Batch Normalization Orthogonalizes Representations in Deep Random Networks

Hadi Daneshmand, Amir Joudaki, Francis Bach

INRIA Paris, ETH Zurich, INRIA-ENS-PSL Paris

NeurIPS@Paris 2021

Batch normalization (BN)

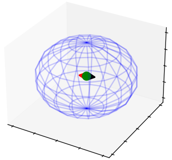
- BN is one of the main building block of modern neural networks[1]
- BN is cited +30K in the literature
- The underlying mechanism of BN is a fundamental open problem in machine learning that has been discussed in various keynotes and plenary talks.
- Even with random weights networks with BN achieves surprisingly good performance[2].

[1] Ioffe, S. & Szegedy, C. *Batch normalization: Accelerating deep network training by reducing internal covariate shift.* in *ICML* (2015).

[2] Frankle, J., Schwab, D. J. & Morcos, A. S. Training batchnorm and only batchnorm: On the expressive power of random features in CNNs. *ICLR* (2021).

The Markov chain of representations

We study a Markov chain of matrices

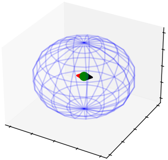


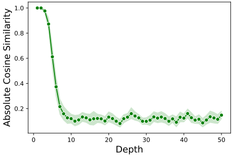
- $BN(M)$ normalizes M row-wise
- Representations:
  $H_{\ell+1} = \left(\frac{1}{\sqrt{width}}\right) BN(W_\ell H_\ell)$
- W_ℓ: $(width \times width)$ with Gaussian elements

The Markov chain of representations

We study a Markov chain of matrices

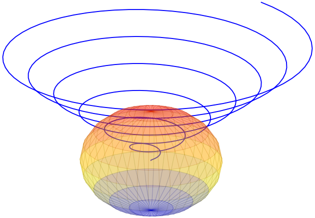
- $BN(M)$ normalizes M row-wise
- Representations:
  $H_{\ell+1} = \left(\frac{1}{\sqrt{width}}\right) BN(W_\ell H_\ell)$
- W_ℓ: $(width \times width)$ with Gaussian elements





Theoretical results

- $\mathbf{E}\left[\text{orthogonality gap}(H_\ell)\right] = \mathcal{O}\left((1-\alpha)^\ell + \frac{\text{batchsize}}{\alpha\sqrt{\text{width}}}\right)$
- $\text{Wasser.}_2(W_\ell H_\ell, \text{Gaussian})^2 = \mathcal{O}\left((1-\alpha)^\ell (\text{batchsize}) + \frac{(\text{batchsize})^2}{\alpha\sqrt{\text{width}}}\right)$

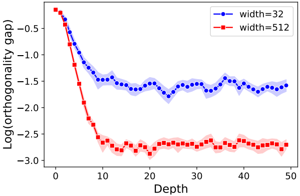


Preliminaries

- Define: orthogonality gap$(H) := \left\| \left(\frac{1}{\|H\|_F^2}\right) H^\top H - \left(\frac{1}{\|I_n\|_F^2}\right) I_n \right\|_F$.
- Assume there exists an absolute positive constant α such that the minimum singular value of H_k is greater than (or equal to) α for all $k = 1, \ldots, \ell$.

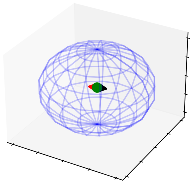
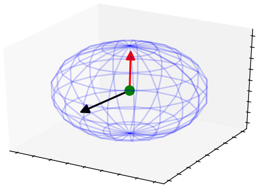
Orthogonalization

- $\mathbf{E}\left[\text{orthogonality gap}(H_\ell)\right] = \mathcal{O}\left((1-\alpha)^\ell + \frac{\text{batchsize}}{\alpha\sqrt{\text{width}}}\right)$

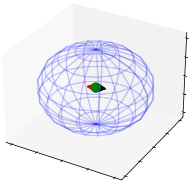
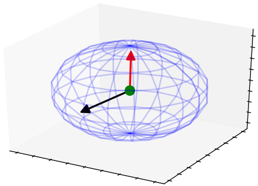


- Recall α is the minimum of smallest singular value of $\{H_1, \ldots, H_\ell\}$.

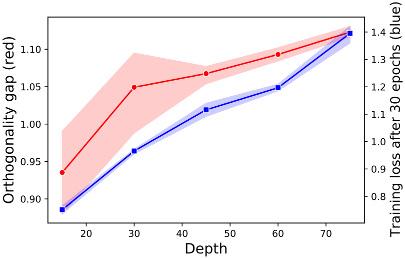
Modern NN vs. historical NN

BN	Without BN
	

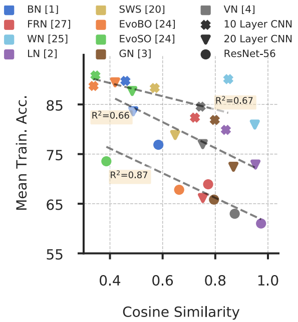
Modern NN vs. historical NN

BN	Without BN
	
$\mathbf{E}\left[\text{orth. gap}(H_\infty)\right] = \mathcal{O}\left(\frac{\text{batch size}}{\alpha\sqrt{\text{width}}}\right)$	$\mathbf{E}\left[\text{orth. gap}(H'_\infty)\right] = \Theta(1)$

The orthogonality influences training



Without normalization

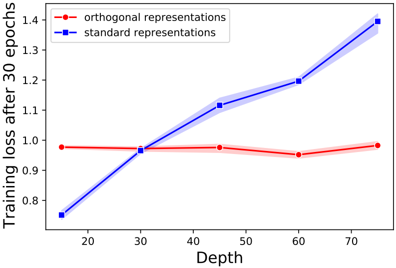


With Normalization[3]

[3] Lubana, E. S., Dick, R. P. & Tanaka, H. Beyond BatchNorm: Towards a General Understanding of Normalization in Deep Learning. *arXiv preprint arXiv:2106.05956* (2021).

Replacing BN with orthogonalization

Saving training time by starting from orthogonal representations



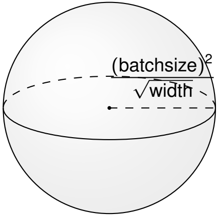
MLPs with ReLU and **without BN** for classifying CIFAR-10

Red: standard initialization with low orthogonality gaps

Blue: novel initialization ensuring orthogonal representations

Gaussian approximation

$$\text{Wasserstein}_2(W_\ell H_\ell, \text{Gaussian})^2 = \mathcal{O}\left((1-\alpha)^\ell (\text{batchsize}) + \frac{(\text{batchsize})^2}{\alpha\sqrt{\text{width}}}\right)$$



History of Gaussian approximation for NNs

∞-Width

1996	A single-layer MLP (Neal).
2015	Going beyond one layer (Hazan and Jaakkola).
2018	Finite-depth MLPs (Matthews et. al. and Lee et. al.) .

History of Gaussian approximation for NNs

∞-Width

1996	A single-layer MLP (Neal).
2015	Going beyond one layer (Hazan and Jaakkola).
2018	Finite-depth MLPs (Matthews et. al. and Lee et. al.) .
2021	Non-asymptotic (in)finite width and depth Linear MLPs.

Applications of the Gaussian approximation

Towards a theory-driven architectural design

- The influence of modern neural components on representations
  - ReLU activations
  - Convolutions layers
  - Normalization layers
  - Residual connections
- Theoretical study of optimization and random representations.
- Design of efficient neural architectures based on theoretical understanding