DeepMind

The Impact of Neural Network Overparameterization on Gradient Confusion and Stochastic Gradient Descent

Soham De (sohamde@google.com)

With: Karthik A Sankararaman, Zheng Xu, Ronny Huang, Tom Goldstein

Paper link: https://arxiv.org/abs/1904.06963

ICML 2020

Stochastic gradient descent (SGD)

Empirically SGD with constant learning rates is very efficient on neural nets

Some recent progress, but behaviour still not fully understood

Stochastic gradient descent (SGD)

Empirically SGD with constant learning rates is very efficient on neural nets

Some recent progress, but behaviour still not fully understood

Existing convergence theory:

- Fast convergence to *neighborhood* of minimizer: depends on variance of gradients
- "Interpolation condition"

Non-Asymptotic Analysis of Stochastic Approximation Algorithms for Machine Learning

Francis Bach
INRIA - Sierra Project-team
Ecole Normale Supérieure, Paris, France
francis.bach@ens.fr

Eric Moulines
LTCI
Telecom ParisTech, Paris, France
eric.moulines@enst.fr

Fast and Faster Convergence of SGD for Over-Parameterized Models (and an Accelerated Perceptron)

Sharan Vaswani[1] **Francis Bach**[2] **Mark Schmidt**[1]
[1]University of British Columbia [2]INRIA, ENS, PSL Research University

Results for neural nets?

Under standard Gaussian initializations:

- **Deeper networks typically harder to train**
  - Innovations: alternate initializations, normalization, residual networks, etc.

How to Start Training:
The Effect of Initialization and Architecture

Boris Hanin
Department of Mathematics
Texas A& M University
College Station, TX, USA
bhanin@math.tamu.edu

David Rolnick
Department of Mathematics
Massachusetts Institute of Technology
Cambridge, MA, USA
drolnick@mit.edu

Results for neural nets?

Under standard Gaussian initializations:

- **Deeper networks typically harder to train**
  - Innovations: alternate initializations, normalization, residual networks, etc.

- **Wider networks typically easier to train**
  - Recent theoretical progress: SGD dynamics simplifies for infinitely wide networks

How to Start Training:
The Effect of Initialization and Architecture

Boris Hanin
Department of Mathematics
Texas A& M University
College Station, TX, USA
bhanin@math.tamu.edu

David Rolnick
Department of Mathematics
Massachusetts Institute of Technology
Cambridge, MA, USA
drolnick@mit.edu

Neural Tangent Kernel:
Convergence and Generalization in Neural Networks

Arthur Jacot
École Polytechnique Fédérale de Lausanne
arthur.jacot@netopera.net

Franck Gabriel
Imperial College London
franckrgabriel@gmail.com

Clément Hongler
École Polytechnique Fédérale de Lausanne
clement.hongler@epfl.ch

Motivating questions

Why is constant learning rate SGD efficient on popular neural net models?

How does the neural network architecture and initialization affect this?

Our approach

Identify a condition: “Gradient Confusion” that affects convergence of SGD

Establish relationships between network depth, layer width and performance

Setting

Empirical risk minimization problem:

$$\min_{\mathbf{w}\in\mathbb{R}^d} F(\mathbf{w}) := \min_{\mathbf{w}\in\mathbb{R}^d} \frac{1}{N}\sum_{i=1}^{N} f_i(\mathbf{w})$$

Objective function for *i*–th example

Setting

Empirical risk minimization problem:

$$\min_{\mathbf{w}\in\mathbb{R}^d} F(\mathbf{w}) := \min_{\mathbf{w}\in\mathbb{R}^d} \frac{1}{N}\sum_{i=1}^{N} \underbrace{f_i(\mathbf{w})}_{\text{Objective function for } i\text{–th example}}$$

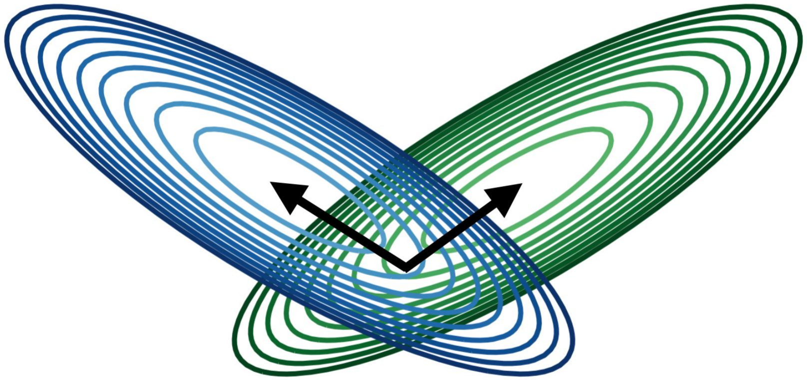
Stochastic gradient descent (SGD):

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \underbrace{\alpha}_{\text{Learning rate}} \underbrace{\nabla \tilde{f}_k(\mathbf{w}_k)}_{\text{Gradient of randomly sampled objective function}}$$

"Gradient Confusion"

A set of objective functions $\{f_i\}_{i\in[N]}$ has gradient confusion $\eta \geq 0$ if:

$$\langle \nabla f_i(\mathbf{w}), \nabla f_j(\mathbf{w}) \rangle \geq -\eta, \ \forall i \neq j \in [N].$$

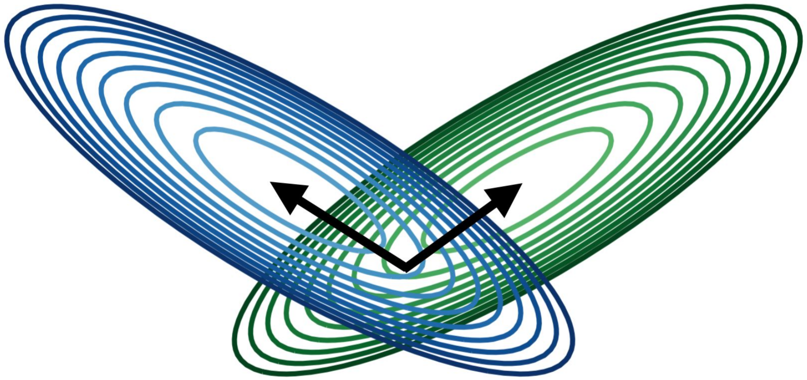


“Gradient Confusion”

A set of objective functions $\{f_i\}_{i \in [N]}$ has gradient confusion $\eta \geq 0$ if:

$$\langle \nabla f_i(\mathbf{w}), \nabla f_j(\mathbf{w}) \rangle \geq -\eta, \ \forall i \neq j \in [N].$$

- **Effect on convergence of SGD?**
- **For which neural network models is it small?**



SGD is fast when gradient confusion is low (example)

Simple linear model example: $f_i(\mathbf{w}) = \mathcal{L}(y_i \mathbf{x}_i^\top \mathbf{w})$

Suppose the data is orthogonal: $\mathbf{x}_i^\top \mathbf{x}_j = 0$

Then, gradients are orthogonal: $\langle \nabla f_i(\mathbf{w}), \nabla f_j(\mathbf{w}) \rangle = 0$

Gradient confusion: $\eta = 0$

Update for example *i* does not affect example *j*

Convergence rate bound

Simplified result:

SGD converges linearly to a *neighborhood* of the minimizer with constant step sizes for *Lipschitz-smooth* and *strongly-convex* functions:

$$F(\mathbf{w}_k) - F(\mathbf{w}^\star) \leq \rho^k \left(F(\mathbf{w}_0) - F(\mathbf{w}^\star)\right) + \frac{\alpha\eta}{1-\rho}$$

where $\alpha < \frac{2}{NL}$, $\rho = 1 - \frac{2\mu}{N}\left(\alpha - \frac{NL\alpha^2}{2}\right)$

(more general results in paper)

Convergence rate bound

Simplified result:

SGD converges linearly to a *neighborhood* of the minimizer with constant step sizes for *Lipschitz-smooth* and *strongly-convex* functions:

$$F(\mathbf{w}_k) - F(\mathbf{w}^\star) \leq \underbrace{\rho^k \left(F(\mathbf{w}_0) - F(\mathbf{w}^\star)\right)}_{\text{decreasing exponentially}} + \underbrace{\frac{\overset{\text{gradient confusion}}{\alpha\eta}}{1-\rho}}_{\text{noise floor}}$$

where $\alpha < \frac{2}{NL}$, $\rho = 1 - \frac{2\mu}{N}\left(\alpha - \frac{NL\alpha^2}{2}\right)$

(more general results in paper)

When gradient confusion is small, SGD has fast convergence

Convergence rate bound

Simplified result:

SGD converges linearly to a *neighborhood* of the minimizer with constant step sizes for *Lipschitz-smooth* and *strongly-convex* functions:

$$F(\mathbf{w}_k) - F(\mathbf{w}^\star) \leq \underbrace{\rho^k \left(F(\mathbf{w}_0) - F(\mathbf{w}^\star)\right)}_{\text{decreasing exponentially}} + \underbrace{\frac{\alpha \overbrace{\eta}^{\text{gradient confusion}}}{1-\rho}}_{\text{noise floor}}$$

where $\alpha < \frac{2}{NL}$, $\rho = 1 - \frac{2\mu}{N}\left(\alpha - \frac{NL\alpha^2}{2}\right)$

(more general results in paper)

When gradient confusion is small, SGD has fast convergence

How likely is it to be small for neural networks?

Effect of neural net architecture at Gaussian initializations

Neural net: $g_{\mathbf{W}}(\mathbf{x}) := \sigma(\mathbf{W}_\beta \sigma(\mathbf{W}_{\beta-1} \ldots \sigma(\mathbf{W}_1 \sigma(\mathbf{W}_0 \mathbf{x}))\ldots))$

ℓ : maximum width of a layer, β : depth of neural network

Activation functions can be ReLUs, tanh or sigmoids

Effect of neural net architecture at Gaussian initializations

Neural net: $g_{\mathbf{W}}(\mathbf{x}) := \sigma(\mathbf{W}_\beta \sigma(\mathbf{W}_{\beta-1} \ldots \sigma(\mathbf{W}_1 \sigma(\mathbf{W}_0 \mathbf{x})) \ldots))$

ℓ : maximum width of a layer, β : depth of neural network

Activation functions can be ReLUs, tanh or sigmoids

Assumptions:

- **Gaussian initializations**: $\mathbf{W}_p \in \mathbb{R}^{\ell_p \times \ell_{p-1}}$ has entries from $\mathcal{N}\left(0, \frac{1}{\kappa \ell_{p-1}}\right)$ for all *p*
- **Random data model**: *x* randomly drawn from surface of *d*-dimensional sphere

κ is typically set to ½ when using ReLUs, and 1 when using tanh non-linearities

Effect of neural net architecture at Gaussian initializations

Simplified result:

Under the above setup, the gradient confusion bound

$$\langle \nabla f_i(\mathbf{w}), \nabla f_j(\mathbf{w}) \rangle \geq -\eta, \;\; \forall i \neq j \in [N].$$

holds with probability at least:

$$1 - \beta \exp(-\Theta(\ell^2)) - N^2 \exp(-\Theta(\ell^2/\beta^5))$$

(more general results in paper)

Effect of neural net architecture at Gaussian initializations

Simplified result:

Under the above setup, the gradient confusion bound

$$\langle \nabla f_i(\mathbf{w}), \nabla f_j(\mathbf{w}) \rangle \geq -\eta, \ \forall i \neq j \in [N].$$

holds with probability at least:

$$1 - \boxed{\beta} \exp(-\Theta(\ell^2)) - N^2 \exp(-\Theta(\ell^2 / \boxed{\beta^5}))$$

network depth

(more general results in paper)

- **Training gets harder with increased depth (higher gradient confusion)**

Effect of neural net architecture at Gaussian initializations

Simplified result:

Under the above setup, the gradient confusion bound

$$\langle \nabla f_i(\mathbf{w}), \nabla f_j(\mathbf{w}) \rangle \geq -\eta, \ \forall i \neq j \in [N].$$

holds with probability at least:

layer width

$$1 - \beta \exp(-\Theta(\boxed{\ell^2})) - N^2 \exp(-\Theta(\boxed{\ell^2}/\beta^5))$$

(more general results in paper)

- **Training gets harder with increased depth (higher gradient confusion)**
- **Training gets easier with increased width (lower gradient confusion)**

Empirically testing the theory: effect of depth

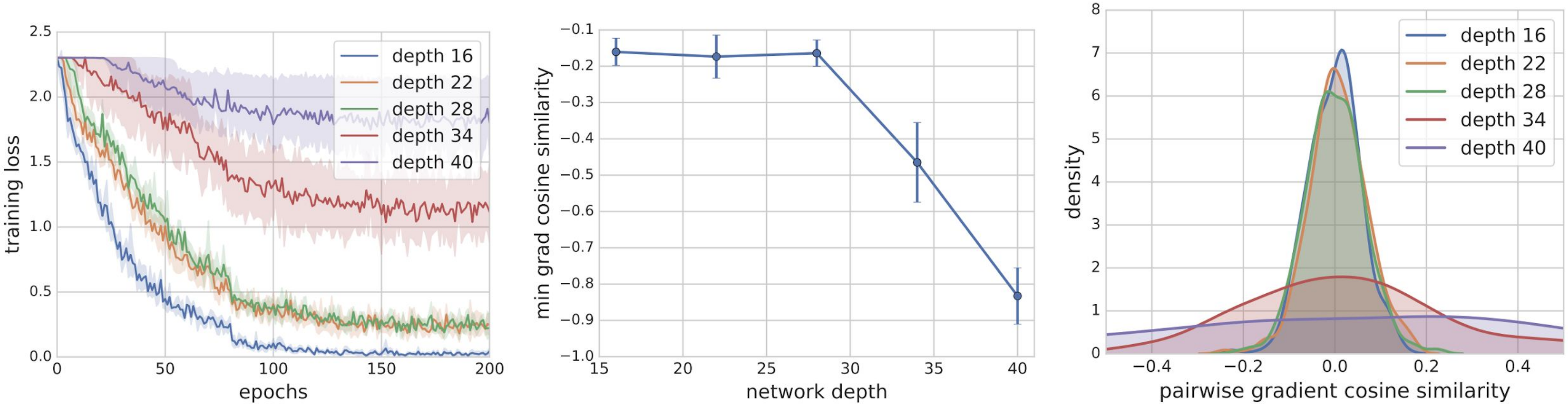


Image Classification on CIFAR-10 with CNNs (more empirical results in the paper)

Increasing depth slows down convergence, and increases gradient confusion

Empirically testing the theory: effect of width

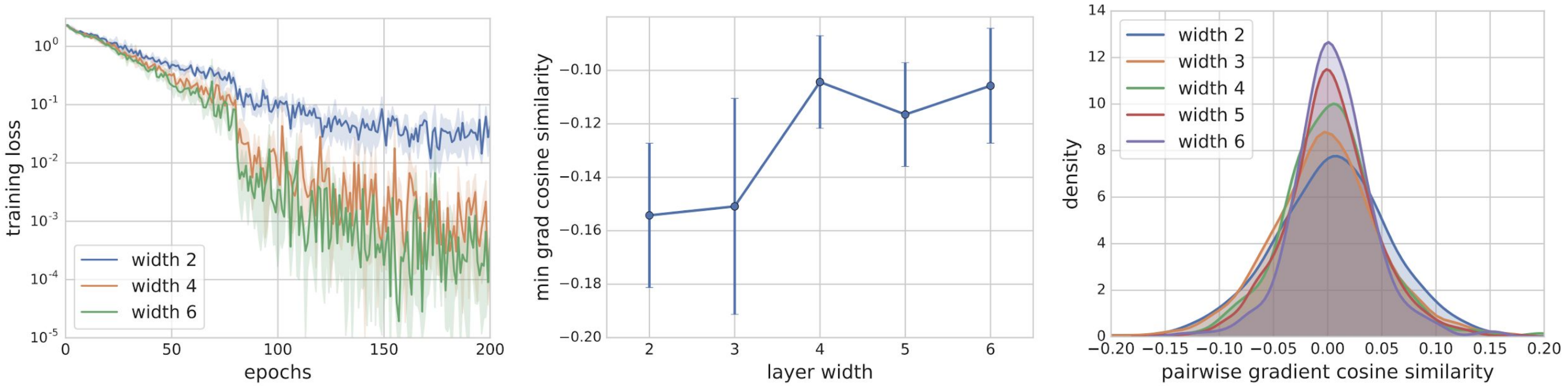


Image Classification on CIFAR-10 with CNNs (more empirical results in the paper)

Increasing width speeds up convergence, and decreases gradient confusion

How can we train very deep networks?

Previous results imply: **increase width with depth**

How do we train very deep networks without increasing the width?

How can we train very deep networks?

Previous results imply: **increase width with depth**

How do we train very deep networks without increasing the width?

- Orthogonal initializations (for linear neural networks)
- Residual networks with batch normalization

Exact solutions to the nonlinear dynamics of learning in deep linear neural networks

Andrew M. Saxe (asaxe@stanford.edu)
Department of Electrical Engineering

James L. McClelland (mcclelland@stanford.edu)
Department of Psychology

Surya Ganguli (sganguli@stanford.edu)
Department of Applied Physics
Stanford University, Stanford, CA 94305 USA

Batch Normalization Biases Residual Blocks Towards the Identity Function in Deep Networks

Soham De
DeepMind, London
sohamde@google.com

Samuel L. Smith
DeepMind, London
slsmith@google.com

Orthogonal init makes early training independent of depth

Informal result

Consider a linear neural network

$$g_{\mathbf{W}}(\mathbf{x}) := \gamma \mathbf{W}_\beta \cdot \mathbf{W}_{\beta-1} \cdot \ldots \cdot \mathbf{W}_1 \cdot \mathbf{x}$$

where recaling parameter $\gamma = \frac{1}{\sqrt{2\beta}}$ and each **W** initialized as an **orthogonal matrix**

Then the gradient confusion bound holds with probability at least

$$\underbrace{1 - N^2 \exp\left(-cd\eta^2\right)}$$

independent of network depth

Effect of batch normalization and skip connections

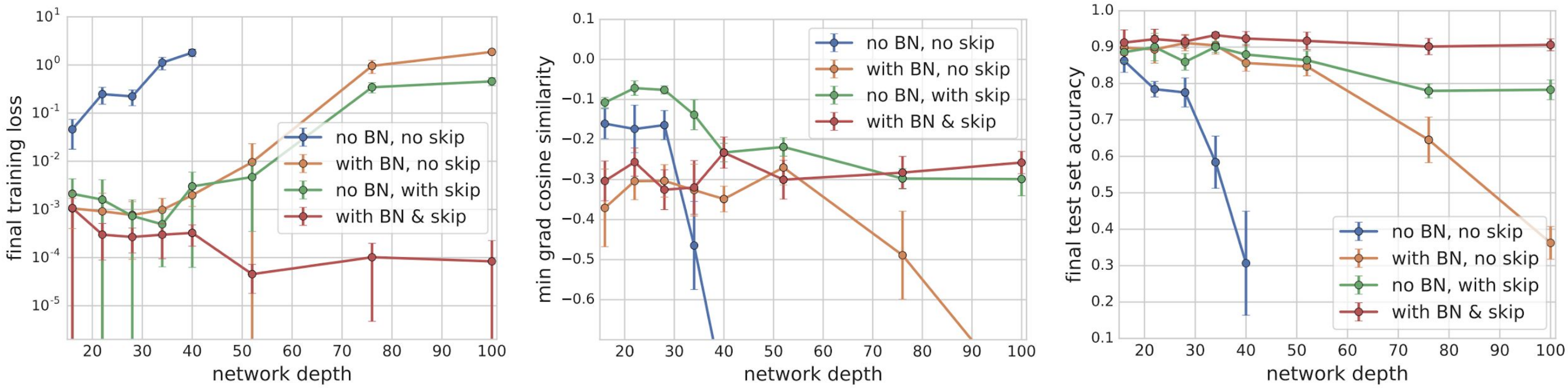


Image Classification on CIFAR-10 with CNNs (more empirical results in the paper)

The combination of batch normalization and skip connections reduces gradient confusion and makes training easier

Summary of key results

We introduce **"Gradient Confusion"** to help analyze trainability of neural networks

1. SGD convergence is faster when gradient confusion is lower

2. Under popular Gaussian initializations:
   - Network depth increases gradient confusion, making training hard
   - Layer width decreases gradient confusion, making training easier

3. How do we train very deep networks without increasing width?
   - Orthogonal initializations make early training independent of depth
   - Using the combination of batch normalization and skip connections

Thank you to my collaborators

Karthik A. Sankararaman

Zheng Xu

W. Ronny Huang

Tom Goldstein

Paper link: https://arxiv.org/abs/1904.06963

Get in touch at sohamde@google.com