Optimization for Deep Learning

SGD Stability

Kun Yuan

Peking University

Main contents in this lecture

- GD stability
- SGD stability
- Sharpness-aware minimization

SGD performs better than GD

- SGD is proposed to reduce the computational burden of GD, but it is often observed to outperform GD in accuracy when training neural network

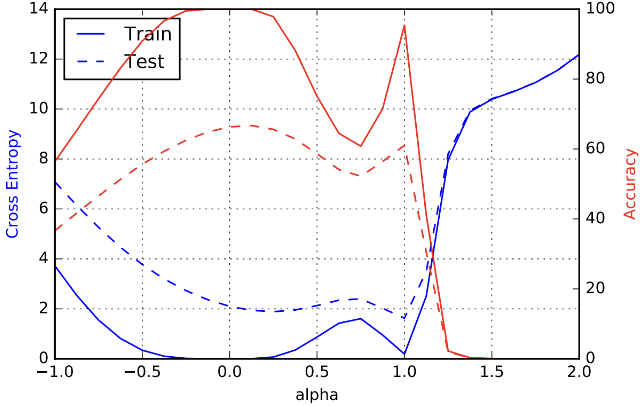
Experiment	Mini-batching	Epochs	Steps	Modifications	Val. Accuracy %
Baseline SGD	✓	300	117,000	-	95.70($\pm$0.11)
Baseline FB	✗	300	300	-	75.42($\pm$0.13)
FB train longer	✗	3000	3000	-	87.36($\pm$1.23)
FB clipped	✗	3000	3000	clip	93.85($\pm$0.10)
FB regularized	✗	3000	3000	clip+reg	95.36($\pm$0.07)
FB strong reg.	✗	3000	3000	clip+reg+bs32	95.67($\pm$0.08)
FB in practice	✗	3000	3000	clip+reg+bs32+shuffle	95.91($\pm$0.14)

Table 2: Summary of validation accuracies in percent on the CIFAR-10 validation dataset for each of the experiments with data augmentations considered in Section 3. All validation accuracies are averaged over 5 runs.

Figure 4: Taken (Geiping et al., 2021)

Flat minima hypothesis

- In neural network, flat solutions generalize better (Keskar et al., 2016)



- It is conjectured that SGD converges to flatter solutions than GD

An intuition behind flat minima hypothesis

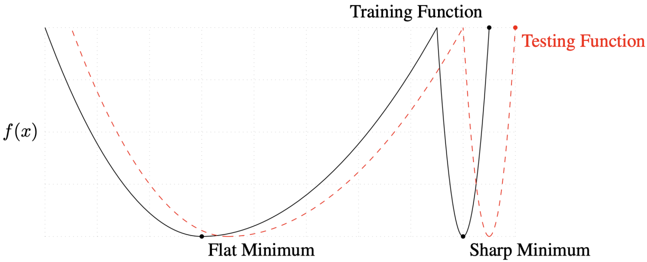


Figure: Substantial difference exists between training function and test function around sharp minima (Keskar et al., 2016).

The escape phenomenon

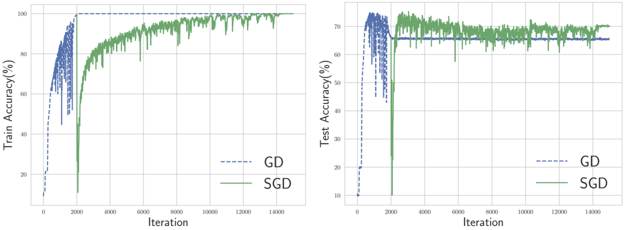


Figure: Fast escape phenomenon in SGD (Wu et al., 2018).

- GD solution is unstable for SGD
- It is conjectured that SGD escapes from GD's sharp minima and converges to a flatter solution

GD stability

- Let $H = \nabla^2 f(x^\star)$ where $x^\star$ is a local minima. Assume x_k is close to $x^\star$:

$$\begin{aligned} x_{k+1} - x^\star &= x_k - x^\star - \gamma \nabla f(x_k) \\ &= x_k - x^\star - \gamma\big(\nabla f(x_k) - \nabla f(x^\star)\big) \\ &= (I - \gamma H)(x_k - x^\star) \\ &= (I - \gamma H)^{k+1}(x_0 - x^\star) \end{aligned}$$

- $x^\star$ is stable for GD if

$$\lambda_{\max}\big(I - \gamma H\big) \leq 1 \quad \iff \quad \lambda_{\max}(H) \leq \frac{2}{\gamma}$$

- Otherwise GD escapes from $x^\star$ at rate $(1 - \gamma \lambda_{\max}(H))^k$; exponentially fast

GD stability

- This result implies that, given learning rate γ, the curvature at $x^\star$ must be flat with eigenvalues less than or equal to $2/\gamma$.

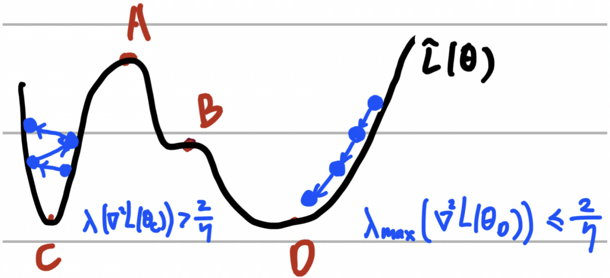


Figure: GD converges to flat solution and escapes from sharp solution (by Dr. Lei Wu in Peking University).

SGD stability

Theorem 1 (Wu et al. (2018))

The global minimum $x^\star$ is linearly stable for SGD with learning rate γ and batch size B if the following condition is satisfied

$$\lambda_{\max}\Big(\underbrace{(I-\gamma H)^2}_{\text{GD condition}} + \underbrace{\frac{\gamma^2(N-B)}{B(N-1)}\Sigma}_{\text{gradient noise}} \Big) \le 1$$

where $\Sigma \succeq 0$ is the covariance matrix of gradient noise at $x^\star$.

It implies that SGD converges to an even flatter solution than GD given the same learning rate γ.

Consider the scalar scenario and let $B=1$, $H=h$, $\Sigma=s$. SGD will converge to $x^\star$ with $h \le \frac{2(1-s)}{\gamma}$, which is flatter than GD with $h \le \frac{2}{\gamma}$.

Why does SGD escape from sharp minima? The noise!

- To show the intuition, we consider a simplified quadratic problem

$$f(x) = \frac{1}{2} x^T H x.$$

- SGD can be regarded as GD with noise:

$$x_{k+1} = x_k - \gamma(\nabla f(x_k) + s_k) = (I - \gamma H)x_k - \gamma s_k$$

  where s_k is gradient noise with $\mathbb{E}[s_k] = 0$ and $\mathbb{E}[s_k s_k^T] = \Sigma$.

- SGD evolves as follows (Wu et al., 2022)

$$\mathbb{E}[f(x_{k+1})] = \mathbb{E}[r(x_k) f(x_k)] + \frac{\gamma^2}{2} \mathrm{Tr}(H\Sigma)$$

  where $r(x) = 1 - 2\gamma \frac{x^T H^2 x}{x^T H x} + \gamma^2 \frac{x^T H^3 x}{x^T H x}$. (The proof is leaved as exercise)

Why does SGD escape from sharp minima? The noise!

$$\mathbb{E}[f(x_{k+1})] = \mathbb{E}[r(x_k)f(x_k)] + \frac{\gamma^2}{2}\mathrm{Tr}(H\Sigma)$$

- $r(x) < 1$ when γ is sufficiently small; drives $f(x)$ to decrease
- The noise term $\mathrm{Tr}(H\Sigma)$ drives x_k away from the local minima
- Noise covariance Σ aligns well with H in neural network (Wu et al., 2022; Zhu et al., 2019); **the sharper the curvature is, the stronger the noise is**
- $\mathrm{Tr}(H\Sigma)$ is a strong force to drive SGD away from sharp minima

Why does SGD escape from sharp minima?

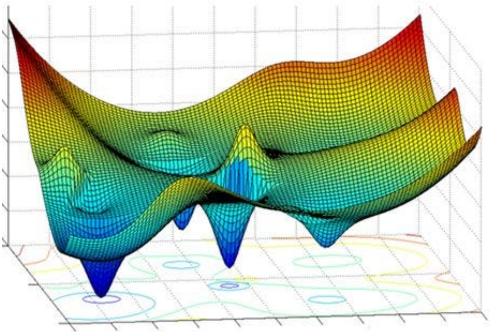


Figure: An illustration of non-convex landscape[1].

[1] https://medium.com/analytics-vidhya/journey-of-gradient-descent-from-local-to-global-c851eba3d367

Sharpness-aware minimization

- To converge to a flatter solution, we consider a new problem

$$\min_{x \in \mathbb{R}^d} \quad f_{\mathrm{SAM}}(x) \quad \text{where} \quad f_{\mathrm{SAM}}(x) = \max_{\|\epsilon\| \le \rho} f(x + \epsilon)$$

  which is called sharpness-aware minimization (SAM) (Foret et al., 2020).

- The above problem is to seek a solution whose neighborhood is flat

- Different from adversarial learning since the perturbation is added to x not ξ

Sharpness-aware minimization

- To efficiently solve the above problem, we linearize $f(x+\epsilon)$ to get

$$\max_{\|\epsilon\|\le\rho} \quad f(x) + \epsilon^\top \nabla f(x),$$

  which leads to

$$\epsilon = \frac{\rho \mathrm{sign}(\nabla f(x))|\nabla f(x)|}{\|\nabla f(x)\|} \in \mathbb{R}^d,$$

  where $\mathrm{sign}(\cdot)$ and $|\cdot|$ are element-wise operation.

- Since ϵ is related with x, we denote it as $\epsilon(x)$

Sharpness-aware minimization

- Substitute $\epsilon(x)$ into $f_{\mathrm{SAM}}(x)$, we have

$$\min_{x \in \mathbb{R}^d} \quad f_{\mathrm{SAM}}(x) \approx f(x + \epsilon(x))$$

- The gradient of $f_{\mathrm{SAM}}(x_k)$ is derived as

$$\nabla f_{\mathrm{SAM}}(x) = \nabla f(x)|_{x=x_k+\epsilon(x_k)} + [\frac{\partial \epsilon}{\partial x} \cdot \nabla f(x)]\Big|_{x=x_k+\epsilon(x_k)}$$

  where $\partial\epsilon/\partial x \in \mathbb{R}^{d\times d}$ is Jacobian matrix

- Since the second term is expensive to compute, it is ignored in SAM algorithm (Foret et al., 2020)

Sharpness-aware minimization

- SAM algorithm can be written as follows

$$\epsilon_k = \frac{\rho \text{sign}(\nabla f(x_k))|\nabla f(x_k)|}{\|\nabla f(x_k)\|}$$
$$x_{k+1} = x_k - \gamma \nabla f(x_k + \epsilon_k)$$

- In stochastic optimization, SAM iterates as follows

$$\epsilon_k = \frac{\rho \text{sign}(\nabla F(x_k;\xi_k))|\nabla F(x_k;\xi_k)|}{\|\nabla F(x_k;\xi_k)\|}$$
$$x_{k+1} = x_k - \gamma \nabla F(x_k + \epsilon_k;\xi_k)$$

- SAM is more expensive than SGD since it requires two gradient evaluations

Sharpness-aware minimization

		CIFAR-10		CIFAR-100	
Model	Augmentation	SAM	SGD	SAM	SGD
WRN-28-10 (200 epochs)	Basic	**2.7**$_{\pm 0.1}$	3.5$_{\pm 0.1}$	**16.5**$_{\pm 0.2}$	18.8$_{\pm 0.2}$
WRN-28-10 (200 epochs)	Cutout	**2.3**$_{\pm 0.1}$	2.6$_{\pm 0.1}$	**14.9**$_{\pm 0.2}$	16.9$_{\pm 0.1}$
WRN-28-10 (200 epochs)	AA	**2.1**$_{\pm <0.1}$	2.3$_{\pm 0.1}$	**13.6**$_{\pm 0.2}$	15.8$_{\pm 0.2}$
WRN-28-10 (1800 epochs)	Basic	**2.4**$_{\pm 0.1}$	3.5$_{\pm 0.1}$	**16.3**$_{\pm 0.2}$	19.1$_{\pm 0.1}$
WRN-28-10 (1800 epochs)	Cutout	**2.1**$_{\pm 0.1}$	2.7$_{\pm 0.1}$	**14.0**$_{\pm 0.1}$	17.4$_{\pm 0.1}$
WRN-28-10 (1800 epochs)	AA	**1.6**$_{\pm 0.1}$	2.2$_{\pm <0.1}$	**12.8**$_{\pm 0.2}$	16.1$_{\pm 0.2}$
Shake-Shake (26 2x96d)	Basic	**2.3**$_{\pm <0.1}$	2.7$_{\pm 0.1}$	**15.1**$_{\pm 0.1}$	17.0$_{\pm 0.1}$
Shake-Shake (26 2x96d)	Cutout	**2.0**$_{\pm <0.1}$	2.3$_{\pm 0.1}$	**14.2**$_{\pm 0.2}$	15.7$_{\pm 0.2}$
Shake-Shake (26 2x96d)	AA	**1.6**$_{\pm <0.1}$	1.9$_{\pm 0.1}$	**12.8**$_{\pm 0.1}$	14.1$_{\pm 0.2}$
PyramidNet	Basic	**2.7**$_{\pm 0.1}$	4.0$_{\pm 0.1}$	**14.6**$_{\pm 0.4}$	19.7$_{\pm 0.3}$
PyramidNet	Cutout	**1.9**$_{\pm 0.1}$	2.5$_{\pm 0.1}$	**12.6**$_{\pm 0.2}$	16.4$_{\pm 0.1}$
PyramidNet	AA	**1.6**$_{\pm 0.1}$	1.9$_{\pm 0.1}$	**11.6**$_{\pm 0.1}$	14.6$_{\pm 0.1}$
PyramidNet+ShakeDrop	Basic	**2.1**$_{\pm 0.1}$	2.5$_{\pm 0.1}$	**13.3**$_{\pm 0.2}$	14.5$_{\pm 0.1}$
PyramidNet+ShakeDrop	Cutout	**1.6**$_{\pm <0.1}$	1.9$_{\pm 0.1}$	**11.3**$_{\pm 0.1}$	11.8$_{\pm 0.2}$
PyramidNet+ShakeDrop	AA	**1.4**$_{\pm <0.1}$	1.6$_{\pm <0.1}$	**10.3**$_{\pm 0.1}$	10.6$_{\pm 0.1}$

Table 1: Results for SAM on state-of-the-art models on CIFAR-$\{10, 100\}$ (WRN = WideResNet; AA = AutoAugment; SGD is the standard non-SAM procedure used to train these models).

Sharpness-aware minimization

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.

References I

N. S. Keskar, D. Mudigere, J. Nocedal, M. Smelyanskiy, and P. T. P. Tang, "On large-batch training for deep learning: Generalization gap and sharp minima," in *International Conference on Learning Representations*, 2016.

L. Wu, C. Ma *et al.*, "How sgd selects the global minima in over-parameterized learning: A dynamical stability perspective," *Advances in Neural Information Processing Systems*, vol. 31, 2018.

L. Wu, M. Wang, and W. Su, "The alignment property of sgd noise and how it helps select flat minima: A stability analysis," *Advances in Neural Information Processing Systems*, vol. 35, pp. 4680–4693, 2022.

Z. Zhu, J. Wu, B. Yu, L. Wu, and J. Ma, "The anisotropic noise in stochastic gradient descent: Its behavior of escaping from sharp minima and regularization effects," in *International Conference on Machine Learning*. PMLR, 2019, pp. 7654–7663.

P. Foret, A. Kleiner, H. Mobahi, and B. Neyshabur, "Sharpness-aware minimization for efficiently improving generalization," in *International Conference on Learning Representations*, 2020.