN18	98.84	0.	62.43	0.0	50.82	96.48
AT[34]†		RN18	**99.35**	95.66	92.78	89.99	**98.92**	**98.95**
TRADES[57]†		RN18	99.14	94.81	90.08	88.66	98.5	98.57
AFD-DCGAN		RN18	99.24	95.72	93.78	88.79	98.65	98.49
AFD-WGAN		RN18	99.14	**97.68**	**97.68**	**90.12**	98.59	98.71
AT[34]†	CIFAR10	RN18	85.92	40.07	40.27	36.14	85.14	85.84
TRADES[57]†		RN18	81.94	53.3	40.24	**43.48**	80.82	81.74
AFD-DCGAN		RN18	**86.82**	44.35	50.93	34.46	**85.73**	**86.68**
AFD-WGAN		RN18	85.95	**59.38**	**62.43**	37.33	84.74	85.79
NT†	CIFAR100	RN18	76.76	0.01	0.52	0.02	-	-
AT[34]†		RN18	56.49	18.54	17.71	18.30	56.07	56.42
TRADES[57]†		RN18	60.32	**25.11**	20.52	**21.10**	59.62	60.29
AFD-DCGAN		RN18	**60.95**	18.06	21.47	16.31	**60.31**	**60.86**
AFD-WGAN		RN18	58.87	22.35	**25.33**	18.00	58.15	58.75
NT†	Tiny-IN	RN18	58.30	0.3	0.0	0.0	-	-
AT[34]†		RN18	43.80	12.62	4.90	9.48	41.87	42.86
TRADES[57]†		RN18	37.70	**13.26**	4.11	**12.57**	36.26	36.72
AFD-WGAN		RN18	**47.70**	11.49	**5.90**	9.45	**43.5**	**44.69**

AFD outperforms other baselines on most white-box and black-box attacks on various datasets.

Results - robust classification against unseen and stronger attacks

Dataset	Model	PGD$_{L_\infty}$	PGD$_{L2}$	PGD$_{L1}$	FGSM	MIM	DDN	DeepFool	C&W	AA
MNIST	NT	0.16	0.06	0.07	0.3	0.19	0.09	0.21	0.57	0.28
	AT	0.74	0.29	0.19	0.83	0.95	0.49	0.55	0.87	0.89
	TRADES	0.71	0.26	0.15	0.79	0.88	0.42	0.47	0.86	0.88
	AFD-DCGAN	0.77	0.33	0.3	0.78	0.91	0.51	0.49	0.9	0.88
	AFD-WGAN	**0.92**	**0.54**	**0.55**	**0.9**	**0.98**	**0.68**	**0.63**	**0.94**	**0.90**
CIFAR10	NT	0.05	0.1	0.17	0.19	0.05	0.1	0.16	0.1	0.12
	AT	0.28	0.2	0.44	0.33	0.31	0.26	0.29	0.31	0.22
	TRADES	0.32	0.22	0.5	0.24	0.32	0.33	0.18	0.28	**0.25**
	AFD-DCGAN	0.34	**0.54**	0.43	0.4	0.31	**0.4**	0.43	0.47	0.22
	AFD-WGAN	**0.56**	**0.54**	**0.66**	**0.59**	**0.56**	**0.4**	**0.52**	**0.62**	0.24
CIFAR100	NT	0.03	0.08	0.1	0.07	0.03	0.08	0.06	0.08	0.09
	AT	0.13	0.1	0.24	0.13	0.14	0.14	0.12	0.15	0.13
	TRADES	0.16	0.13	0.31	0.12	0.17	**0.18**	0.1	0.16	**0.15**
	AFD-DCGAN	0.14	0.12	0.27	0.17	0.16	0.15	0.16	0.18	0.13
	AFD-WGAN	**0.18**	**0.16**	**0.31**	**0.22**	**0.19**	0.16	**0.19**	**0.23**	0.13
Tiny-IN	NT	0.04	0.03	0.08	0.05	0.04	0.06	0.07	0.07	0.07
	AT	**0.10**	0.03	0.16	**0.15**	**0.09**	0.14	0.13	0.11	0.14
	TRADES	**0.10**	0.03	0.16	0.07	**0.09**	**0.15**	0.11	0.09	**0.16**
	AFD-WGAN	**0.10**	**0.04**	**0.19**	0.12	**0.09**	**0.15**	**0.16**	**0.12**	0.15



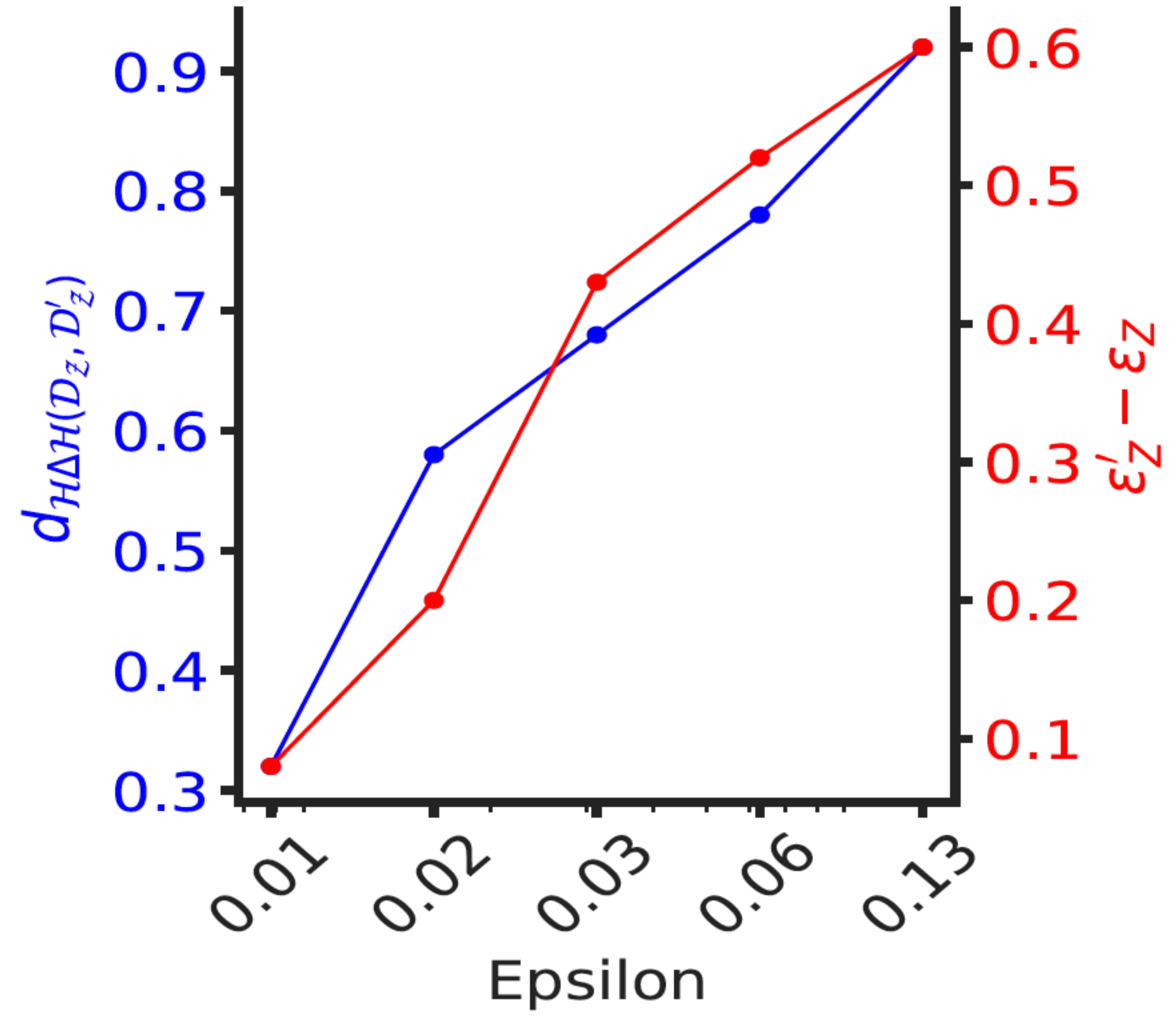
AFD's robust performance generalizes better to unseen and stronger (larger ϵ) attacks.

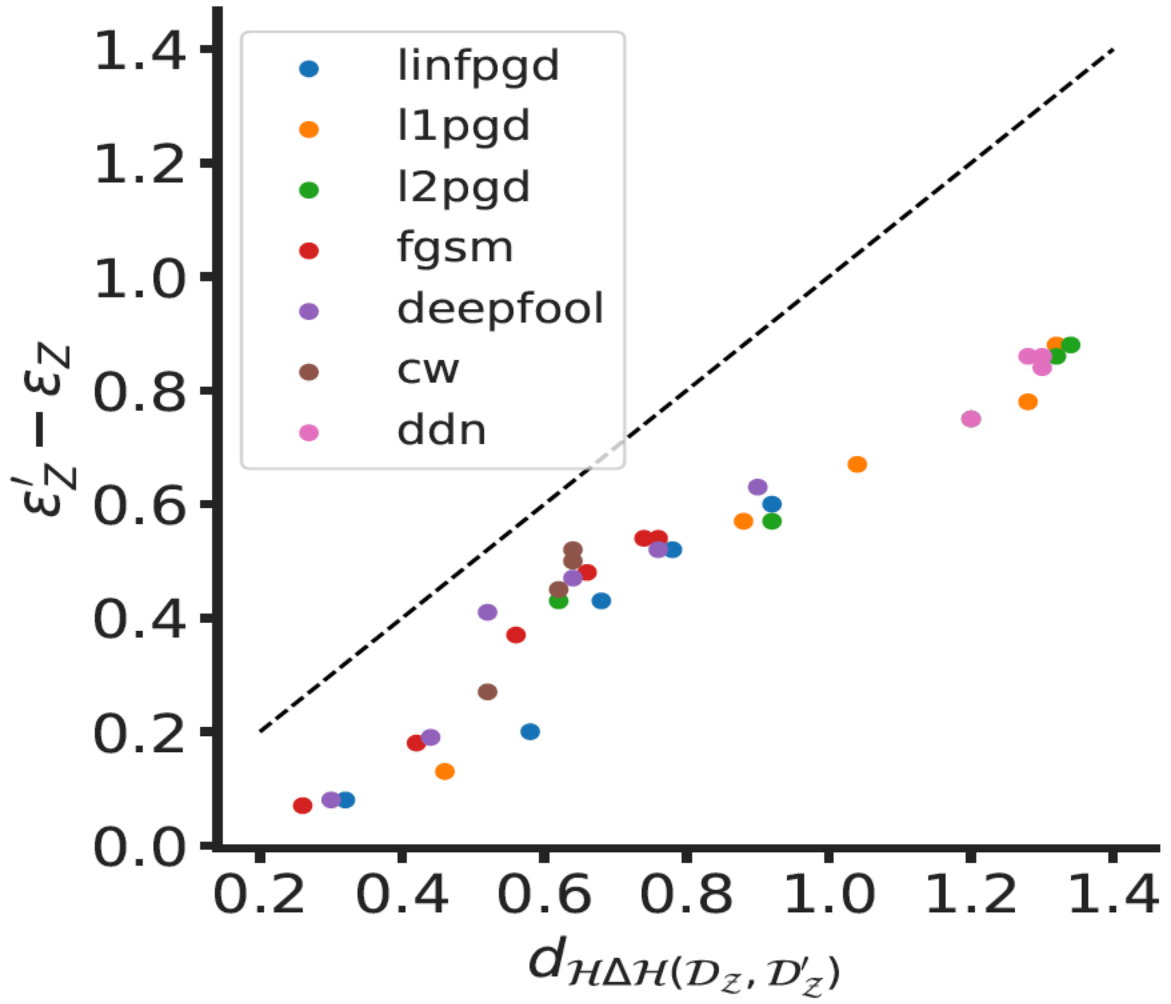
Results - $\mathscr{H}\Delta\mathscr{H}$-distance and generalization gap

- Theory of domain adaptation *predicts higher generalization gap between adversarial and natural domains with increasing* $\mathscr{H}\Delta\mathscr{H}$*-distance*
- We empirically confirmed this prediction when:
  1. increasing the attack strength (ϵ) when using a fixed attack ($PGD - L_\infty$)
  2. using various attacks of diverse magnitudes

$$\epsilon'_{\mathscr{Z}}(h) - \epsilon_{\mathscr{Z}}(h) \leq \frac{1}{2} d_{\mathscr{H}\Delta\mathscr{H}}(\mathscr{D}_{\mathscr{Z}}, \mathscr{D}'_{\mathscr{Z}}) + c$$

The domain discriminator (trained on $PGD - L_\infty$ attack with a fixed ϵ) generalizes to unseen attacks and attack-magnitudes.





Limitations

- AFD occasionally performed worse than other baselines, especially in datasets with more classes like tiny-imagenet. This could potentially be due to the difficulty of training domain classifiers in these datasets and leaves much space for future work on investigating the effect of domain classifiers on the robustness of feature learning functions.

- AFD required more backward computations compared to some other baselines such as adversarial training and as a result its training time was on average about 31% longer than adversarial training.

Thanks!

- See our full paper for more details.
- If you have any questions you can reach out to us at bashivap@mila.quebec or irina.rish@mila.quebec
- You can find our code at: https://github.com/pbashivan/afd