萌芽型共同研究課題: EX24402

Smoothness-Adaptive Sharpness-Aware Minimization for Finding Flatter Minima

Hiroki Naganuma*(Mila, UdeM), Junhyung Lyle Kim*(Rice CS), Anastasios Kyrillidis(Rice CS), Ioannis Mitliagkas(Mila, UdeM)

SAM (Sharpness Aware Minimization) overview

- Flatness of loss function is known to correlated with generalozation.
- Sharpness Aware Minimization (a.k.a SAM) is introduced leverage the benefit of flatter minima.
- Mathematically, we formalize to solve

$$\min_w L^{\mathrm{SAM}}(w) + \lambda\|w\|_2^2 \quad \text{where} \quad L^{\mathrm{SAM}}(w) := \max_{\|\epsilon\|_p \le \rho} L(w+\epsilon)$$

where $p \ge 0$ is the perturbation radius, which is a hyperparameter that needs to be tuned , and $p \in [1, \infty]$ can be changed, but $p = 2$ is typically used.

- Further, in practice, the maximization step is approximated with a single (stochastic) gradient ascent step:

$$\hat{\epsilon}(w) = \rho \frac{\nabla_w L(w)}{\|\nabla_w L(w)\|} \approx \arg\max_{\|\epsilon\|_p \le \rho} L(w+\epsilon)$$

where the gradient can be computed efficiently via: $\nabla_w L^{\mathrm{SAM}} \approx \nabla_w L(w)|_{w+\hat{\epsilon}}$

- Finally, the above approximation results in the following SAM update:

$$w_{t+1} = w_t - \eta_t \nabla L\left(w_t + \rho_t \frac{\nabla L(w_t)}{\|\nabla L(w_t)\|_2}\right)$$

Pros and Cons of Sharpness Aware Minimization

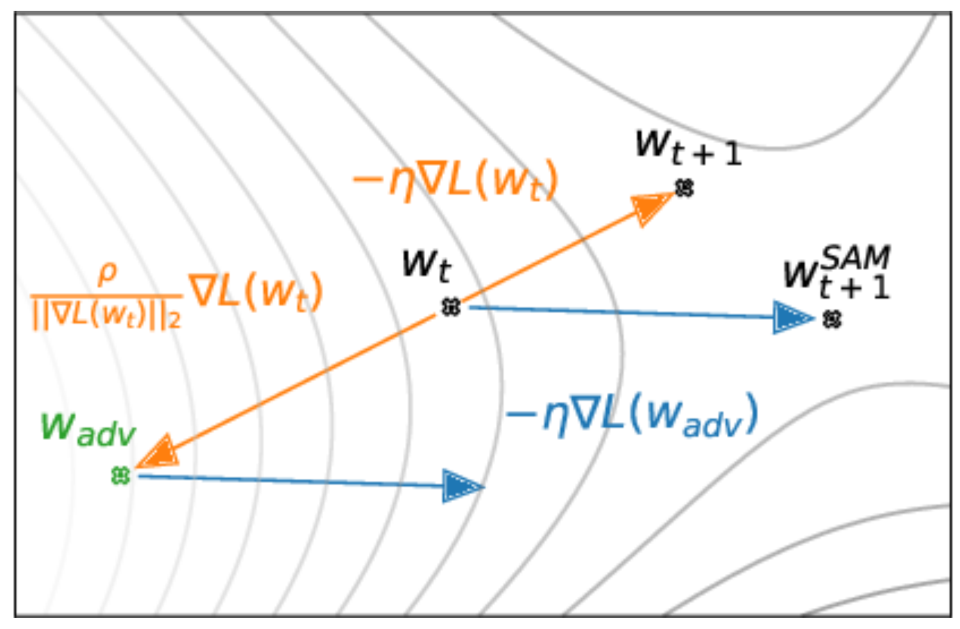

Fig: Schematic of the SAM parameter update (taken from [P Foret et al, 2017])

Pros:

- Better generalization performance because of flatter minima.
- Outperform other optimizer under distributionl shift.
- Minimizer curvature such as $Tr(H)$, and which is proposed as indicator for downstream task .

Cons:

- Difficult to conduct comprehensive hyperparameter-search on large models because two calculations, gradient ascent and gradient descent, are required for one update.
- Compared to SGD, there are more hyperparameters to be tuned.

Proposed algorithms (SA-SAM)

Algorithm ¯ Smoothness-Adaptive Sharpness-Aware Minimization (SA-SAM)

1: **input**: $w_0 \in \mathbb{R}^d$, $\eta_0 > 0$, $\theta_0 = +\infty$, and ξ_0.

2: $w_1 = w_0 - \eta_0 \nabla L\left(w_0 + \rho_0 \frac{\nabla L(w_0;\xi_0)}{\|\nabla L(w_0;\xi_0)\|_2}; \xi_0\right)$

3: **for** each round $t = 1, \dots$ **do**

4: Sample mini-batch ξ_t — **update η_t : SAM descent step size**

5: $\boxed{\eta_t = \min\left\{\frac{\|w_t - w_{t-1}\|}{2\|\nabla L(w_t;\xi_t) - \nabla L(w_{t-1};\xi_t)\|}, \sqrt{1+\theta_t}\eta_{t-1}\right\}}$

6: $\boxed{\rho_t = \sqrt{\eta_t}}$ **update ρ_t : SAM ascent step size**

7: $w_{t+1} = w_t - \eta_t \nabla L\left(w_t + \rho_t \frac{\nabla L(w_t;\xi_t)}{\|\nabla L(w_t;\xi_t)\|_2}; \xi_t\right)$

8: $\theta_t = \eta_t / \eta_{t-1}$

9: **end for**

Why is the step size $\eta = 1/\beta$ popular?

- β-smooth functions: $|L(y) - L(x) - \langle \nabla L(x), y - x\rangle| \le \frac{\beta}{2}\|y-x\|^2 \ \forall x, y$
- Gradient descent: $w_{t+1} = w_t - \eta \nabla L(w_t)$
- Descent lemma:

$$L(w_{t+1}) \le L(w_t) + \langle \nabla L(w_t), w_{t+1} - w_t\rangle + \frac{\beta}{2}\|w_{t+1} - w_t\|^2$$
$$= L(w_t) - \eta\left(1 - \frac{\eta \cdot \beta}{2}\right)\|\nabla L(w_t)\|^2$$

- $\eta = 1/\beta$ is the "optimal" step size for gradient descent.

Connection to Edge of Stability

- The local smoothness-adaptive step size, constrained by the global smoothness constant, aligns with the "edge of stability" theory, suggesting an implicit regularization that favors less sharp minima.
- Combining the properties of the adaptive step size with the SAM strategy, the SA-SAM algorithm enhances implicit regularization that encourages convergence to these flatter minima.

Adaptive Step Size via Local Smoothness

- [Malisky & Mishchenko, 2020] proposed the following step size for (centralized) gradient descent:

$$\eta_t = \min\left\{\frac{\|w_t - w_{t-1}\|}{2\|\nabla L(w_t) - \nabla L(w_{t-1})\|}, \sqrt{1+\theta_{t-1}}\eta_{t-1}\right\}, \quad \theta_{t-1} = \eta_{t-1}/\eta_{t-2}$$

- The first condition approximates the local smoothness

$$\|\nabla L(w_t) - \nabla L(w_{t-1})\| \le \beta_t \cdot \|w_t - w_{t-1}\|, \quad \forall t = 1, 2, \dots$$

and the second condition ensures η_t to not increase too fast.

- Following [Andriushchenko & Flammarion (2022, Theorem 2)], we adapt the above step size to the ascent descent step size in SAM setting:

$$\rho_t \leftarrow \sqrt{\eta_t}$$

Experimental Results: SA-SAM exhibits superior performance in all settings without difficulty of tuning

Experimental Setup

- Dataset: CIFAR10 (train and validation), CIFAR10-C(test)
- Model Arch: VGG-19, ViT_Small(Vision Transformer Small)
- Optimizers: SGD, MSGD(MomentumSGD), Adam, SAM, SA-SGD(Smoothness Aware SGD), SA-SAM(Smoothness Aware SAM)

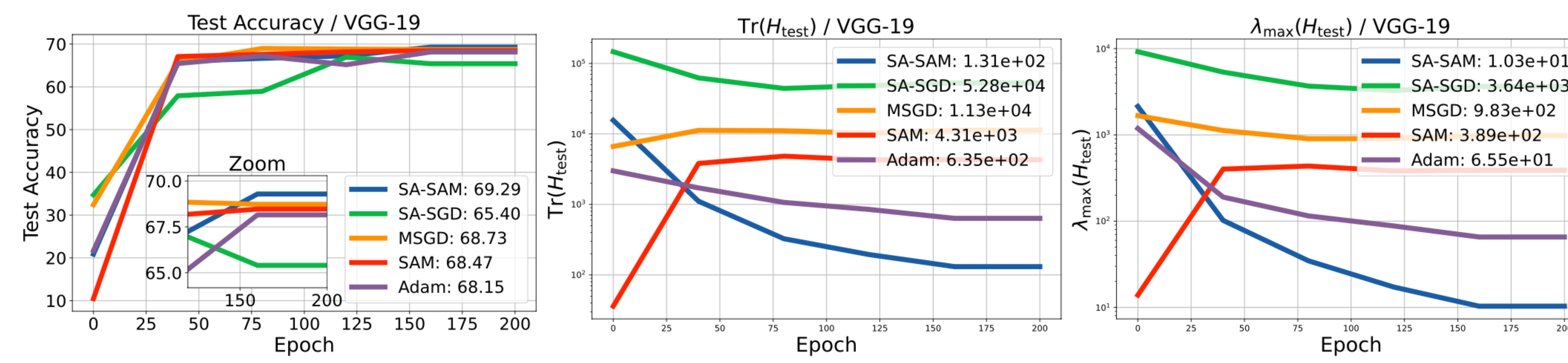

Fig: Test accuracy (OOD generalization) and the curvature information, measured by the $Tr(H)$ and the leading eigenvalue $\lambda_{max}(H)$ of the Hessian, on CIFAR10-C dataset trained with VGG-19 for the five optimizers.

Experimental Results

- (Top Right): SA-SAM, not only achieves the best test accuracy but also converges to flatter minima.
- (Bottom Left): SA-SAM generally exhibits better performance compared to other optimizers. As expected, the performance of MSGD, SAM, and Adam varies significantly with LR, in contrast to SA-SAM and SA-SGD.
- (Bottom Right): SA-SAM consistently achieves lower curvature for both models, regardless of the hyperparameters. In the extreme case, $\mathrm{Tr}(H)$ of SA-SAM is smaller than that of Adam by 10^{24}.

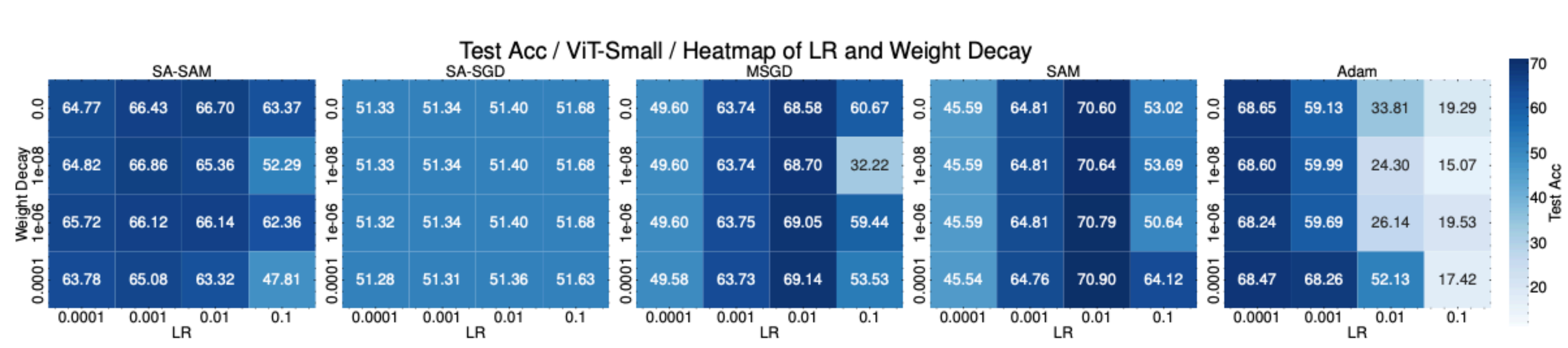

Fig: Heatmap of test accuracy for considered optimizers with various learning rates and weight decay parameters, for CIFAR10-C dataset trained with ViT-Small.

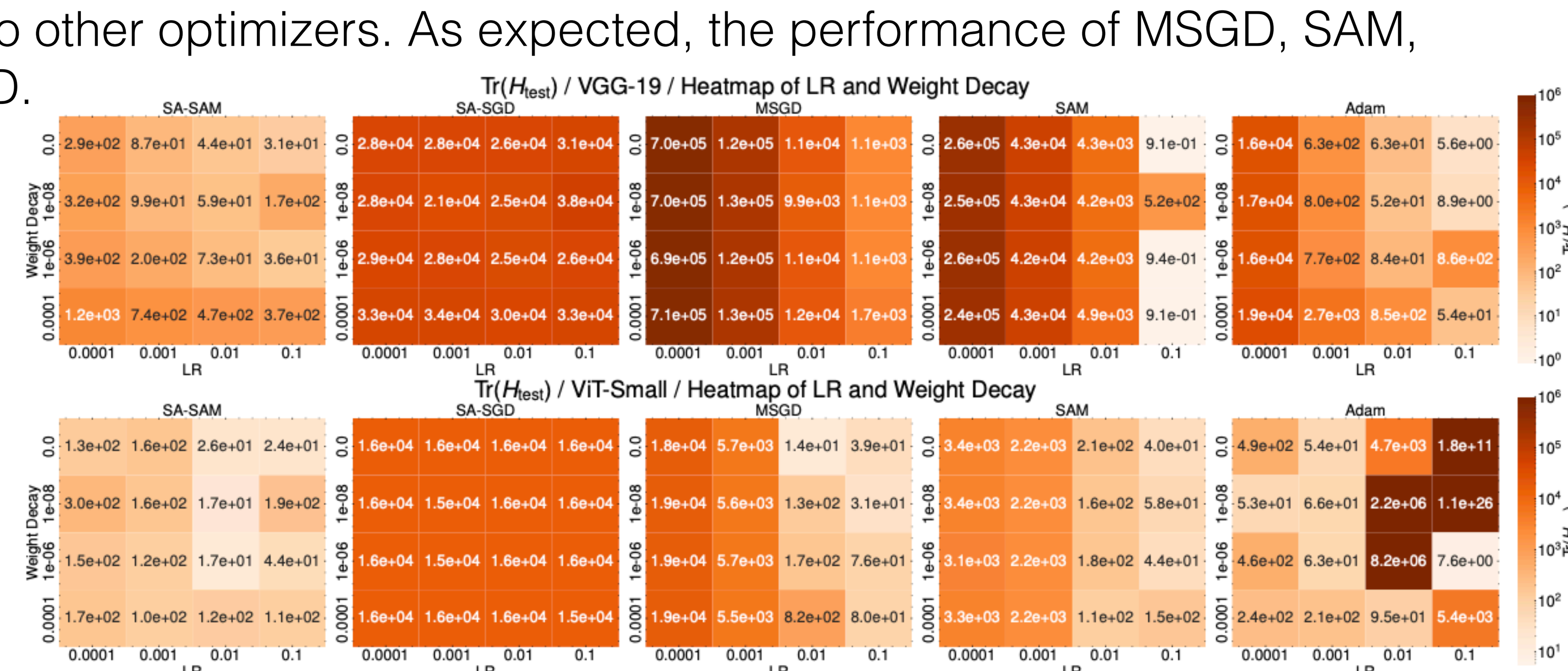

Fig: Heatmap of the test $\mathrm{Tr}(H)$ and $\lambda_{max}(H)$ for considered optimizers with various learning rates and weight decay parameters, for CIFAR10-C dataset trained with VGG-19 (top) and ViT-Small (bottom)