Flatness-Aware Regularization for Robust Generalization in Deep Neural Networks

Chisom Chibuike*[1, 2]
chisom.chibuike.246093@unn.edu.ng

Fatimo Adebanjo*[1]
adebanjofatimo2000@gmail.com

[1]ML Collective, [2]University of Nigeria

Understanding the geometry of the loss landscape in deep neural networks (DNNs) is central to machine learning research because of its role in generalization. [2] suggested that solutions in flatter regions generalize better than those in sharper ones. Empirical evidence supports this: [3] showed that small-batch training often converges to flatter minima, and methods like Entropy-SGD were designed to encourage exploration of wider valleys. Yet the link between loss landscape geometry and generalization remains unsettled. [1] argued that sharp minima can also generalize, though without strong empirical validation. The prevailing view, however, is that flatter minima are usually linked to better generalization, while sharper minima tend to cause overfitting.

Motivated by this connection between geometry and generalization, we propose a flatness-aware regularization technique that explicitly penalizes the curvature of the loss surface by incorporating an estimate of the trace of the squared Hessian into the training loss. We define the total loss function for a mini-batch (x, y) as:

$$\mathcal{L}_{total} = \mathcal{L}_{task}(f_\theta(x), y) + \lambda \cdot \frac{Tr(H^2)}{B} \tag{1}$$

where $\mathcal{L}_{task}(f_\theta(x), y)$ denotes the base task-specific loss, λ is a regularization coefficient controlling penalty strength, B is the batch size, and $Tr(H^2)$ is the trace of the squared Hessian with respect to θ, estimated via Hutchinson's method. By penalizing this curvature measure, the optimizer is encouraged to find flatter regions of the loss surface, which we hypothesize leads to more robust generalization on unseen data.

Computing the full Hessian is notoriously expensive for modern deep networks. To make our curvature penalty feasible, we used Hutchinson's stochastic trace estimator to approximate $Tr(H^2)$ efficiently. This involves multiplying the Hessian by randomly sampled probe vectors and using their quadratic forms to estimate the trace. This leverages the identity $E[v^T H^2 v] = Tr(H^2)$ for random v with zero mean and unit variance.

We used a 2-layer MLP with ReLU activation on CIFAR-100 to examine how flatness-aware (FA) regularization reshapes both optimization and generalization. With $\lambda = 0$ the model (no FA) reached a final accuracy of $\sim$26.3% and applied no curvature penalty. Introducing FA changed the behavior: a moderate penalty ($\lambda = 0.01$) gave the best balance with final accuracy $\sim$27.0% and peak accuracy $\sim$27.8%, while maintaining lower curvature than the baseline. Larger penalties ($\lambda = 0.1$ and $\lambda = 1.0$) further reduced curvature estimates but slightly decreased final accuracy ($\sim$26.7% and $\sim$25.8%). The efficiency trade-off was clear: the baseline finished in $\sim$60s, whereas all FA settings increased training time substantially ($\sim$2200-2400 s due to the second order computation), with little added benefit at higher λ. Taken together, these results show that FA regularization is most effective when applied at a moderate strength, striking a balance between fitting capacity, generalization, and computational efficiency.

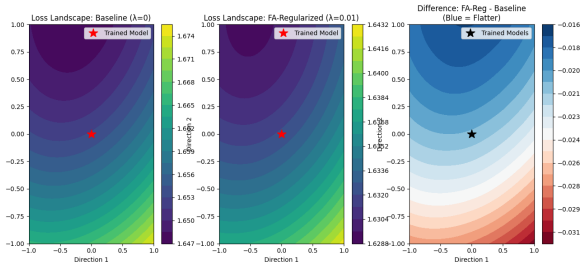

Figure 1: Loss landscape comparison showing the difference between Baseline ($\lambda = 0$) and FA-Regularized ($\lambda = 0.01$) models. Blue areas indicate flatter loss for the FA-Regularized model.

Table 1: Performance of FA regularization on CIFAR-100.

Lambda	Final Acc	Best Acc	Avg Flatness	Train Time
0.000	0.2633	0.2800	0.000000	60.855
0.001	0.2633	0.2833	23.050173	2290.775
0.010	0.2700	0.2783	10.739012	2433.535
0.100	0.2667	0.2967	4.144254	2203.875
1.000	0.2583	0.2983	1.549933	2337.865

Acknowledgements We thank ML Collective (https://mlcollective.org/) for the invaluable mentorship and supportive research environment provided throughout the course of this work. We also extend our sincere appreciation to Elvis Dohmatob (https://mila.quebec/en/directory/elvis-dohmatob) for his constructive feedback and insightful suggestions, which greatly improved the quality of this study.

References

[1] Dinh, Laurent, Razvan Pascanu, Samy Bengio and Yoshua Bengio. "Sharp Minima Can Generalize For Deep Nets." *International Conference on Machine Learning* (2017).

[2] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Flat minima. *Neural Comput.* 9, 1 (Jan. 1, 1997), 1–42. https://doi.org/10.1162/neco.1997.9.1.1

[3] Keskar, Nitish Shirish, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy and Ping Tak Peter Tang. "On Large-Batch Training for Deep Learning: Generalization Gap and Sharp Minima." *ArXiv* abs/1609.04836 (2016): n. pag..

*Signifies equal contribution