Train longer, generalize better: closing the generalization gap in large batch training of neural networks

Elad Hoffer*, Itay Hubara*, Daniel Soudry
{elad.hoffer, itayhubara, daniel.soudry}@gmail.com

TECHNION
Israel Institute of Technology

Background

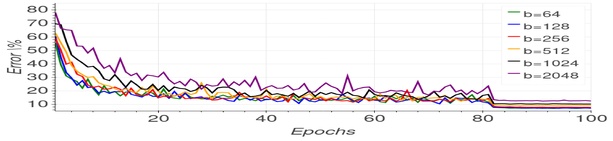
Deep learning models are typically trained using stochastic gradient descent or one of its variants.

- These methods update the weights using their gradient, estimated from a small fraction of the training data.
- Observation: large batch sizes persistently degrade generalization performance - known as the "generalization gap" phenomenon.
- Identifying the origin of this gap and closing it had remained an open problem.

Generalization gap

Previous work by Keskar et al. (2017) studied the performance and properties of models which were trained with relatively large batches and reported the following observations:

- The "generalization gap" seemed to remain even when the models were trained without limits, until the loss function ceased to improve.
- Low generalization was correlated with "sharp" minima
- Briefly noted: Small-batch regimes produce weights that are farther away from the initial point, in comparison with large-batch regime.

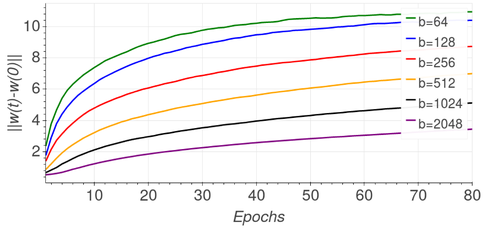


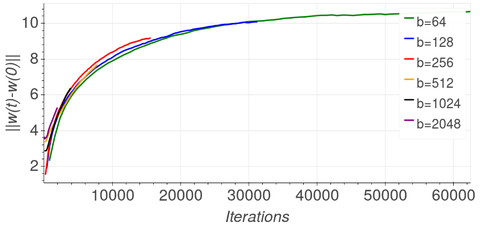
Understanding the gap

We examine $\|\mathbf{w}_t - \mathbf{w}_0\|$ during the initial training phase. We found that the weight distance from initialization point increases logarithmically with the number of training iterations (weight updates):

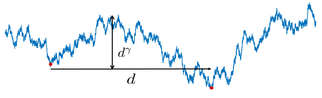
$$\|\mathbf{w}_t - \mathbf{w}_0\| \sim \log t\,. \tag{1}$$

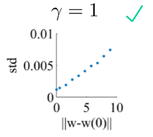
Similar logarithmic graph is observed for all batch sizes, where each graph seems to have a somewhat different slope. This indicates a different diffusion rate for different batch sizes. All models were trained with constant number of epochs, thus smaller batch sizes entail more training iterations resulting with larger weight distance reached at the end of the initial learning phase.

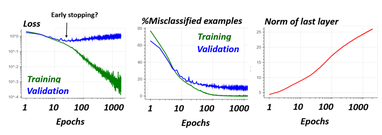




Hypothesis: During the early phase of training, we traverse a noisy landscape with noisy step approximations. Known as "random walk on random potential". we measured the std of the loss and estimated gamma to be equal to 1.
Ultra Slow Diffusion: For the particle to move a distance d - it has to pass a potential barrier of height d^γ



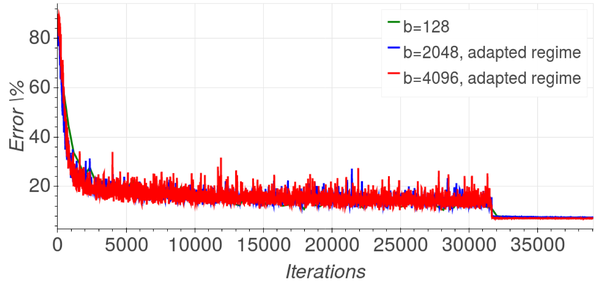




In preliminary results (Soudry et al., 2017), we show that similar behavior occurs even in a simple, unregularized, linearly separable logistic regression problems, optimized using gradient descent. We prove that $w(t)/\|w(t)\|$ converges logarithmically slow to the L_2 maximum margin separator, i.e. , to the solution of the hard margin SVM.

Results - closing the gap

These observations led us to believe that "generalization gap" phenomenon stems from the relatively small number of updates rather than the batch size.
Specifically, using these insights, we adapted the training regime to better suit the usage of large mini-batch. We "stretched" the time-frame of the optimization process, where each time period of E epochs in the original regime, will be transformed to $\frac{|B_L|}{|B_S|}E$ epochs according to the mini-batch size used. This modification ensures that the number of optimization steps taken is identical to those performed in the small batch regime.
As can be seen,, combining this modification with learning rate adjustment completely eliminates the generalization gap observed earlier.



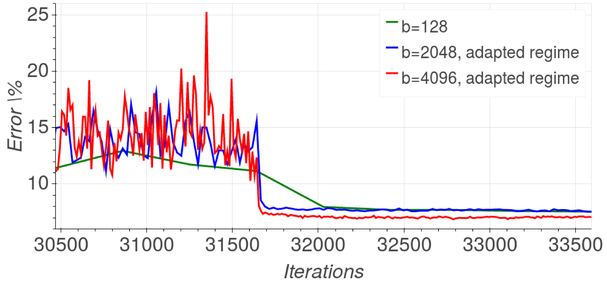


Figure 1: Comparing generalization of large-batch regimes, adapted to match performance of small-batch training.

Network	Dataset	SB	LB	+LR	+GBN	+RA
F1 (Keskar et al., 2017)	MNIST	98.27%	97.05%	97.55%	97.60%	98.53%
C1 (Keskar et al., 2017)	Cifar10	87.80%	83.95%	86.15%	86.4%	88.20%
Resnet44 (He et al., 2016)	Cifar10	92.83%	86.10%	89.30%	90.50%	93.07%
VGG (Simonyan, 2014)	Cifar10	92.30%	84.1%	88.6%	91.50%	93.03%
C3 (Keskar et al., 2017)	Cifar100	61.25%	51.50%	57.38%	57.5%	63.20%
WResnet16-4 (Zagoruyko, 2016)	Cifar100	73.70%	68.15%	69.05%	71.20%	73.57%

Figure 2: Validation accuracy results, SB/LB represent small and large batch respectively. GBN stands for Ghost-BN, and RA stands for regime adaptation

Network	LB size	Dataset	SB	LB	+LR	+GBN	+RA
Alexnet	4096	ImageNet	57.10%	41.23%	53.25%	54.92%	59.5%
Alexnet	8192	ImageNet	57.10%	41.23%	53.25%	53.93%	59.5%

Table 1: ImageNet top-1 results using Alexnet topology (Krizhevsky, 2014), SB/LB represent small and large batch respectively. GBN stands for Ghost-BN, and RA stands for regime adaptation

Followup work

Significant speed-ups are possible: **Accurate, Large Minibatch SGD: Training ImageNet in 1 Hour** P. Goyal et al. (Facebook whitepaper, several weeks later)

- Similar, except learning rate.
- Warm up v.s gradients clipping.
- ×29 times faster than a single worker
- Only works for specific model+dataset?

References

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.

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. In *ICLR*, 2017.

Krizhevsky, A. One weird trick for parallelizing convolutional neural networks. *arXiv preprint arXiv:1404.5997*, 2014.

Simonyan, K. e. a. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

Soudry, D., Hoffer, E., and Srebro, N. The Implicit Bias of Gradient Descent on Separable Data. *ArXiv e-prints*, October 2017.

Zagoruyko, K. Wide residual networks. In *BMVC*, 2016.