Adversarial Robustness through Wide Local Minima: A Simple Training Technique

Anton S. Khritankov
MIPT, HSE Univ.
akhritankov@hse.ru

Andrey S. Veprikov
MIPT

Sergey A. Smirnov
IITP RAS

ABSTRACT

Achieving adversarial robustness is a critical aspect of ensuring the security and reliability of machine learning models, particularly in applications where trustworthiness is paramount. This paper delves into the theoretical aspects and impact of width of the local minima and learning parameters on adversarial robustness in Deep Neural Networks (DNNs) for image classification tasks. Through our investigation of gradient learning methods, we identify that certain optimization parameters can enhance robustness without compromising prediction quality. Building on these findings, we introduce a novel adversarial defense technique aimed at improving the model's resilience against attacks.

PROBLEM

- Achieving **adversarial robustness** (AR) is critical in many ML and AI applications
- Adversarial training is a popular technique but requires modifications to training data and/or models

HYPOTHESIS

- Increasing width of local minima leads to better robustness
- We can achieve wider minima by proper training only
- **Wider minima training** would tolerate larger noise if converges

METHOD

- Take MNIST and FF CNN w/ FC layers for image classification
- Use gradient noise from SGD to test for width of minima
- Use FGS method to attack the model **after training** to evaluate AR
- Explore batch size and learning step size to find the boundary where both accuracy and robustness are high

RESULTS

- **Better definition** of width of a minimum in Def. 1 and Thm. 1
- Thm. 2 proves **wide minima SGD training** improves robustness
- Longer training with smaller batch and/or higher rates lead to high robustness measured as accuracy on adversarial data ADV_ACC
- Achieved ADV_ACC is comparable or **better than the SOTA**
- **No modifications** to model and/or data

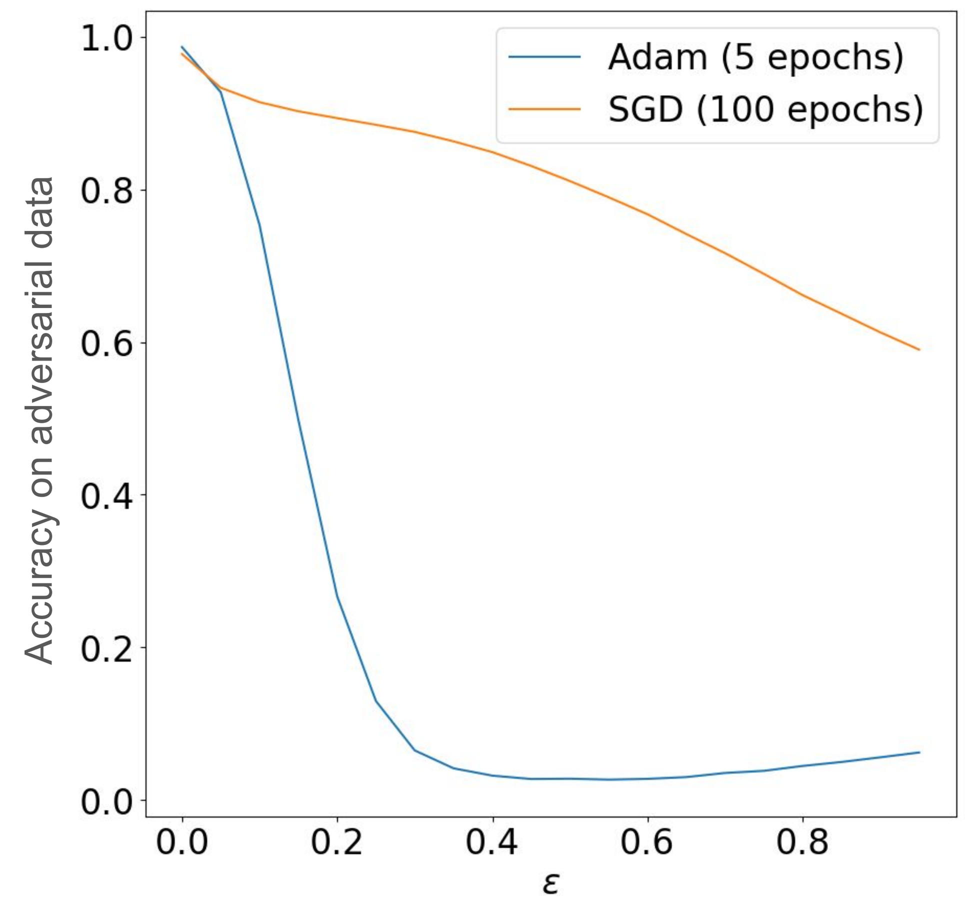
RELATED WORK

Others: complex PuVAE (Hwang et al., 2019) and BPFC (Addepalli et al., 2020) give adversarial accuracy ADV_ACC ≈ 81% on a similar task

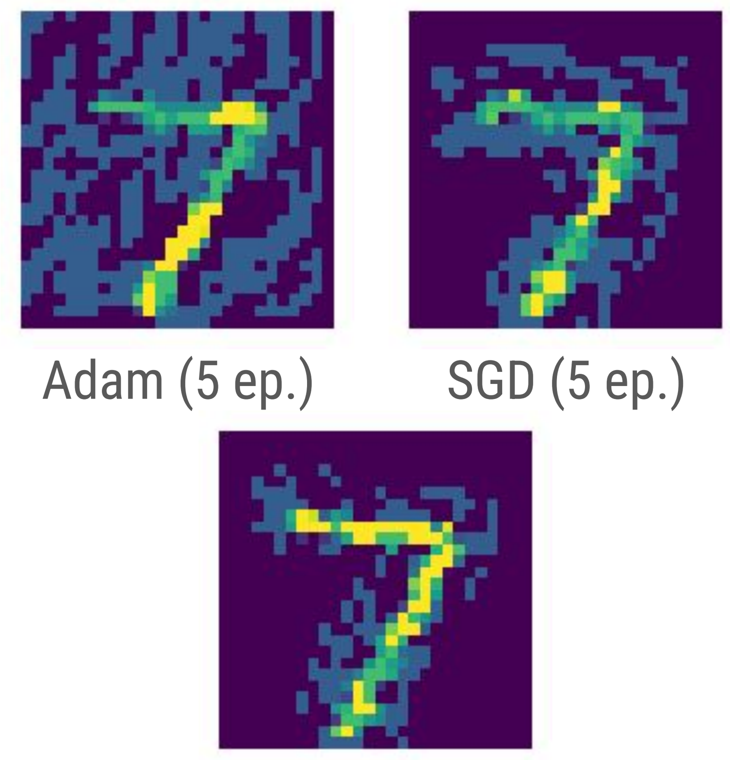
Ours: direct training with Adam and SGD optimizers with specific parameters achieve ADV_ACC 96% and 89% without accuracy loss

FUTURE RESEARCH

- Computational efficiency
- More useful noise
- Prove for other models
- Explore for other learning algorithms

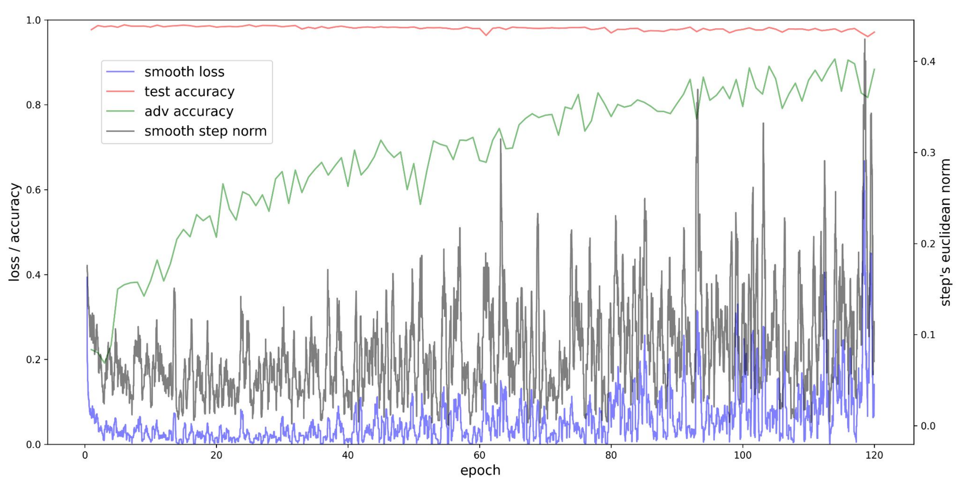


FGSM Adversarial examples (ε=0.3)

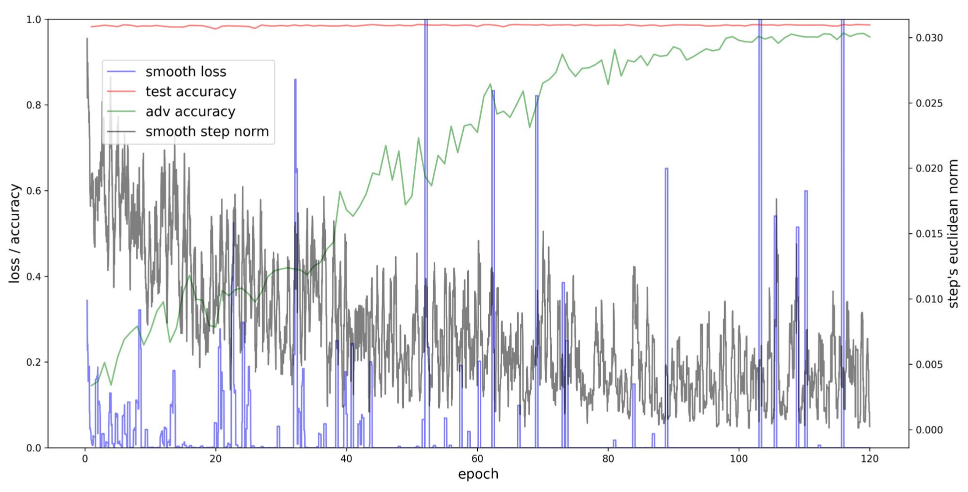


Adam (5 ep.) SGD (5 ep.)

SGD (100 ep.) — "on-manifold"



SGD training curves



Adam training curves

Definition 1 (width of a minimum) *The width $w_x(x^*, c)$ of a local minimum at x^* of $\phi(x)$ with respect to x for some c is a function*

$$w_x(x^*, c) = \frac{d(c)}{c - \phi(x^*)}, \quad \begin{array}{l} d(c) = \min_{x \in V(c)}\{\|x - x^*\|\}, \\ \text{level set } V(c) \end{array}$$

Theorem 1 *For all subsets $\mathcal{S} \subset \mathbb{R}^d$ such that $\max_{x\in\mathcal{S}}\{\|x - x^*\|\} \le d(c)$ it holds that*

1. *If ϕ is continuous on $\mathcal{S}$, then for all $x \in \mathcal{S}$ it holds that $|\phi(x)| \le |c|$.*
2. *If ϕ is convex on $\mathcal{S}$, then for all $x \in \mathcal{S}$ it holds that*

$$\phi(x) - \phi(x^*) \le \frac{1}{w(x^*, c)} \cdot \|x - x^*\|.$$

Theorem 2 (wide minima training) *Let θ^* be a minimizer to the loss $L(X, y; \theta) < \epsilon$ for a small ϵ and given X^*, y^* an SGD algorithm converges to. If the predictor $f(x;\theta)$ is a function of the inner product of its parameters $f(x;\theta) = f(x^T\theta)$, then the following holds around X^*, y^*, θ^*:*

1. *L is locally quasi-convex w.r.t. θ and w.r.t. X,*
2. *width w_θ w.r.t. to θ and the width w_X w.r.t. X have a common non-negative multiplier,*
3. *the larger the width w_θ, the larger the width w_X.*

Batch Size		Learning Rate																	
		0.003		0.006		0.01		0.03		0.06		0.1		0.3		0.6		1	
		AVG	STD	AVG	STD	AVG	STD	AVG	STD	AVG	STD	AVG	STD	AVG	STD	AVG	STD	AVG	STD
1	epoch	47	19	117	1	48	8	1	2	1	1	0	1	0	0	1	1	2	1
	acc	**98.90**	0.12	97.87	0.38	96.53	1.36	66.09	31.50	11.14	0.46	11.14	0.48	10.72	0.57	10.47	0.50	10.43	0.52
	adv_acc	55.19	2.91	**89.48**	2.01	82.27	2.52	31.63	15.20	11.14	0.46	11.14	0.48	10.72	0.57	10.47	0.50	10.43	0.52
2	epoch	35	6	41	11	115	2	16	7	0	0	1	1	0	0	0	0	1	1
	acc	**98.92**	0.09	**98.88**	0.11	**98.30**	0.21	95.53	0.82	84.31	8.39	11.14	0.46	10.92	0.59	10.72	0.57	10.47	0.50
	adv_acc	40.94	4.44	53.45	3.08	**83.80**	3.80	67.99	3.97	37.36	6.22	11.14	0.46	10.92	0.59	10.72	0.57	10.47	0.50
4	epoch	45	5	35	4	33	3	97	10	12	4	1	1	1	1	0	0	0	0
	acc	**98.89**	0.09	**98.93**	0.07	**98.93**	0.15	96.78	0.71	96.42	1.22	90.44	4.84	11.14	0.48	10.92	0.59	10.72	0.58
	adv_acc	33.58	2.03	39.44	2.78	48.60	3.89	**89.26**	1.12	71.90	4.99	43.61	8.11	11.14	0.48	10.92	0.59	10.72	0.58
8	epoch	78	15	47	10	37	3	75	38	86	9	25	8	2	1	1	1	0	0
	acc	**98.84**	0.13	**98.87**	0.11	**98.86**	0.12	**98.71**	0.27	97.29	1.02	96.69	1.01	11.14	0.46	11.14	0.48	10.92	0.59
	adv_acc	27.71	2.64	33.45	3.04	38.81	3.75	66.06	8.72	**89.38**	1.23	77.04	4.52	11.14	0.46	11.14	0.48	10.92	0.59
16	epoch	103	9	74	3	51	13	30	6	100	35	115	1	4	3	2	1	1	1
	acc	**98.72**	0.09	**98.77**	0.13	**98.83**	0.19	**98.80**	0.13	**98.60**	0.17	97.77	0.25	94.98	1.16	11.14	0.46	11.14	0.48
	adv_acc	22.57	2.84	27.36	2.51	32.99	1.50	43.05	3.15	70.23	7.32	**89.87**	1.59	54.90	4.52	11.14	0.46	11.14	0.48
32	epoch	45	61	103	11	87	19	39	7	35	6	41	8	44	14	2	1	2	1
	acc	60.26	35.63	**98.79**	0.10	**98.78**	0.12	**98.87**	0.09	**98.89**	0.15	**98.85**	0.07	97.64	0.36	11.14	0.46	11.14	0.46
	adv_acc	17.99	5.37	22.13	2.78	27.53	2.95	35.43	1.52	44.67	4.19	55.18	6.14	**80.06**	6.51	11.14	0.46	11.14	0.46
64	epoch	0	0	44	60	108	8	72	19	41	10	31	4	112	6	21	28	1	1
	acc	40.88	16.53	60.36	35.45	**98.36**	0.59	**98.83**	0.15	**98.86**	0.15	**98.80**	0.06	**98.34**	0.14	45.62	46.93	11.14	0.46
	adv_acc	17.02	7.32	18.23	5.14	21.61	3.09	29.00	1.30	33.98	1.56	42.44	2.22	81.70	2.36	40.16	39.46	11.14	0.46
128	epoch	2	1	0	0	65	59	99	12	66	24	43	4	35	5	50	48	1	1
	acc	37.47	14.20	40.95	16.54	71.57	37.34	**98.79**	0.09	**98.90**	0.06	**98.85**	0.15	**98.93**	0.09	80.31	38.57	11.35	0.00
	adv_acc	17.48	6.06	17.02	7.22	16.27	6.51	24.41	2.35	29.28	1.56	32.69	2.26	49.63	7.58	40.40	31.97	11.35	0.00

Table 2: Accuracy on adversarial test set (FGSM) for SGD optimizer. Average (AVG) and approximate standard deviation (STD) over five runs. For each mini-batch size B and learning rate γ: epoch when the best adversarial accuracy is achieved (epoch), the corresponding neutral accuracy (acc) and the adversarial accuracy (adv_acc).