Randomized Adversarial Training via Taylor Expansion

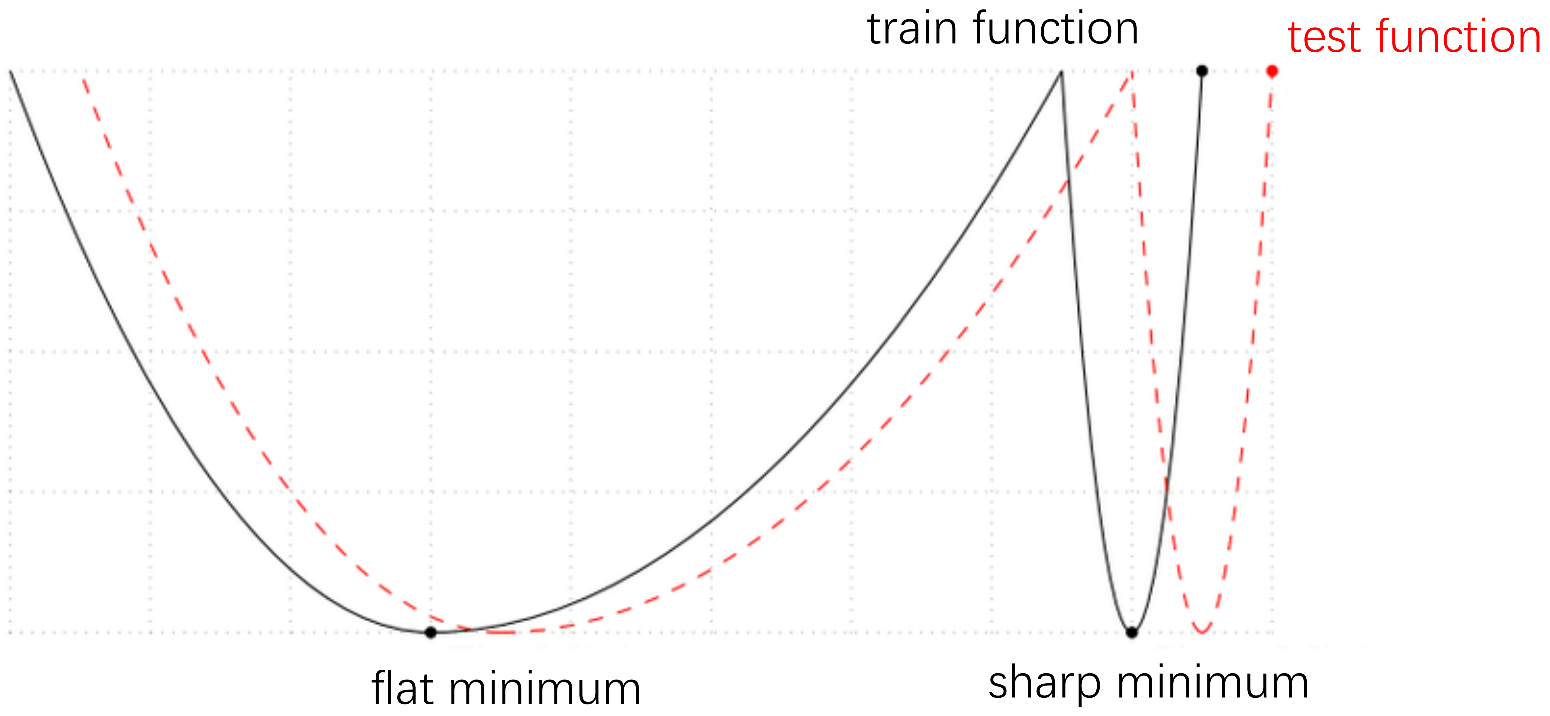
Gaojie Jin, Xinping Yi, Dengyu Wu, Ronghui Mu, Xiaowei Huang

Code: https://github.com/Alexkael/Randomized-Adversarial-Training

Paper: https://arxiv.org/abs/2303.10653

Start point: flatness

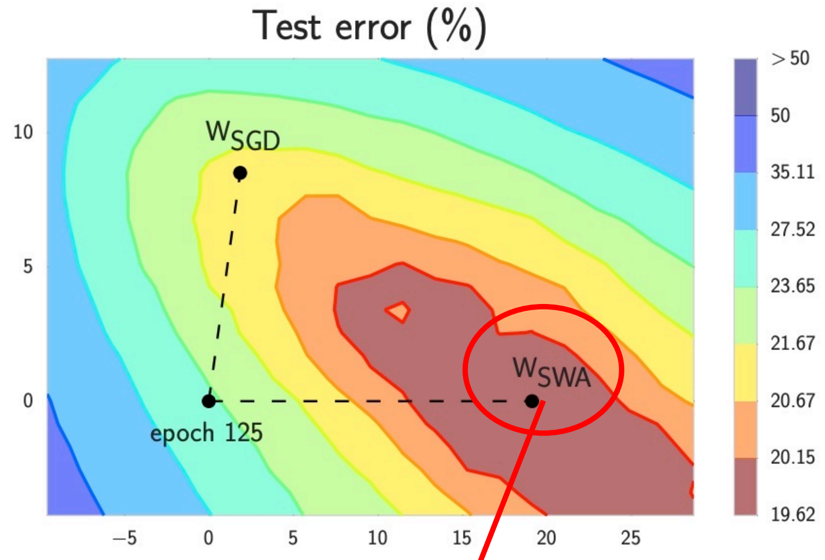
- Flat minima can help to improve generalization and robustness
- How to find a flat minima in adversarial training?

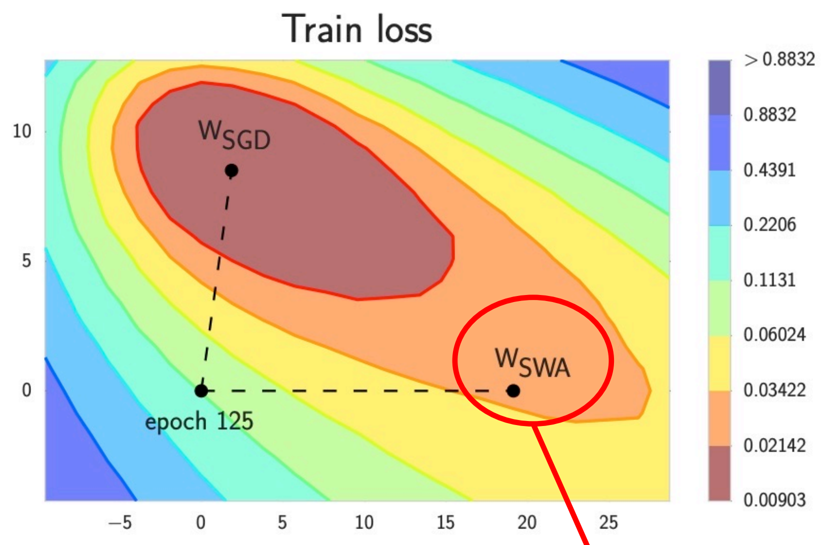


Jin, Gaojie, et al. " Randomized Adversarial Training via Taylor Expansion" CVPR 2023.

Start point: SWA

- **SWA：** save check points during training, utilize the average of checkpoints to update the weights





May be optimal in test set

May be not optimal in train set

- **Other than SWA, can we use other methods to smooth weights during training?**

Izmailov, Pavel, et al. "Averaging weights leads to wider optima and better generalization." UAI 2018.
Jin, Gaojie, et al. " Randomized Adversarial Training via Taylor Expansion" CVPR 2023.

Randomized Adversarial Training via Taylor Expansion

- **Adversarial training**
- **Average checkpoints**
- **Smooth weights**

Apply small noise to weights during adversarial training, then train on these noisy smooth weights

method

- Taylor expand noisy weights, train on Taylor terms efficiency

▶ Optimization: $\mathbf{w} \leftarrow \mathbf{w} - \eta_l \frac{1}{n} \sum_{i=1}^{n} \nabla_{\mathbf{w}} \Big[\mathcal{L}(f_{\mathbf{w}}(\mathbf{s}_i), y_i)$

(zeroth term) $+\mathcal{L}(g_{\mathbf{s}_i}(\mathbf{w}), g_{\mathbf{s}'_i}(\mathbf{w}))/\lambda$ 0-th Taylor term

(first term) $+\eta \mathbb{E}_{\mathbf{u}} \big(\mathcal{L}(g'_{\mathbf{s}_i}(\mathbf{w})^T \mathbf{u}, g'_{\mathbf{s}'_i}(\mathbf{w})^T \mathbf{u}) \big)$ 1-st Taylor term

(second term) $+\frac{\eta}{2} \mathbb{E}_{\mathbf{u}} \big(\mathcal{L}(\mathbf{u}^T g''_{\mathbf{s}_i}(\mathbf{w}) \mathbf{u}, \mathbf{u}^T g''_{\mathbf{s}'_i}(\mathbf{w}) \mathbf{u}) \big) \Big]$ 2-nd Taylor term

approximation: close first orders

approximation: close second orders

Empirical results

Table 2. First and Second Derivative terms optimization on CIFAR-10/CIFAR-100 with ℓ_∞ threat model for WideResNet, compared with current state-of-the-art. Classification accuracy (%) on clean images and under PGD-20 attack, CW-20 attack ($\epsilon = 0.031$) and Auto Attack ($\epsilon = 8/255$). The results of our methods are in **bold**. Note that * is under PGD-40 attack and ** is under PGD-10 attack.

Dataset	Method	Architecture	Clean	PGD-20	CW-20	AA
	Lee et al. (2020) [42]	WRN-34-10	92.56	59.75	54.53	39.70
	Wang et al. (2020) [74]	WRN-34-10	83.51	58.31	54.33	51.10
	Rice et al. (2020) [62]	WRN-34-20	85.34	-	-	53.42
	Zhang et al. (2020) [83]	WRN-34-10	84.52	-	-	53.51
	Pang et al. (2021) [56]	WRN-34-20	86.43	57.91**	-	54.39
CIFAR-10 ℓ_∞	Jin et al. (2022) [36]	WRN-34-20	86.01	61.12	57.93	55.90
	Gowal et al. (2020) [24]	WRN-70-16	85.29	58.22*	-	57.20
TRADES	Zhang et al. (2019) [82] (0_{th})	WRN-34-10	84.65	56.68	54.49	53.0
	+ Ours (1_{st})	WRN-34-10	**85.51**	**58.34**	**56.06**	**54.0**
	+ Ours (1_{st}+2_{nd})	WRN-34-10	**85.98**	**58.47**	**56.13**	**54.2**
AWP-TRADES	Wu et al. (2020) [76] (0_{th})	WRN-34-10	85.17	59.64	57.33	56.2
	+ Ours (1_{st})	WRN-34-10	**86.10**	**61.47**	**58.09**	**57.1**
	+ Ours (1_{st}+2_{nd})	WRN-34-10	**86.12**	**61.45**	**58.22**	**57.4**
	Cui et al. (2021) [11]	WRN-34-10	60.43	35.50	31.50	29.34
	Gowal et al. (2020) [24]	WRN-70-16	60.86	31.47*	-	30.03
	Zhang et al. (2019) [82] (0_{th})	WRN-34-10	60.22	32.11	28.93	26.9
CIFAR-100 ℓ_∞	**+ Ours (1_{st})**	WRN-34-10	**63.01**	**33.26**	**29.44**	**28.1**
	+ Ours (1_{st}+2_{nd})	WRN-34-10	**62.93**	**33.36**	**29.61**	**27.9**
	Wu et al. (2020) [76] (0_{th})	WRN-34-10	60.38	34.09	30.78	28.6
	+ Ours (1_{st})	WRN-34-10	**63.98**	**35.36**	**31.63**	**29.8**
	+ Ours (1_{st}+2_{nd})	WRN-34-10	**64.71**	**35.73**	**31.41**	**30.2**

- Auto Attack (AA): popular adversarial attack method
- Compare with TRADES and AWP-TRADES on CIFAR10/100
- Improvements on clean accuracy and AA accuracy

Summary

- Adversarial training / smooth weights / SWA → Adversarial training on weight distribution → Taylor expansion → Train on 0-th, 1-nd and 2-rd Taylor terms

- Empirical support

  1. Improve clean accuracy and adversarial robustness on Wide-Resnet

  2. Ablation experiment on hyper-parameter

Thanks