Robustness and Accuracy Could Be Reconcilable by (Proper) Definition

Tianyu Pang[1,2], Min Lin[1], Xiao Yang[2], Jun Zhu[2], Shuicheng Yan[1]

[1]Sea AI Lab [2]Tsinghua University

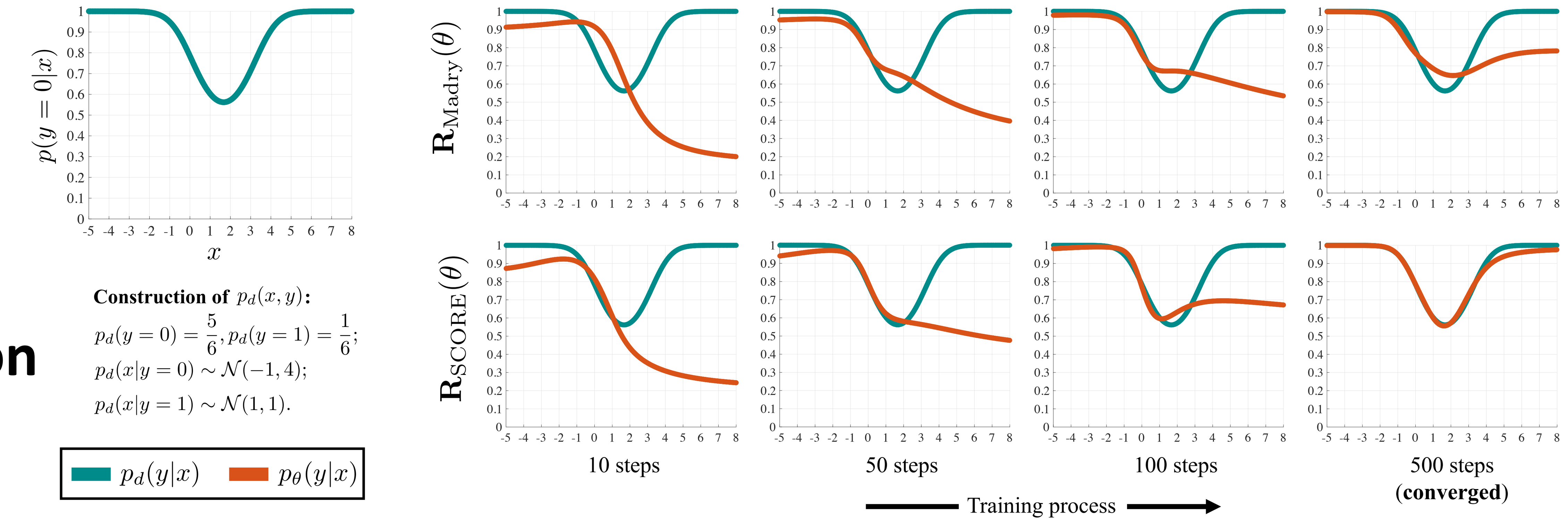
ICML | 2022

- What is an **accurate** model?

An **accurate** model has **low standard error**:

$$\mathbf{R}_{\text{Standard}} = \mathbb{E}_{p_d(x)}\left[\text{KL}\left(\underbrace{p_d(y|x)}_{\text{data distribution}} \| \underbrace{p_\theta(y|x)}_{\text{model distribution}}\right)\right]$$

Optimal solution: $p_{\theta^*}(y|x) = p_d(y|x)$



60,000 training pairs, mimics the expectation form

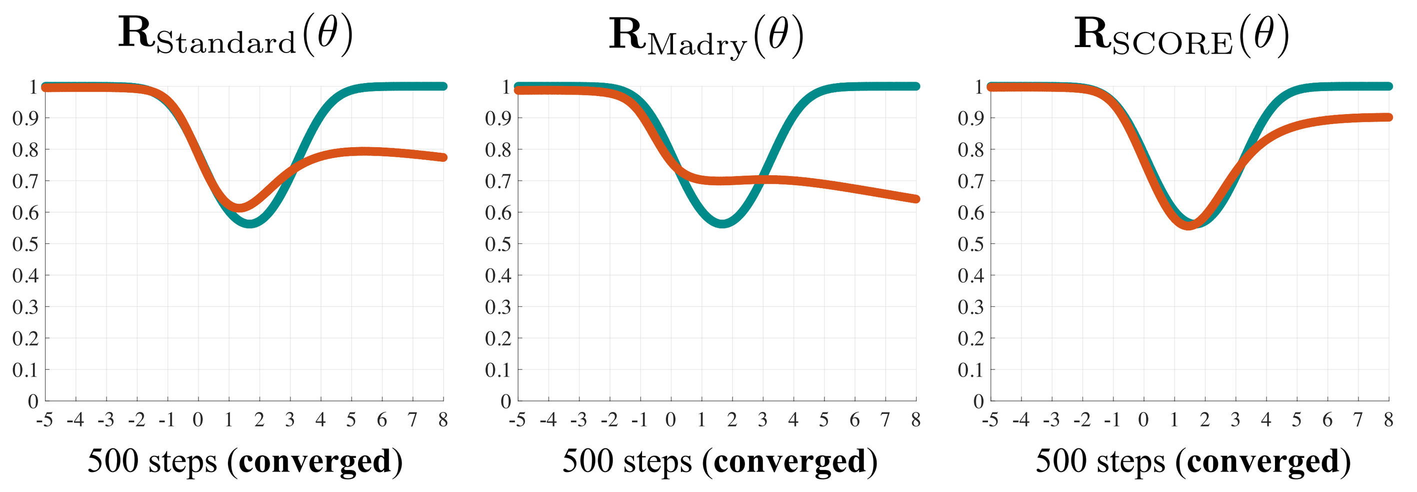
- What is a **robust** model?

A **robust** model has **low robust error**:

(Madry et al. ICLR 2018)

$$\mathbf{R}_{\text{Madry}} = \mathbb{E}_{p_d(x)}\left[\max_{x' \in B(x)} \text{KL}\left(p_d(y|x) \| p_\theta(y|x')\right)\right]$$

Optimal solution: $p_{\theta^*}(y|x) \neq p_d(y|x)$



6 training pairs, mimics the finite-sample form

$\mathbf{R}_{\text{Madry}}(\theta)$ **(local invariance)**

$\Downarrow$

$\mathbf{R}_{\text{SCORE}}(\theta)$ **(equivariance)**

- Self-COnsistent Robust Error (**SCORE**)

$$\mathbf{R}_{\text{SCORE}}(\theta) = \mathbb{E}_{p_d(x)}\left[\max_{x' \in B(x)} \text{KL}\left(p_d(y|x') \| p_\theta(y|x')\right)\right]$$

(1) **Self-consistency** : $p_{\theta^*}(y|x) = p_d(y|x)$

(2) Keep the paradigm of **robust optimization**

- How to practically optimize SCORE?

Substitute KL divergence with any **distance metric** $\mathcal{D}$

$$\mathbf{R}^{\mathcal{D}}_{\text{Madry}}(\theta) = \mathbb{E}_{p_d(x)}\left[\max_{x' \in B(x)} \mathcal{D}\left(p_d(y|x) \| p_\theta(y|x')\right)\right];$$

$$\mathbf{R}^{\mathcal{D}}_{\text{SCORE}}(\theta) = \mathbb{E}_{p_d(x)}\left[\max_{x' \in B(x)} \mathcal{D}\left(p_d(y|x') \| p_\theta(y|x')\right)\right]$$

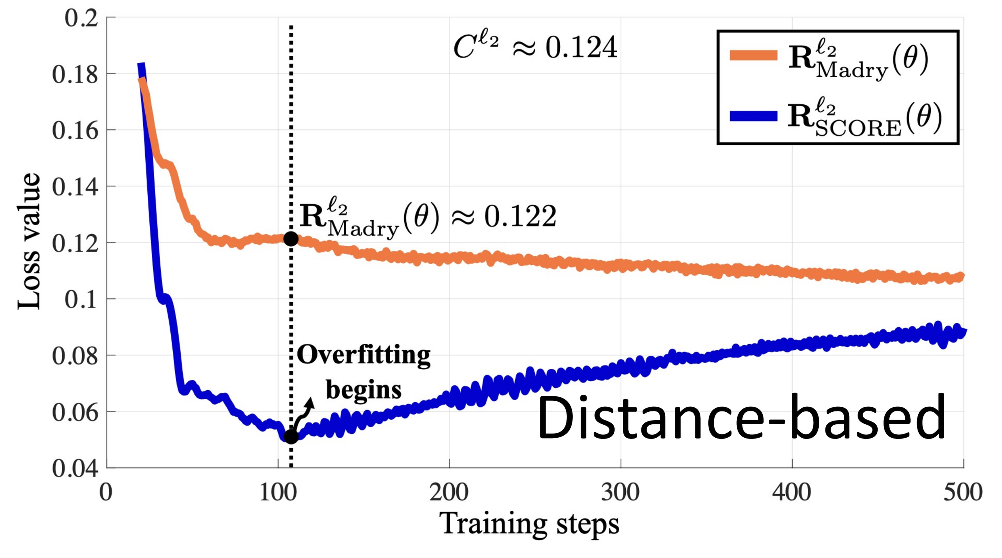
- Upper and lower bounds for SCORE

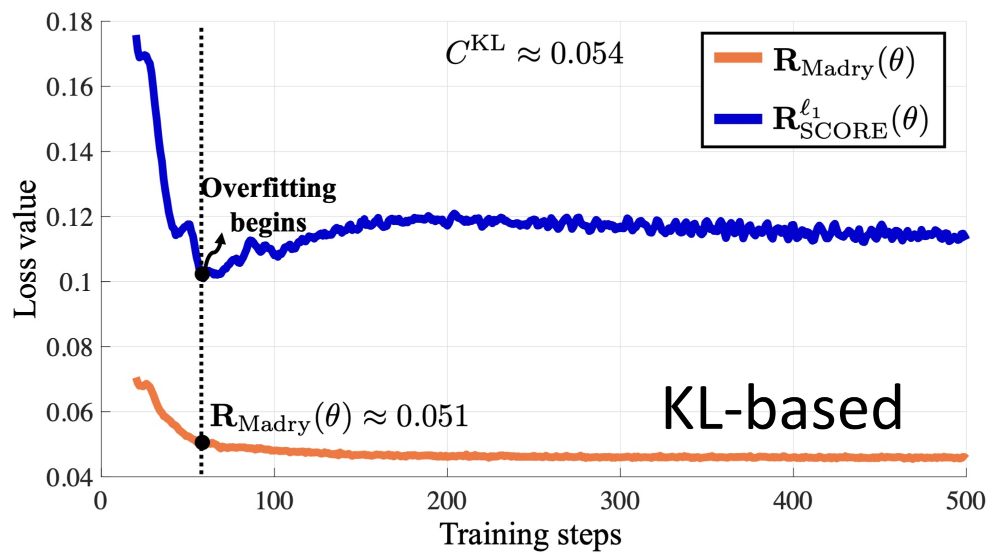
$$|\mathbf{R}^{\mathcal{D}}_{\text{Madry}}(\theta) - C^{\mathcal{D}}| \leq \mathbf{R}^{\mathcal{D}}_{\text{SCORE}}(\theta) \leq \mathbf{R}^{\mathcal{D}}_{\text{Madry}}(\theta) + C^{\mathcal{D}},$$

where $C^{\mathcal{D}} = \mathbb{E}_{p_d(x)}\left[\max_{x' \in B(x)} \mathcal{D}\left(p_d(y|x) \| p_d(y|x')\right)\right]$

Upper bound: without estimating $\nabla_x \log p_d(y|x)$

Lower bound: indicates overfitting phenomena





Dataset	Method	Architecture	DDPM	Batch	Epoch	Clean	AutoAttack
CIFAR-10 ($\ell_\infty, \epsilon = 8/255$)	Rice et al. (2020)	WRN-34-20	✗	128	200	85.34	53.42
	Zhang et al. (2020)	WRN-34-10	✗	128	120	84.52	53.51
	Pang et al. (2021)	WRN-34-20	✗	128	110	86.43	54.39
	Wu et al. (2020)	WRN-34-10	✗	128	200	85.36	56.17
	Gowal et al. (2020)	WRN-70-16	✗	512	200	85.29	57.14
	Rebuffi et al. (2021)‡	WRN-28-10	1M	1024	800	85.97	60.73
	+ **Ours** (KL $\rightarrow$ SE, $\beta = 3$)	WRN-28-10	1M	512	400	**88.61**	**61.04**
	+ **Ours** (KL $\rightarrow$ SE, $\beta = 4$)	WRN-28-10	1M	512	400	**88.10**	**61.51**
	Rebuffi et al. (2021)‡	WRN-70-16	1M	1024	800	86.94	63.58
	+ **Ours** (KL $\rightarrow$ SE, $\beta = 3$)	WRN-70-16	1M	512	400	**89.01**	**63.35**
	+ **Ours** (KL $\rightarrow$ SE, $\beta = 4$)	WRN-70-16	1M	512	400	**88.57**	**63.74**
	Gowal et al. (2021)	WRN-70-16	100M	1024	2000	88.74	66.10
CIFAR-100 ($\ell_\infty, \epsilon = 8/255$)	Wu et al. (2020)	WRN-34-10	✗	128	200	60.38	28.86
	Gowal et al. (2020)	WRN-70-16	✗	512	200	60.86	30.03
	Rebuffi et al. (2021)‡	WRN-28-10	1M	1024	800	59.18	30.81
	+ **Ours** (KL $\rightarrow$ SE, $\beta = 3$)	WRN-28-10	1M	512	400	**63.66**	**31.08**
	+ **Ours** (KL $\rightarrow$ SE, $\beta = 4$)	WRN-28-10	1M	512	400	**62.08**	**31.40**
	Rebuffi et al. (2021)‡	WRN-70-16	1M	1024	800	60.46	33.49
	+ **Ours** (KL $\rightarrow$ SE, $\beta = 3$)	WRN-70-16	1M	512	400	**65.56**	**33.05**
	+ **Ours** (KL $\rightarrow$ SE, $\beta = 4$)	WRN-70-16	1M	512	400	**63.99**	**33.65**

Find more interesting conclusions in our paper!