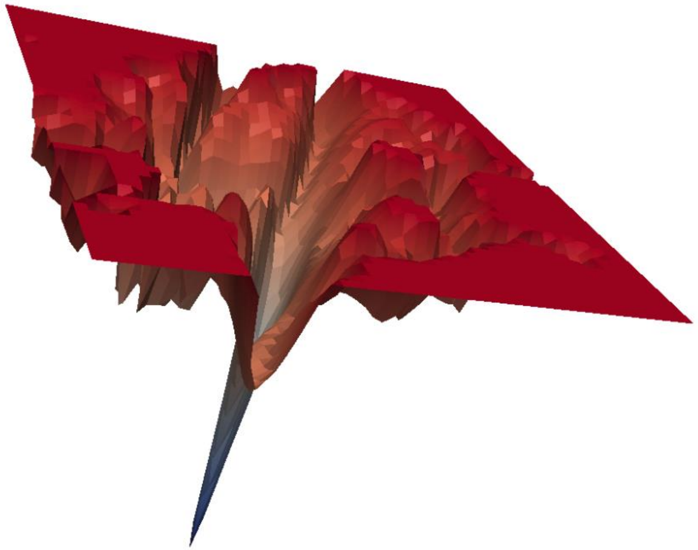
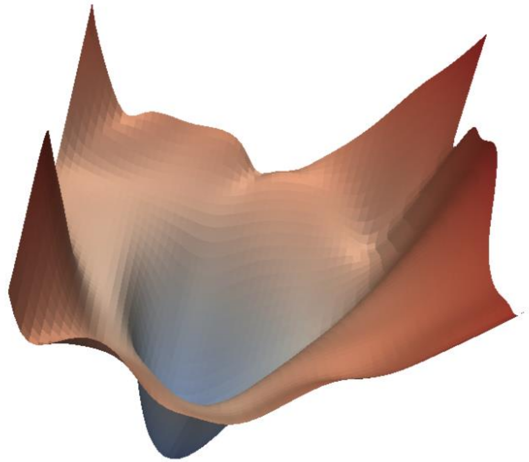
Sharpness-aware minimization

REFORM reading group
02/04

Hippolyte Wallaert





All figures in this presentation are taken from https://arxiv.org/abs/2206.06232 and https://arxiv.org/abs/2010.01412

Context and motivation

- Context :

Ensuring better generalisation of over-parametrised networks has been a subject for a long time (Batch Norm, Dropout, Data augmentation ...)

Different types of bad generalisation : fundamental reason (overfitting), label noise, adversarial perturbations ...

- Setting :

Over-parametrized networks admit a lot of different global minima with different generalisation performance - **how to find the best one ?**

Intuition :

"Flatter" minima (where loss changes slowly in a neighborhood) are thought to generalize better than "sharp" ones.

Mitigations :

Reparametrization : Minima can be made arbitrarily "sharp" or "flat" by simple weight scaling without changing the model's output functions... so **why do SAM still works well ?**

Sharpness

- Given a training dataset $S_{train} = \{x_i, y_i\}_{i=1}^{n}$, a classifier with weights w and $L_S(w)$ the empirical loss of the classifier on a subset S $\subseteq S_{train}$, the sharpness is defined as :

$$s(w, \mathcal{S}) = \max_{\|\delta\|_2 \leq \rho} \left[L_{\mathcal{S}}(w + \delta) - L_{\mathcal{S}}(w) \right]$$

Usually $S = S_{train}$ or S is a batch of size m.

- An informal motivation is given by the following result (even though experiments illustrate it is loose) :

Theorem (stated informally) 1. *For any $\rho > 0$, with high probability over training set $\mathcal{S}$ generated from distribution $\mathscr{D}$,*

$$L_{\mathscr{D}}(\boldsymbol{w}) \leq \max_{\|\boldsymbol{\epsilon}\|_2 \leq \rho} L_{\mathcal{S}}(\boldsymbol{w} + \boldsymbol{\epsilon}) + h(\|\boldsymbol{w}\|_2^2/\rho^2),$$

where $h : \mathbb{R}_+ \to \mathbb{R}_+$ is a strictly increasing function (under some technical conditions on $L_{\mathscr{D}}(\boldsymbol{w})$).

Sharpness-aware minimization

- Replace objective by :

$$\min_{w} \max_{\|\epsilon\|_2 \leq \rho} L_S(w+\epsilon) + \lambda \|w\|_2^2$$

- 1st order approximation of L_S to solve the inner maximisation problem in one step :

$$\epsilon^* = \rho \cdot \frac{\nabla_w L_S(w)}{\|\nabla_w L_S(w)\|_2}$$

- Interpretation as an *extra-gradient method* BUT with an adversary ("+") anticipation (after removing L-2 normalization in ϵ^*):

$$w_{t+1} = w_t - \gamma \nabla L\left(w_t + \rho \nabla L(w_t)\right).$$

m-sharpness

- In practice training is usually performed using batches of size m which changes slightly the update rule during training.

$$\underbrace{\max_{\|\delta\|_2 \le \rho} \frac{1}{|\mathcal{S}|} \sum_{i:(x_i,y_i)\in\mathcal{S}} \ell_i(w+\delta) - \ell_i(w)}_{n\text{-sharpness}} \implies \underbrace{\frac{1}{m}\sum_{j=1}^{m} \max_{\|\delta\|_2 \le \rho} \frac{1}{n} \sum_{i\in\mathcal{S}_j} \ell_i(w+\delta) - \ell_i(w)}_{m\text{-sharpness}}$$

- Update rule :

$$w_{t+1} = w_t - \frac{\gamma}{m}\sum_{j=1}^{m} \nabla L_j(w_t + \rho \nabla L_j(w_t))$$

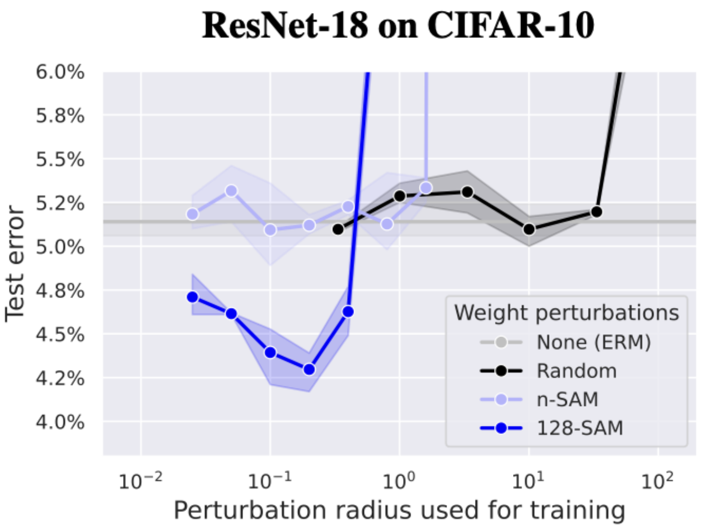
Performance of SAM

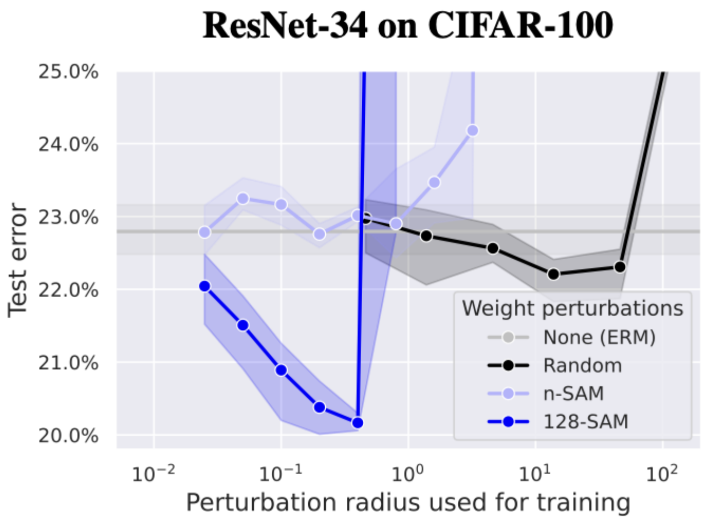
Model	Epoch	SAM Top-1	SAM Top-5	Standard Training (No SAM) Top-1	Standard Training (No SAM) Top-5
ResNet-50	100	**22.5**$_{\pm 0.1}$	6.28$_{\pm 0.08}$	22.9$_{\pm 0.1}$	6.62$_{\pm 0.11}$
	200	**21.4**$_{\pm 0.1}$	5.82$_{\pm 0.03}$	22.3$_{\pm 0.1}$	6.37$_{\pm 0.04}$
	400	**20.9**$_{\pm 0.1}$	5.51$_{\pm 0.03}$	22.3$_{\pm 0.1}$	6.40$_{\pm 0.06}$
ResNet-101	100	**20.2**$_{\pm 0.1}$	5.12$_{\pm 0.03}$	21.2$_{\pm 0.1}$	5.66$_{\pm 0.05}$
	200	**19.4**$_{\pm 0.1}$	4.76$_{\pm 0.03}$	20.9$_{\pm 0.1}$	5.66$_{\pm 0.04}$
	400	**19.0**$_{\pm <0.01}$	4.65$_{\pm 0.05}$	22.3$_{\pm 0.1}$	6.41$_{\pm 0.06}$
ResNet-152	100	**19.2**$_{\pm <0.01}$	4.69$_{\pm 0.04}$	20.4$_{\pm <0.0}$	5.39$_{\pm 0.06}$
	200	**18.5**$_{\pm 0.1}$	4.37$_{\pm 0.03}$	20.3$_{\pm 0.2}$	5.39$_{\pm 0.07}$
	400	**18.4**$_{\pm <0.01}$	4.35$_{\pm 0.04}$	20.9$_{\pm <0.0}$	5.84$_{\pm 0.07}$

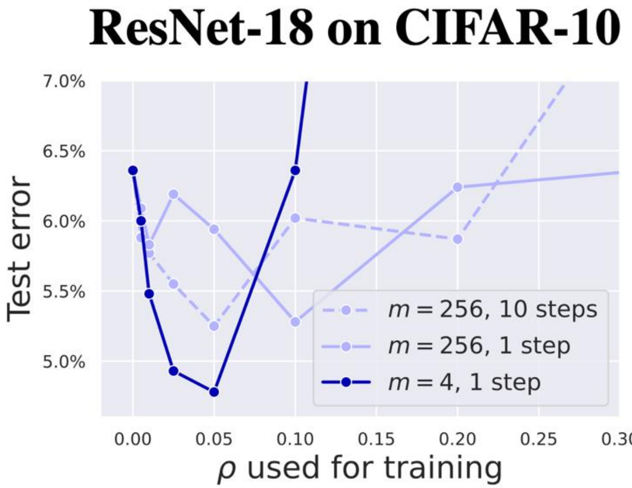
Table 2: Test error rates for ResNets trained on ImageNet, with and without SAM.

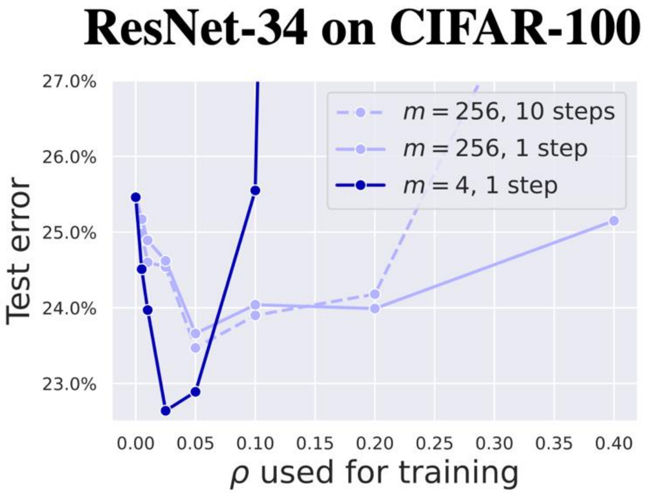
- Very easy to implement and good generalization results
- Improving classifier robustness to label noise
- Improving generalization if used for fine-tuning

Observations









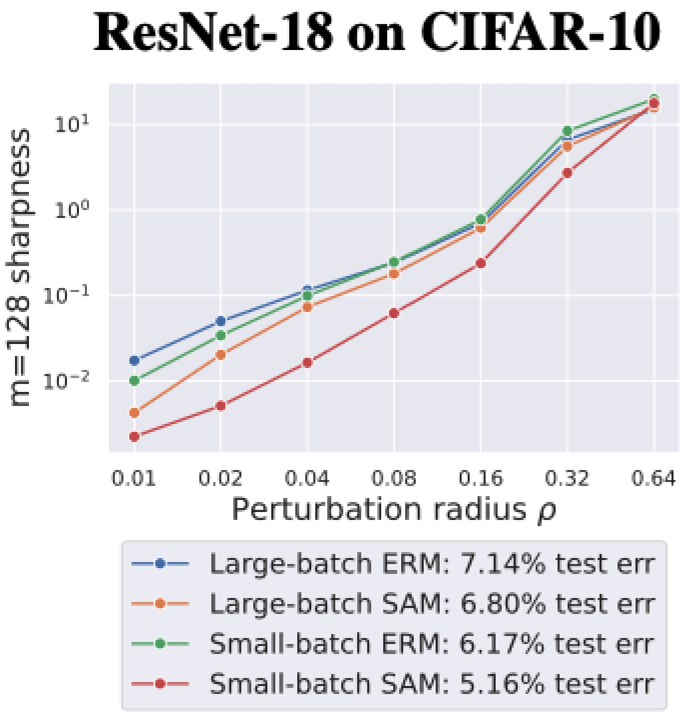
- Generalization is better with lower m values
- Solving the inner maximisation problem more precisely (using 2nd order term and/or multiple gradient iterations) does NOT improve generalization

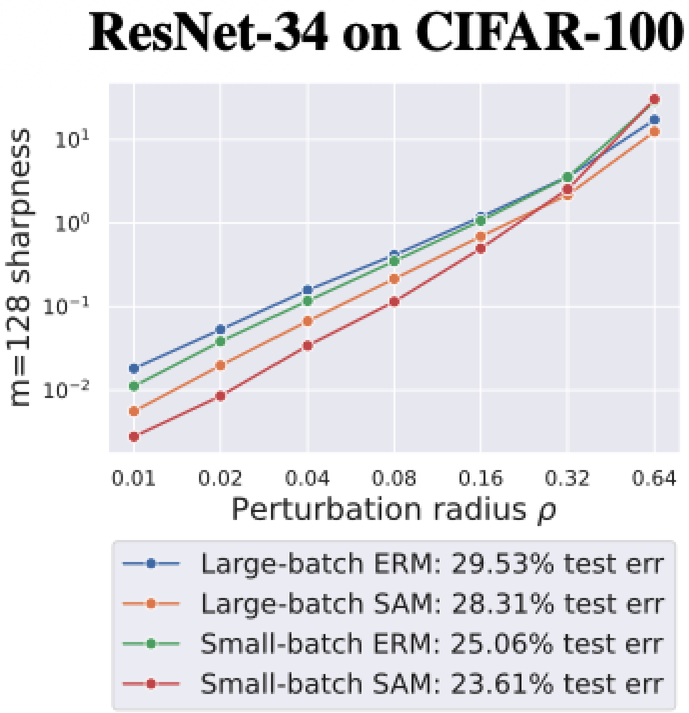
Challenging current understanding

Question : Does flatter minima mean better generalisation ?

Observation : Not necessarily.

None of the radii ρ gives the correct ranking between the methods according to their test error, although m-sharpness ranks correctly SAM and ERM for the same batch size.





Generalization because of implicit bias

- Implicit bias : the solution obtained using a specific optimization algorithm is biased to have a certain property among all the global minimizers

Eg : in linear regression, gradient descent initialized at 0 converges to the solution with minimal L-2 norm

- Core result on implicit bias of gradient descent for spare regression using diagonal linear networks (Woodworth et. al. 2020):

Task :

$$\min_{w_+, w_- \in \mathbb{R}^d} L(w) := \frac{1}{4n} \sum_{i=1}^{n} \left(\langle w_+^2 - w_-^2, x_i \rangle - y_i \right)^2$$

$$\text{where } \beta = w_+^2 - w_-^2$$

$$w_+(0) = w_-(0) = \alpha \mathbf{1}_d, \quad \alpha > 0$$

Bias (solving using GD) :

$$\boxed{\beta_\infty^\alpha = \underset{\beta \in \mathbb{R}^d \text{ s.t. } X\beta = y}{\arg\min} \ \phi_\alpha(\beta),}$$

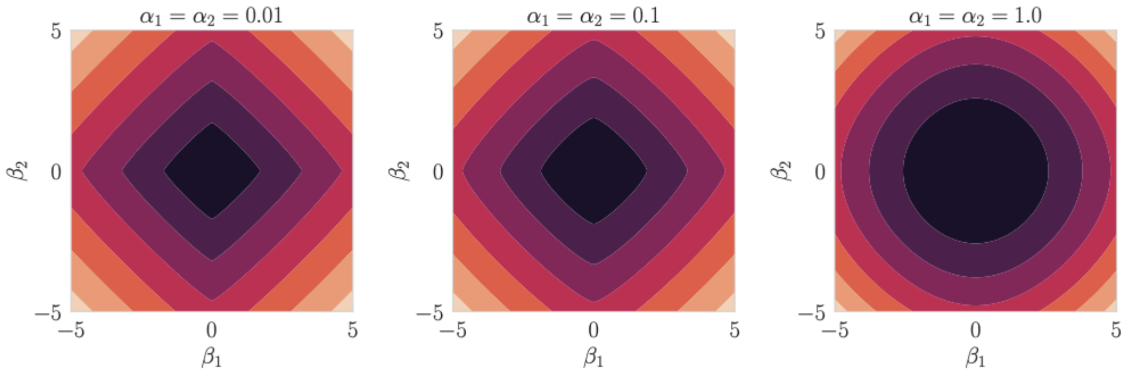


Figure 5: Illustration of the hyperbolic entropy $\phi_\alpha(\beta)$ for $\beta \in \mathbb{R}^2$ that interpolates between $\|\beta\|_1$ for small α and $\|\beta\|_2$ for large α.

Empirical results in Non-Linear Networks

- <u>Setting</u> : a one hidden layer ReLU network applied to a simple 1D regression problem

12 data points and 100 ReLU trained using full batch GD with ERM and SAM

- <u>Result</u> : SAM favours sparse combination of ReLUs which is more stable across different initializations

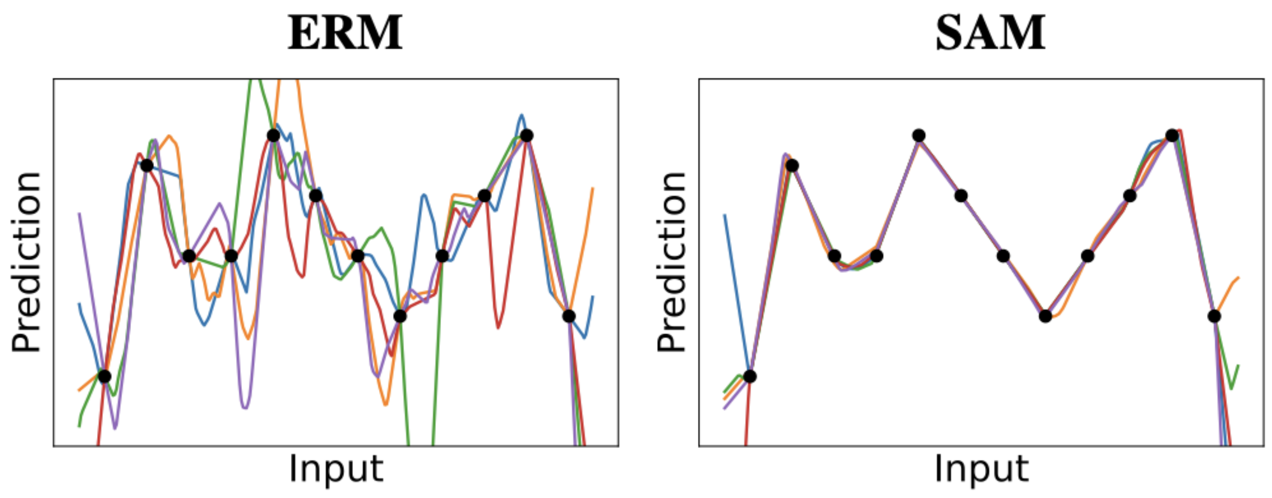


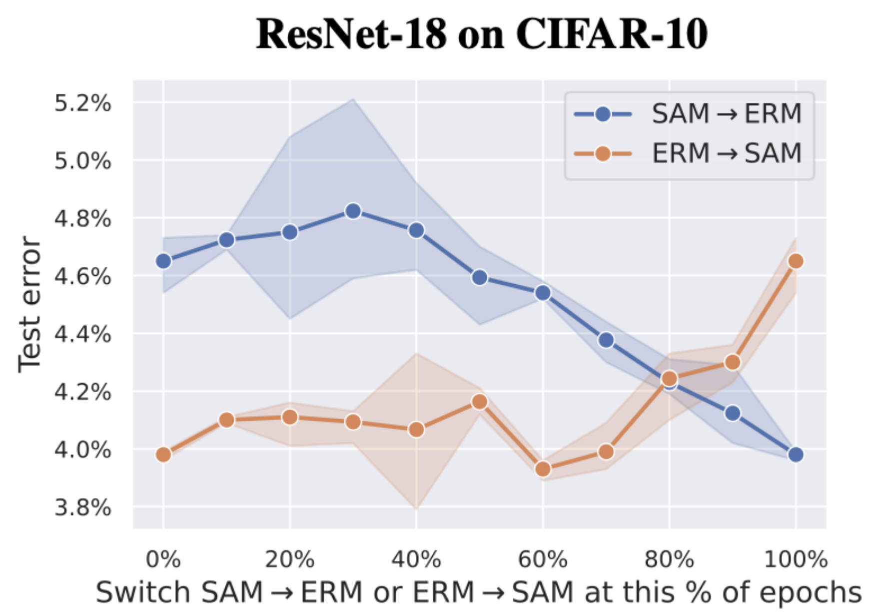
Figure 7: The effect of the implicit bias of ERM vs. SAM for a one hidden layer ReLU network trained with full-batch gradient descent. Each run is replicated over five random initializations.

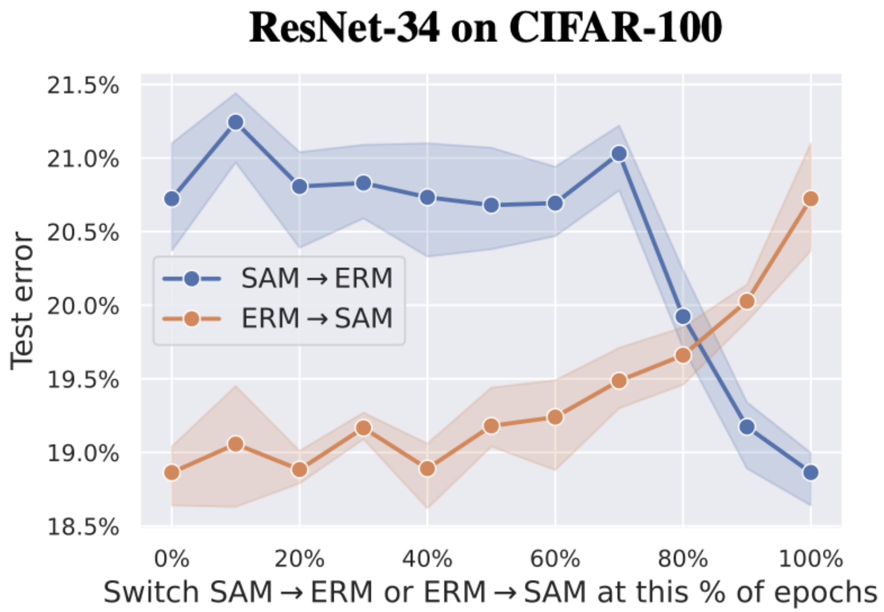
Additional results

- Question : in which part of training is it important to steer towards better-generalizing minimum ?

- Observations :

1) A method that is used at the beginning of training has little influence on the final performance
2) The performance is very continuous relative to time of switching ! Suggests convergence in a connected valley where some directions generalize better





Additional results

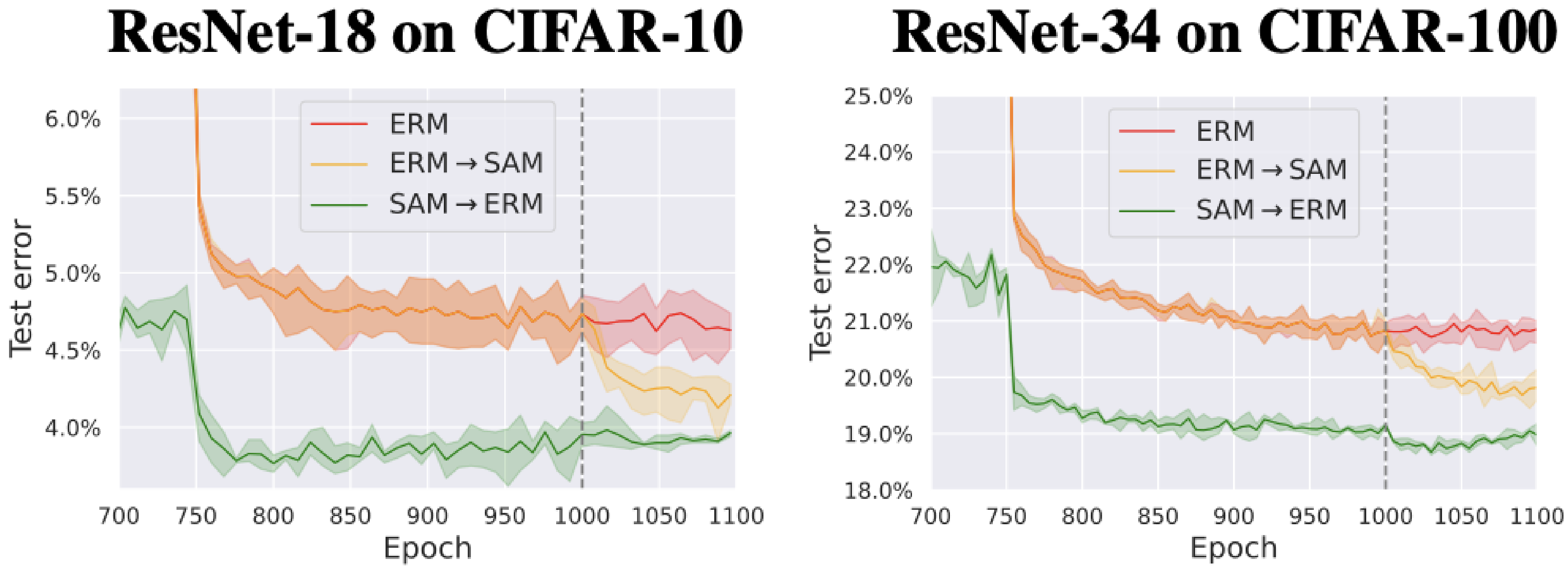


Figure 9: Test error over epochs for ERM compared to ERM $\rightarrow$ SAM and SAM $\rightarrow$ ERM training where the methods are switched only at the end of training. In particular, we can see that SAM can gradually escape the worse-generalizing minimum found by ERM.

Discussion

- How to look at sharpness ? Should sharpness be considered a proxy for a deeper geometric property we haven't fully defined yet ? **Probably not**

- **Useful intuition : think of this technique as adversarial training in the weight space**

- Given the fact that SAM success seems to come from implicit bias rather than sharpness, is flatness necessarily a desirable property of minnimizers ? **No, see other presentation**