Maximilian Müller*, Tiffany Vlaar+, David Rolnick+, Matthias Hein*
*University of Tübingen and Tübingen AI Center, +McGill University and Mila - Quebec AI Institute

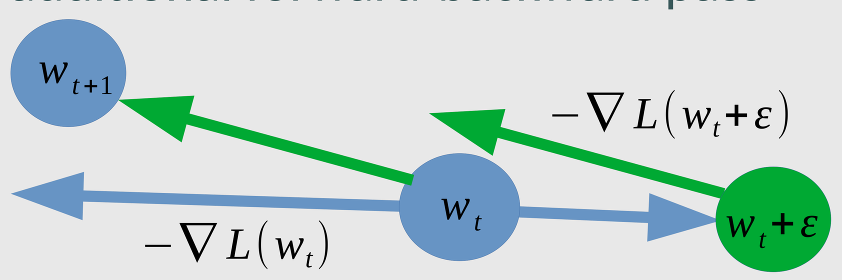
Normalization Layers Are all that Sharpness-Aware Minimization Needs

Sharpness-Aware Minimization

- Optimization method designed to implicitly maximize the flatness in weight space during gradient descent:

$$\min_{\mathbf{w}} \max_{\|\epsilon\|_p < \rho} \mathcal{L}(\mathbf{w} + \epsilon)$$

- In practice: approximate inner maximization with 1-step and use $\nabla \mathcal{L}(\mathbf{w} + \epsilon)$ for batch-wise weight update instead of $\nabla \mathcal{L}(\mathbf{w}) \longrightarrow$ resulting SAM-algorithm requires additional forward-backward pass



- Adaptive variant (ASAM) has objective:

$$\min_{\mathbf{w}} \max_{\|T_w^{-1}\epsilon\|_p < \rho} \mathcal{L}(\mathbf{w} + \epsilon)$$

- T_w is a normalization operator (diagonal matrix) making the perturbation (partly) invariant to rescaling of the parameters
- The 1-step solution of the inner maximization

$$\epsilon = \rho \frac{T_w^2 \nabla \mathcal{L}(\mathbf{w})}{\|T_w \nabla \mathcal{L}(\mathbf{w})\|_2} \text{ for } p = 2$$

$$\epsilon = \rho T_w \text{sign}(\nabla \mathcal{L}(\mathbf{w})) \text{ for } p = \infty$$

- Justifications for empirical success of SAM and its variants remain inconclusive

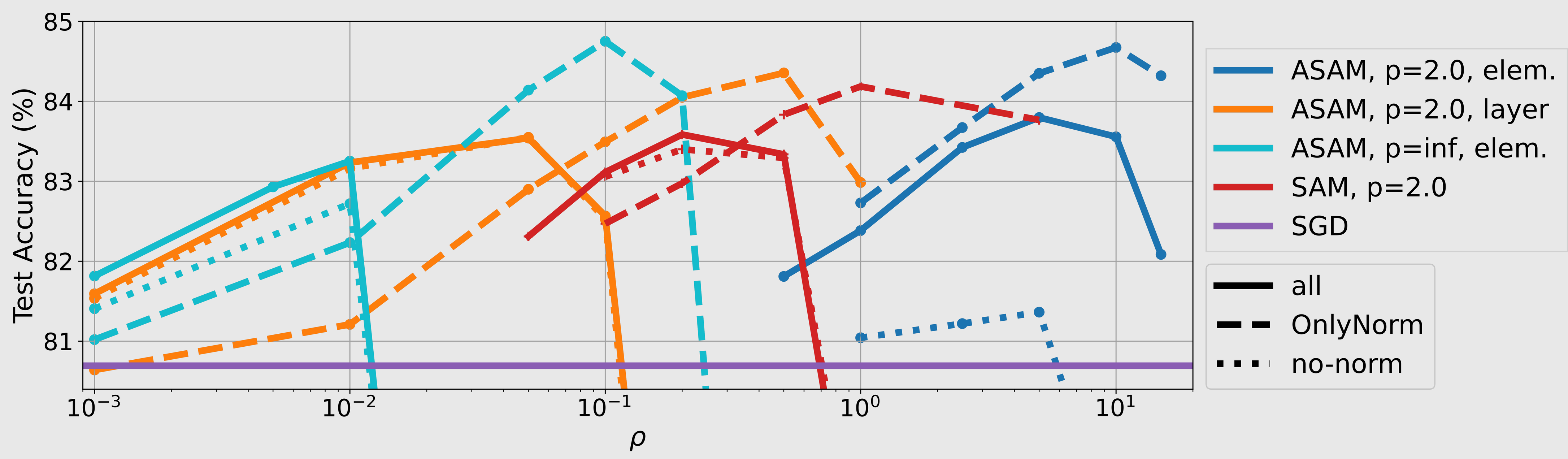
SAM-ON

- Normalization layers normalize input x with mean μ and variance σ^2

$$Norm(x) = \gamma \times \frac{x - \mu}{\sigma} + \beta$$

- We propose *SAM-ON*, which perturbs **only** γ and β (typically 0.1% of all parameters) in the adversarial step of SAM
- For ResNets with BatchNorm and VisionTransformers with Layernorm we observe that *SAM-ON* either matches the performance of the conventional SAM algorithm (called *SAM-all*), or even outperforms it

ResNets on CIFAR-100



- *SAM-ON* (dashed) improves generalization performance across a range of SAM-variants compared to *SAM-all* and vanilla SGD (shown is a WRN28-10, more models in paper!)
- omitting the normalization layers in the perturbation (no-norm, dotted) can harm training

Central Message

Applying Sharpness-Aware Minimization **only to the normalization layers** of a network typically enhances its performance

ViT-S/32 on ImageNet

- For AdamW as base optimizer, *SAM-ON* improves strongly over the vanilla optimizer and either improves over *SAM-all* or performs on par
- For Lion as base optimizer, *SAM-ON* always improves over *SAM-all* and the vanilla optimizer

		AdamW			Lion		
		vanilla	SAM-all	SAM-ON	vanilla	SAM-all	SAM-ON
ID	ImageNet	$66.89^{\pm0.04}$	$71.47^{\pm0.12}$	$71.37^{\pm0.026}$	$68.20^{\pm0.02}$	$71.90^{\pm0.19}$	**72.64** $^{\pm0.14}$
	ImageNetV2	$48.43^{\pm0.48}$	$53.61^{\pm0.11}$	$53.67^{\pm0.29}$	$50.20^{\pm0.01}$	$54.20^{\pm0.27}$	**55.38** $^{\pm0.09}$
OOD	ImageNetR	$25.04^{\pm0.04}$	$31.56^{\pm0.48}$	**32.98** $^{\pm0.10}$	$25.61^{\pm0.04}$	$32.17^{\pm0.41}$	**33.87** $^{\pm0.47}$
	ImageNetA	$4.72^{\pm0.15}$	$5.21^{\pm0.05}$	$5.19^{\pm0.18}$	$5.45^{\pm0.19}$	$5.01^{\pm0.22}$	**5.77** $^{\pm0.21}$
	ImageNetSketch	$13.68^{\pm0.24}$	$18.50^{\pm0.44}$	**19.35** $^{\pm0.17}$	$14.47^{\pm0.02}$	$18.22^{\pm0.34}$	**20.48** $^{\pm0.12}$
	ObjectNet	$11.32^{\pm0.39}$	$13.75^{\pm0.12}$	$13.55^{\pm0.25}$	$12.06^{\pm0.02}$	$13.93^{\pm0.40}$	**15.35** $^{\pm0.13}$
adv. rob.	$\ell_2, \epsilon = 0.25$	$19.67^{\pm0.47}$	$37.53^{\pm0.69}$	**41.16** $^{\pm0.24}$	$22.01^{\pm0.78}$	$38.52^{\pm0.66}$	**43.12** $^{\pm0.97}$
	$\ell_2, \epsilon = 0.50$	$5.47^{\pm0.18}$	$17.71^{\pm0.61}$	**22.72** $^{\pm0.25}$	$6.63^{\pm0.46}$	$19.03^{\pm0.92}$	**24.27** $^{\pm1.34}$
	$\ell_\infty, \epsilon = 0.25/255$	$33.45^{\pm0.80}$	$48.08^{\pm0.14}$	**49.34** $^{\pm0.08}$	$35.31^{\pm0.08}$	$49.57^{\pm0.60}$	**51.37** $^{\pm0.99}$
	$\ell_\infty, \epsilon = 0.5/255$	$14.98^{\pm0.18}$	$29.68^{\pm0.09}$	**32.46** $^{\pm0.15}$	$15.86^{\pm0.13}$	$31.68^{\pm0.62}$	**34.23** $^{\pm1.73}$

Sharpness

SAM-ON models are sharper, yet generalize better. Shown is logit-normalized ℓ_∞ m-sharpness, averaged over three models per method for a WRN-28 on CIFAR-100.

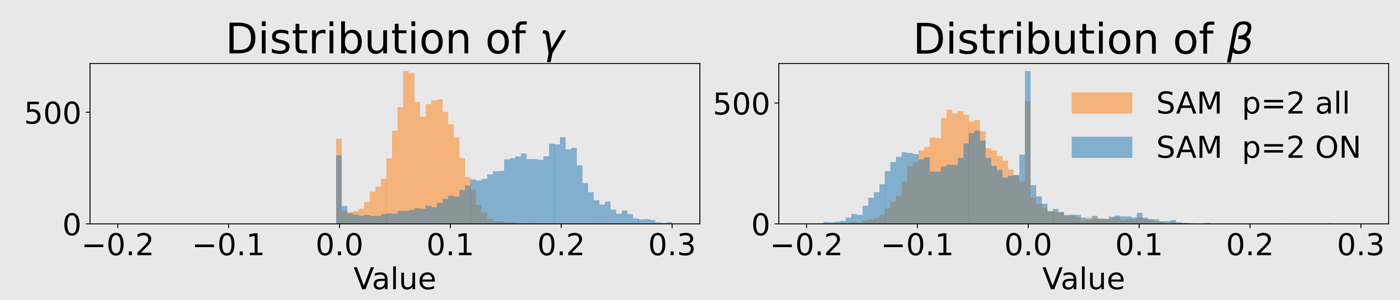
		SGD	SAM-all	SAM-ON
	Test Accuracy (%)	$80.71^{\pm0.2}$	$83.11^{\pm0.3}$	$\mathbf{84.19}^{\pm0.2}$
ℓ_∞-sharp.	20 steps, $\rho = 0.003$	$0.071^{\pm0.000}$	$\mathbf{0.048}^{\pm0.001}$	$0.090^{\pm0.005}$
	20 steps, $\rho = 0.007$	$0.433^{\pm0.002}$	$\mathbf{0.309}^{\pm0.011}$	$0.585^{\pm0.018}$
	1 step, $\rho = 0.01$	$0.204^{\pm0.005}$	$\mathbf{0.183}^{\pm0.002}$	$0.315^{\pm0.010}$
	1 step, $\rho = 0.03$	$0.809^{\pm0.003}$	$\mathbf{0.769}^{\pm0.017}$	$0.843^{\pm0.007}$

Other sparse perturbation approaches

Other sparse perturbation approaches are less effective than *SAM-ON*, especially when probed at very high sparsity levels.

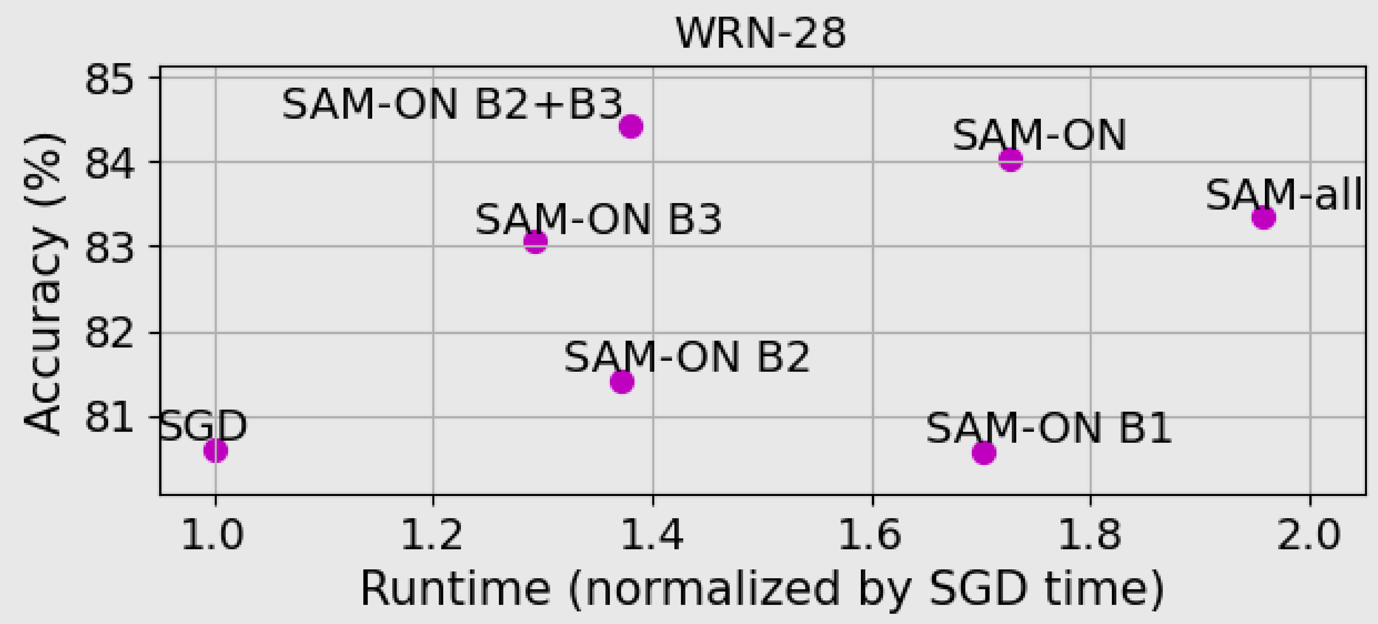
	SAM	SAM-ON	Random Mask	SSAM-F	
Sparsity	0%	99.95%	99.95%	50%	99.95%
Test Accuracy (%)	83.11±0.3	**84.19**±0.2	80.97±0.2	83.94±0.1	83.14±0.1

SAM-ON induces a parameter shift



For *SAM-ON* the distribution of γ shifts towards larger values

Computational savings



SAM-ON reduces the computational load in practice, and further gains might be achieved by perturbing the normalization layers of selected blocks (B1-B3)

Paper & code

Find paper and code at `github.com/mueller-mp/SAM-ON`.