Multi-Step FGSM as an Alternative Way to Solve Sharpness-Aware Minimization Problem

Ivan Shchekotov [*1] **Kirill Tamogashev** [*1] **Emil Alkin** [*1]

Abstract

Sharpness-Aware Minimization is an algorithm that helps to address the problem of sharp minima which prevents Deep Learning models from generalization. It was first formulated by (Foret et al., 2020; Zhuang et al., 2022) and since then many improvements and modifications were invented. In this work we consider the variant that uses Multi-Step Fast Gradient Singed Method to calculate the best perturbation of the weights at each step.

1. Introduction

The problem of finding optimal weight for the model lies in the heart of Machine Learning. A lot of ML models are over-parameterized. Thus finding a solution that allows model to generalize well is hard. Often models just memorize the training dataset and fail if they are fed the data they have not seen. This problem is attributed to the geometry of the loss surface, and as it was shown in many papers (see, for example (Keskar et al., 2016)) bad generalization occurs because the loss of the model is sharp at this point. More formally this means that for a given optimal point w the loss $L(w) \not\approx L(w+\epsilon)$, where ϵ is a small perturbation.

Such sharp minima points, although they give a small value for the loss, are generally bad solutions for the model. Learning such solutions results in a poor performance of the model and failure to generalize properly. This problem is especially evident while training large-batch models, as it was shown in (Keskar et al., 2016).

The purpose of our project is to explore the dynamics of the special case of SAM – SAM coupled with FSGM (Goodfellow et al., 2014). In particular, we attempt to show the performance of multi-step FGSM (Kurakin et al., 2016) in comparison to vanilla SAM. We also explain, that SAM with FGSM is closely connected to SAM, and is actually a special case of it.

Our project repost is organized as follows: in **Part 2** we discuss the previous research conducted on SAM. We talk about both experiments and theoretical models of SAM. We present different perspectives from which one can consider it. In **Part 3** we present the method we use in our experiments. In particular, we state that sharpness-aware minimization and be seen as an adversarial attack and present the well-know FGSM model (Goodfellow et al., 2014; Kurakin et al., 2016) that can be used to solve that problem. In **Part 4** we present our own numerical experiments. We show, that although being inferior to SAM, FGSM outperforms the gradient descent with no regularization on a CIFAR 10. We also show that it beats SAM on CIFAR 100, which we believe is an interesting result. We also discuss the obtained results and attempt to give and intuitive explanation for them. In **Part 5** we summarize out project results.

2. Literature review

2.1. Initial problem: Vanilla SAM

One of the algorithms that seeks to address the problem of sharp minima is called Sharpness-Aware Minimization (SAM). It was proposed concurrently by (Foret et al., 2020; Zhuang et al., 2022). The approach introduced in the papers is based on adding a small perturbation ϵ to the loss. The authors argue that it improves the ability of the model to generalize. The addition of ϵ leads to the optimization of the following criterion:

$$\min_{w} \max_{\|\epsilon\|_p \le \rho} L(w+\epsilon) + \lambda \|w\|_2$$

$$\epsilon^* = \arg\max_{\|\epsilon\|_p \le \rho} L(w+\epsilon) = \arg\max_{\|\epsilon\|_p \le \rho} \epsilon^T \nabla_w L(w)$$

And optimal ϵ can be found as a solution to the dual norm problem:

$$\epsilon^* = \rho \cdot sign(\nabla_w L(w)) \frac{|\nabla_w L(w)|^{q-1}}{\|\nabla_w L(w)\|_q^{\frac{q}{p}}}$$

[*]Equal contribution [1]Skolkovo Institute of Science and Technology. Correspondence to: Kirill Tamogashev <Kirill.Tamogashev@skoltech.ru>.

Project report for Bayesian Methods of Machine Learning Course at Skoltech.

This algorithm serves as the basis for similar approaches to the sharp minima problem and it lays a foundation for future research in this area.

2.2. Improvements to Vanilla SAM

Although the SAM algorithm gives good results, it has some issues concerning the stability and the scale of parameters. The former issue was addressed by the (Liu et al., 2022). The fact that the loss landscape of the deep neural network can be sharp may lead to the gradient being unstable. This significantly decreases the speed of convergence and the resulting quality. The authors propose to add a small noise δ to smoothen the optimization procedure. The noise is sampled from gaussian distribution $\delta \sim \mathcal{N}(0, \gamma^2 I)$ and plugged into the weights in the following way:

$$w_{SAM} = w + \delta + \rho \frac{g(w+\delta)}{\|g(w+\delta)\|_2}$$

where $g(w) = \nabla L_w(w)$ As authors show this addition of perturbation improves the quality of the model.

The problem of scale can be dealt with by an improvement proposed in (Kwon et al., 2021). The authors propose a scale-invariant version of SAM: Adaptive SAM. The algorithm deals with the case, in which parameters of the model grow significantly in scale without directly affecting the model performance. However, this growth significantly affects SAM performance. To solve that issues the authors introduce to an addition of a scaling factor to the loss:

$$\max_{\|T^{-1}\epsilon\|_p \leq \rho} L_(w+\epsilon) - L(w)$$

Here T is a normalization operator that facilitates the proper scaling of parameters. As authors show, this approach improves the classification results as well as robustness of the algorithm to label noise.

2.3. Probabilistic perspective

The other direction of the research of Sharpness-Aware Minimization is to try to grasp the theoretical justification for why the algorithm improves training and bring about broader theoretical perspective, in which SAM can be considered. One such attempt is to view the SAM as Variational Inference. This perspective is presented in (Ujváry et al., 2022). They show that one can rewrite the generalized SAM objective as a logarithm of density of some probability distribution:

$$L(\mu, \Sigma) = \max_{\epsilon^T \Sigma \epsilon \leq p} [L(\mu+\epsilon) - L(\mu)] + L(\mu) + \alpha\|\mu\|_2^2$$

This allows to theoretically connect two basic ideas underneath the the SAM: enforcing the flatness of minima and perturbing the weights of the model with gaussian noise during training. The authors show, that such reformulation leads to finding the posterior distribution:

$$p(\theta|\mathcal{D}) \propto e^{L(\theta)} \cdot q(\theta),$$

where $q(\theta)$ is chosen to be gaussian with $\theta = \{\mu, \Sigma\}$. The latter problem is rigorously studied and can be solved using Mean Field approximation and reparametrization trick, as the authors show.

2.4. Recent Advances

The most recent advances in the research of SAM seek to gain a better theoretical understanding of the reasons, that properly explain the ability of SAM to avoid sharp minima and produce better generalizations. (Wen et al., 2022) attempt to give a proper definition of sharpness. Moreover, (Andriushchenko & Flammarion, 2022) rigorously study the aspect of generalization. They show that Sharpness-Aware Minimization algorithms always produces better generalization for a certain class of problem and provide theoretical analysis that justifies this conclusion.

3. Our contribution

3.1. Using Multi-Step FGSM to find ϵ^*

The addition of ϵ can also be considered as an adversarial attack on the loss function. The notion of adversarial attacks was firstly introduced in (Goodfellow et al., 2014) and since then as rigorously studied. The algorithm that allows to find the best adversarial attack is called Fast Gradient Signed Method (FGSM)(Goodfellow et al., 2014) and can be written by

$$\epsilon^* = \rho \cdot sign\left(\nabla_w L(w)\right)$$

where ρ is a hyperparameter. Such method is efficient and can give nice results as shown in (Goodfellow et al., 2014). The extension of this method is called Multi-Step FGSM and it is explained in (Kurakin et al., 2016). The basic idea is to run FGSM for a number of steps. As opposed to referenced paper we do not employ clipping after each step due to the fact that clipping across weight parameters would lead to low expressiveness of the model. The resulting formula looks as follows (we slightly change notation to better facilitate our research purpose):

$$\mathrm{w}_0^{adv} = w$$
$$w_{n+1}^{adv} = w_n^{adv} + \rho \cdot sign\left(\nabla_w L(w_n^{adv})\right)$$

4. Experiments

4.1. Empirical Setting

In this project, we focus on evaluating performance of Vanilla SAM and FGSM methods in classical setting of image classification on toy datasets (CIFAR10 and CIFAR100). CIFAR10 contains 32x32 coloured images of 10 classes, 6000 image per class and is divided into splits of 50000 training images and 10000 validation images. CIFAR100 contains images of the same size with 1000 classes, 600 images per class and also has the sample split distribution.

We employ basic augmentations containing padding by four pixels, random cropping and pixel normalization.

As a model we choose ResNet56 from (He et al., 2016) with 0.85 M parameters. The decision to take this model is due to the fact that this model is known to give high classification accuracy on CIFAR10 dataset. We have conducted multiple experiments with ImageNet version of ResNet50 (approx. 25M parameters) provided in Torchvision package (Marcel & Rodriguez, 2010), but it's performance is subpar comparing to smaller model on 32x32 images due to aggressive downsampling which is employed in the first convolutional layers leading to significant information loss.

However such choice of model due to different structure of CIFAR100 dataset leads to subpar results on it. We hypothesize that in this scenario we are able to see clear performance boost from using SAM or FGSM methods.

SAM[1] method has one hyperparameter ρ, which we take to be equal to 0.05 as it was shown to have strong performance across variety of datasets in original paper. Original SAM uses 2-norm for finding optimum epsilon.

In our FGSM experiments we set $\rho = 0.00025$, as larger values of ρ in our experiments lead to significantly slower learning dynamics. We denote multi-n-step FGSM as FGSM(n).

All experiments were run in the same setup. We decide not to use any scheduler to compare bare methods without trying to push for the best result. We choose SGD optimizer with constant learning rate 0.1, momentum 0.9 and weight decay 0.0001. We choose batch size of 512 for train epoch and batch 128 for test epoch with full true inference on each test image every 10 epochs. We compute top-1 accuracy and top-5 accuracy on test data to measure quality of generalization. We run our experiments for 360 epochs to accumulate more data about each method. Due to long training time of multi-step methods (i.e. SAM, FGSM) CIFAR100 results are not fully incomplete, so we provide results only on 120 epochs for this dataset. This is not an issue due to the fact that it is difficult for the current model to generalize further on CIFAR100. On complete runs we observe that there is no significant improvement over 120 epoch runs, however it is still interesting to obtain complete results for a clear picture.

[1]Pytorch SAM implementation was taken from https://github.com/davda54/sam

Table 1. Test Error Rates on CIFAR10

Epoch	SAM		FGSM(2)		Standard training	
	Top-1	Top-5	Top-1	Top-5	Top-1	Top-5
90	**12.35**	0.41	13.36	0.44	14.63	0.60
180	**10.41**	0.32	11.17	0.31	12.13	0.49
270	**10.10**	0.34	10.23	0.28	10.48	0.43
360	**9.36**	0.44	9.88	0.35	10.43	0.38

Table 2. Test Error Rates on CIFAR100

Epoch	SAM		FGSM(1)		Standard training	
	Top-1	Top-5	Top-1	Top-5	Top-1	Top-5
30	**44.75**	16.19	71.86	39.67	45.10	16.14
60	**39.07**	13.35	41.53	14.26	41.51	14.32
90	38.33	12.88	**36.14**	11.32	39.00	13.09
120	36.91	12.57	**34.90**	10.76	40.88	14.69

Our main results are presented in Tables 1 and 2. We present result on each of the method, choosing the best FGSM(n) among $n \in \{1, 2, 3\}$. We should take in account that SAM and FGSM make multiple backward passes, so it is not completely fair to compare standard training with other methods across the same epoch, however in most cases we notice that SAM and FGSM methods quality observed at two times less epochs is better or matching the quality of standard training method. Moreover, further SAM and FGSM training does not lead to overfitting.

On CIFAR10 SAM outperforms all of the suggested methods by sufficient margin, while FGSM(2) tries to keep up with SAM and also consistently beats standard training procedure.

On CIFAR100 we observe that FGSM beats SAM with described setup.

4.2. Discussion of the results

FGSM is closely related to SAM: if we choose $p = \infty, q = 1 : \frac{1}{p} + \frac{1}{q} = 1$ then we will get coinciding optimal epsilon for both methods equal to FGSM. From SAM perspective this is equivalent finding maximum epsilon by absolute value (Chebyshev norm) which is less than hyperparameter ρ value, which results in FGSM rule.

The main and crucial difference between the steps in these two methods is that we need to explicitly choose ρ and thus ϵ^* in FGSM, when in SAM our hyperparameter is not the optimal epsilon itself. The fact that SAM relies on the value of the gradient and not only on the sign gives it more power to regularize the gradient descent in comparison to FGSM. Therefore, FGSM is just a special (reduced) case of SAM.

We also find that performing FGSM(n), where $n > 3$ massively hinders learning capabilities of the model and results in failure in our experiments. We hypothesize that this happens, because running FGSM a lot of steps perturbes the weights to the extent they no longer retain valuable information about the data. In other words, FGSM pushes the gradients far away from the true optimal path, thus model fails to learn. As we stated before, performing up to 3 steps seems enough.

5. Conclusions

As we have shown in our experiments, the optimal ϵ for SAM can be found with FGSM algorithm. It successfully beats the gradient descent with no regularization, and in certain cases it manages to outperform the classic version of SAM. We also explain the relation of FGSM to SAM and show that the former is just a special case of the latter. Overall, FGSM can perform as well as SAM, but it need a careful hyperparameter tuning.

References

Andriushchenko, M. and Flammarion, N. Towards understanding sharpness-aware minimization, 2022. URL https://arxiv.org/abs/2206.06232.

Foret, P., Kleiner, A., Mobahi, H., and Neyshabur, B. Sharpness-aware minimization for efficiently improving generalization. *CoRR*, abs/2010.01412, 2020. URL https://arxiv.org/abs/2010.01412.

Goodfellow, I. J., Shlens, J., and Szegedy, C. Explaining and harnessing adversarial examples, 2014. URL https://arxiv.org/abs/1412.6572.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2016. doi: 10.1109/cvpr.2016.90. URL http://dx.doi.org/10.1109/cvpr.2016.90.

Keskar, N. S., Mudigere, D., Nocedal, J., Smelyanskiy, M., and Tang, P. T. P. On large-batch training for deep learning: Generalization gap and sharp minima, 2016. URL https://arxiv.org/abs/1609.04836.

Kurakin, A., Goodfellow, I., and Bengio, S. Adversarial examples in the physical world, 2016. URL https://arxiv.org/abs/1607.02533.

Kwon, J., Kim, J., Park, H., and Choi, I. K. ASAM: adaptive sharpness-aware minimization for scale-invariant learning of deep neural networks. *CoRR*, abs/2102.11600, 2021. URL https://arxiv.org/abs/2102.11600.

Liu, Y., Mai, S., Cheng, M., Chen, X., Hsieh, C.-J., and You, Y. Random sharpness-aware minimization. In Oh, A. H., Agarwal, A., Belgrave, D., and Cho, K. (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=htUvh7xPoa.

Marcel, S. and Rodriguez, Y. Torchvision the machine-vision package of torch. In *Proceedings of the 18th ACM International Conference on Multimedia*, MM '10, pp. 1485–1488, New York, NY, USA, 2010. Association for Computing Machinery. ISBN 9781605589336. doi: 10.1145/1873951.1874254. URL https://doi.org/10.1145/1873951.1874254.

Ujváry, S., Telek, Z., Kerekes, A., Mészáros, A., and Huszár, F. Rethinking sharpness-aware minimization as variational inference, 2022. URL https://arxiv.org/abs/2210.10452.

Wen, K., Ma, T., and Li, Z. How does sharpness-aware minimization minimize sharpness?, 2022. URL https://arxiv.org/abs/2211.05729.

Zhuang, J., Gong, B., Yuan, L., Cui, Y., Adam, H., Dvornek, N., Tatikonda, S., Duncan, J., and Liu, T. Surrogate gap minimization improves sharpness-aware training, 2022. URL https://arxiv.org/abs/2203.08065.