DeepMind

On the Generalization Benefit of Noise in Stochastic Gradient Descent

Samuel L. Smith, Erich Elsen and Soham De

ICML 2020

Joint work with

Soham De

Erich Elsen

With thanks to:
Esme Sutherland, James Martens, Yee Whye Teh
Sander Dieleman, Chris Maddison, Karen Simonyan, ...

SGD Crucial to Success of Deep Networks

- Model performance depends strongly on:
    1. Batch size
    2. Learning rate schedule
    3. Number of training epochs

- Many authors have sought to develop rules of thumb to simplify hyper-parameter tuning

- No clear consensus

SGD Crucial to Success of Deep Networks

- Model performance depends strongly on:
    1. Batch size
    2. Learning rate schedule
    3. Number of training epochs

- Many authors have sought to develop rules of thumb to simplify hyper-parameter tuning

- No clear consensus

Accurate, Large Minibatch SGD:
Training ImageNet in 1 Hour

Priya Goyal Piotr Dollár Ross Girshick Pieter Noordhuis
Lukasz Wesolowski Aapo Kyrola Andrew Tulloch Yangqing Jia Kaiming He

The Power of Interpolation: Understanding the Effectiveness of SGD in Modern Over-parametrized Learning

Siyuan Ma, Raef Bassily, Mikhail Belkin
Department of Computer Science and Engineering
The Ohio State University
{ma.588, bassily.1}@osu.edu, mbelkin@cse.ohio-state.edu

DON'T DECAY THE LEARNING RATE, INCREASE THE BATCH SIZE

Samuel L. Smith*, Pieter-Jan Kindermans*, Chris Ying & Quoc V. Le
Google Brain
{slsmith, pikinder, chrisying, qvl}@google.com

Which Algorithmic Choices Matter at Which Batch Sizes? Insights From a Noisy Quadratic Model

Guodong Zhang[1,2,3]*, Lala Li[3], Zachary Nado[3], James Martens[4], Sushant Sachdeva[1], George E. Dahl[3], Christopher J. Shallue[3], Roger Grosse[1,2]
[1]University of Toronto, [2]Vector Institute, [3]Google Research, Brain Team, [4]DeepMind

Measuring the Effects of Data Parallelism on Neural Network Training

Christopher J. Shallue*	SHALLUE@GOOGLE.COM
Jaehoon Lee*†	JAEHLEE@GOOGLE.COM
Joseph Antognini†	JOE.ANTOGNINI@GMAIL.COM
Jascha Sohl-Dickstein	JASCHASD@GOOGLE.COM
Roy Frostig	FROSTIG@GOOGLE.COM
George E. Dahl	GDAHL@GOOGLE.COM

Key questions

Previous papers have studied some of these questions, but often reach contradictory conclusions.

1) How does SGD behave at different batch sizes?

We provide a rigorous empirical study.

2) Do large batch sizes generalize poorly?

3) What is the optimal learning rate for train vs. test performance?

Key questions

Previous papers have studied some of these questions, but often reach contradictory conclusions.

We provide a rigorous empirical study.

1) How does SGD behave at different batch sizes?

Small batch sizes	Large batch sizes
"Noise dominated"	"Curvature dominated"

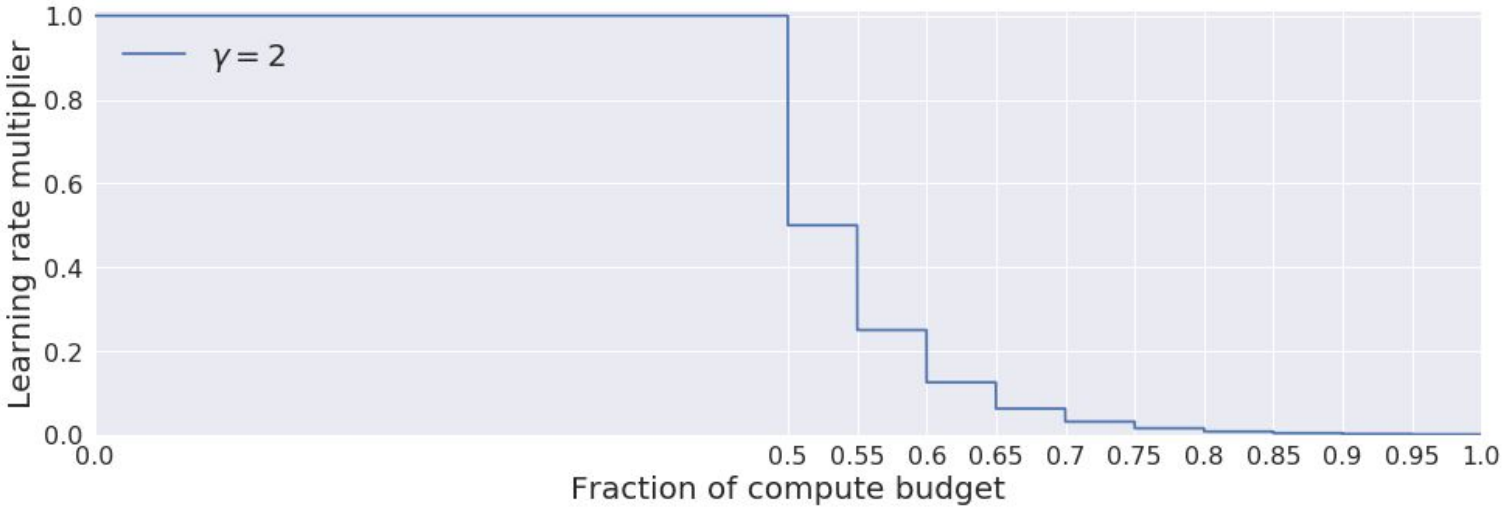
2) Do large batch sizes generalize poorly?

Yes
(may require very large batches)

3) What is the optimal learning rate for train vs. test performance?

Optimal learning rate on train	Optimal learning rate on test
governed by epoch budget	near-independent of epoch budget

To study SGD you must specify a learning rate schedule



Matches or exceeds the original test accuracy
for every architecture we consider

Single hyperparameter -> initial learning rate ε

To study SGD you must specify the compute budget

Constant epoch budget

Compute cost independent of batch size, but number of updates inversely proportional to batch size.

Unlimited compute budget

Train for as long as needed to minimize the training loss or maximize the test accuracy.

Constant step budget

Compute cost proportional to batch size, but number of updates independent of batch size.

Journal of Machine Learning Research 20 (2019) 1-49 Submitted 11/18; Published 7/19

Measuring the Effects of Data Parallelism on Neural Network Training

Christopher J. Shallue* SHALLUE@GOOGLE.COM
Jaehoon Lee*† JAEHLEE@GOOGLE.COM
Joseph Antognini† JOE.ANTOGNINI@GMAIL.COM
Jascha Sohl-Dickstein JASCHASD@GOOGLE.COM
Roy Frostig FROSTIG@GOOGLE.COM
George E. Dahl GDAHL@GOOGLE.COM

Google Brain
1600 Amphiteatre Parkway
Mountain View, CA, 94043, USA

To study SGD you must specify the compute budget

Constant epoch budget

Compute cost independent of batch size, but number of updates inversely proportional to batch size.

Confirm existence of two SGD regimes

Unlimited compute budget

Train for as long as needed to minimize the training loss or maximize the test accuracy.

Verify benefits of large learning rates

Constant step budget

Compute cost proportional to batch size, but number of updates independent of batch size.

Confirm small minibatches generalize better

Journal of Machine Learning Research 20 (2019) 1-49 Submitted 11/18; Published 7/19

Measuring the Effects of Data Parallelism on Neural Network Training

Christopher J. Shallue* SHALLUE@GOOGLE.COM
Jaehoon Lee*† JAEHLEE@GOOGLE.COM
Joseph Antognini† JOE.ANTOGNINI@GMAIL.COM
Jascha Sohl-Dickstein JASCHASD@GOOGLE.COM
Roy Frostig FROSTIG@GOOGLE.COM
George E. Dahl GDAHL@GOOGLE.COM

Google Brain
1600 Amphiteatre Parkway
Mountain View, CA, 94043, USA

Sweeping batch size at constant epoch budget

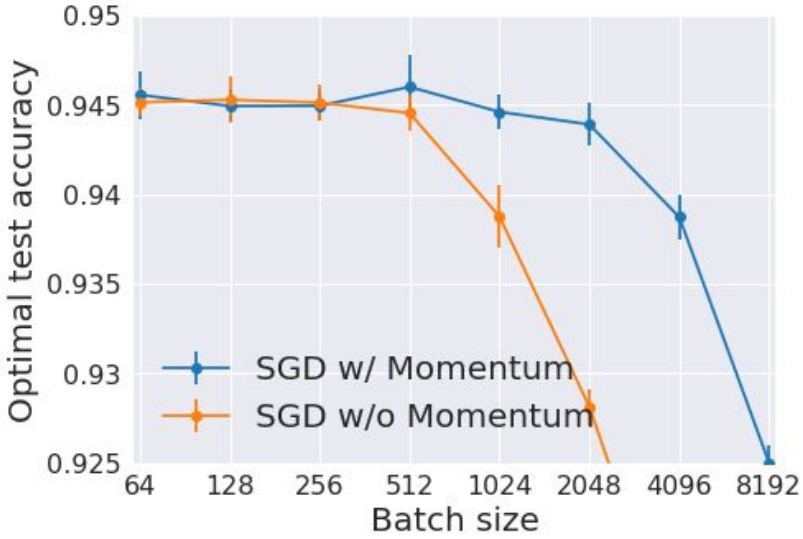
- Four popular benchmarks:
  - 16-4 Wide-ResNet on CIFAR-10 (w/ and w/o batch normalization)
  - Fully Connected Auto-Encoder on MNIST
  - LSTM language model on Penn-TreeBank
  - ResNet-50 on ImageNet

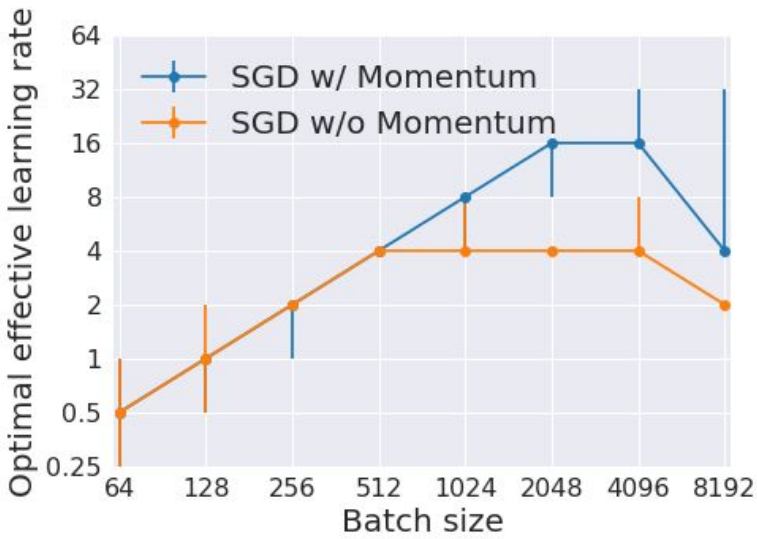
Model	Ghost Batch Size
Wide-ResNet	64
ResNet-50	256

- Grid search over learning rates at all batch sizes
- Similar behaviour in all cases, we pick one example for brevity

Wide-ResNet w/ Batch Normalization (200 epochs)

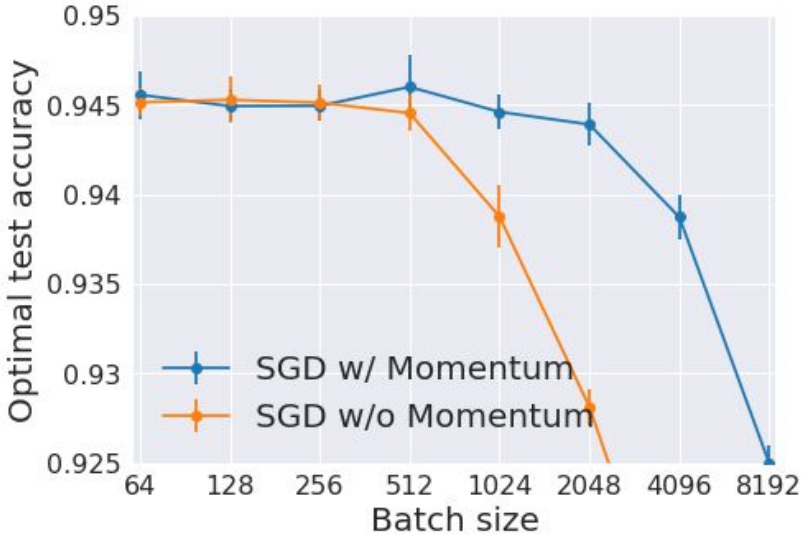
$$\epsilon_{eff} = \frac{\epsilon}{1-m}$$

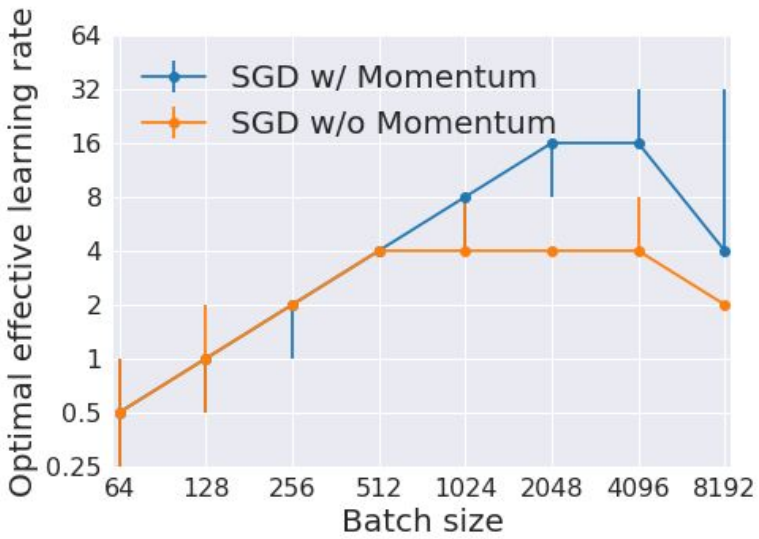




Wide-ResNet w/ Batch Normalization (200 epochs)

$$\epsilon_{eff} = \frac{\epsilon}{1-m}$$





Noise dominated (B < 512):

- Test accuracy independent of batch size
- Both methods identical
- Learning rate proportional to batch size

Curvature dominated (B > 512):

- Test accuracy falls as batch size increases
- Momentum outperforms SGD
- Learning rate independent of batch size

The Two Regimes of SGD

Learning rate
ε

Batch size
B

Training set size N

$$\omega_{i+1} = \omega_i - \frac{\epsilon_i}{B} \sum_{j=1}^{B} \frac{dL(y_j, x_j, \omega_i)}{d\omega}$$

Dynamics governed by error in gradient estimate

"**Noise dominated**"

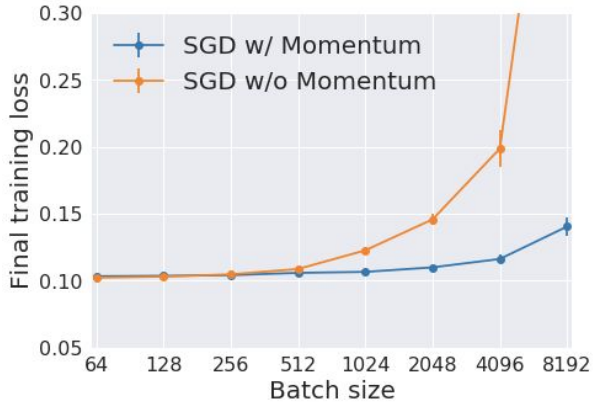
Dynamics governed by shape of loss landscape

"**Curvature dominated**"

Transition surprisingly sharp in practice

Sweeping batch size at constant step budget

- Previous section demonstrated that the optimal test accuracy was higher for smaller batches (under a constant epoch budget)
- However, this is primarily because large batches were unable to minimize the training loss
- To establish whether small batches also help generalization, we consider a constant step budget

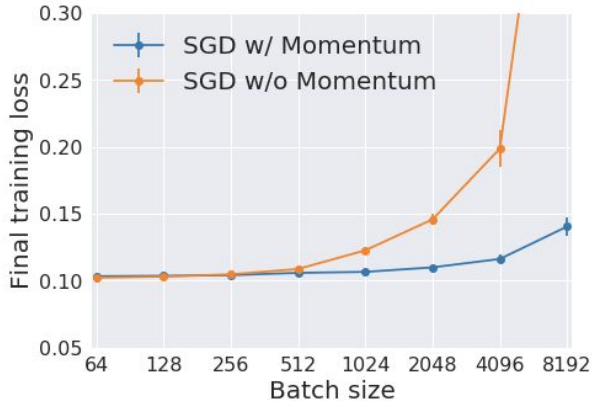


(Training loss rises with batch size under constant epoch budget)

Sweeping batch size at constant step budget

From now on, only consider SGD w/ Momentum

- Previous section demonstrated that the optimal test accuracy was higher for smaller batches (under a constant epoch budget)
- However, this is primarily because large batches were unable to minimize the training loss
- To establish whether small batches also help generalization, we consider a constant step budget



(Training loss rises with batch size under constant epoch budget)

Wide-ResNet w/ Batch Normalization (9765 steps)

Batch size	Optimal test accuracy (%)	Optimal effective learning rate
256	93.6 ± 0.1	2^2 (2^1 to 2^2)
512	94.2 ± 0.1	2^2 (2^2 to 2^3)
1024	94.5 ± 0.1	2^3 (2^3 to 2^3)
2048	94.9 ± 0.1	2^3 (2^3 to 2^3)
4096	94.7 ± 0.1	2^4 (2^4 to 2^5)
8192	94.6 ± 0.1	2^2 (2^2 to 2^2)
16384	92.5 ± 0.6	2^5 (2^4 to 2^5)
32768	89.9 ± 0.7	2^5 (2^0 to 2^5)

Test accuracy falls for large batches, even under a constant step budget!

Learning rate increases sublinearly with batch size

Wide-ResNet w/ Batch Normalization (9765 steps)

Batch size	Optimal test accuracy (%)	Optimal effective learning rate
256	93.6 ± 0.1	2^2 (2^1 to 2^2)
512	94.2 ± 0.1	2^2 (2^2 to 2^3)
1024	94.5 ± 0.1	2^3 (2^3 to 2^3)
2048	94.9 ± 0.1	2^3 (2^3 to 2^3)
4096	94.7 ± 0.1	2^4 (2^4 to 2^5)
8192	94.6 ± 0.1	2^2 (2^2 to 2^2)
16384	92.5 ± 0.6	2^5 (2^4 to 2^5)
32768	89.9 ± 0.7	2^5 (2^0 to 2^5)

Test accuracy falls for large batches, even under a constant step budget!

Learning rate increases sublinearly with batch size

Conclusion: SGD noise can help generalization

(likely you could replace noise with explicit regularization)

Sweeping epoch budget at fixed batch size

- Thus far, we have studied how the test accuracy depends on the batch size under fixed compute budgets

- We now fix the batch size, and study how the test accuracy and optimal learning rate change as the compute budget increases

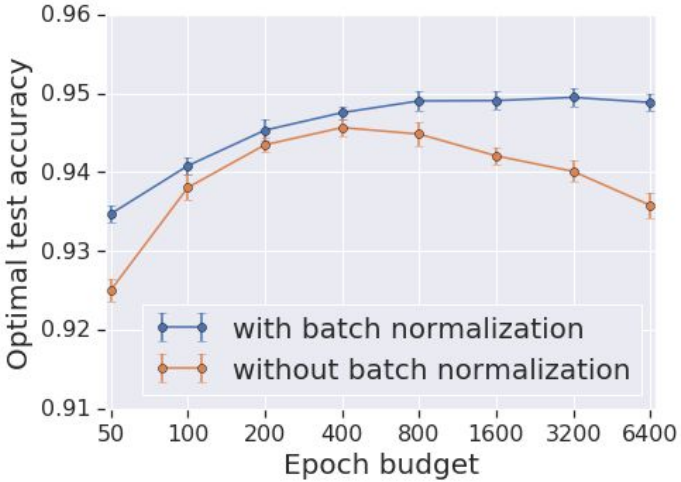
Sweeping epoch budget at fixed batch size

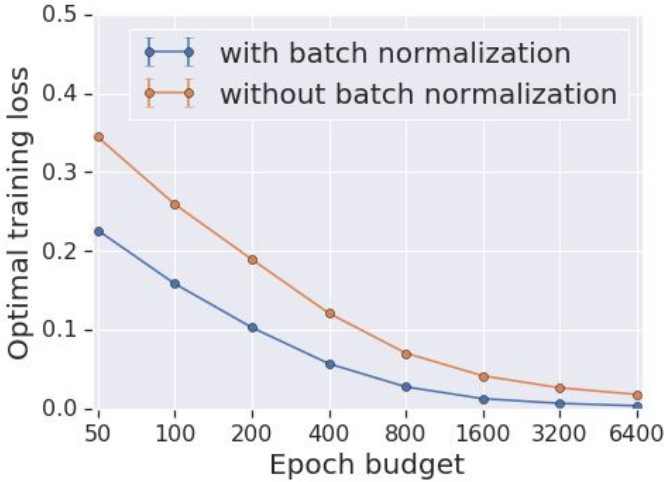
- Thus far, we have studied how the test accuracy depends on the batch size under fixed compute budgets

- We now fix the batch size, and study how the test accuracy and optimal learning rate change as the compute budget increases

- Independently measure:
    - Learning rate which maximizes test accuracy
    - Learning rate which minimizes training loss

Wide-ResNet on CIFAR-10 at batch size 64:



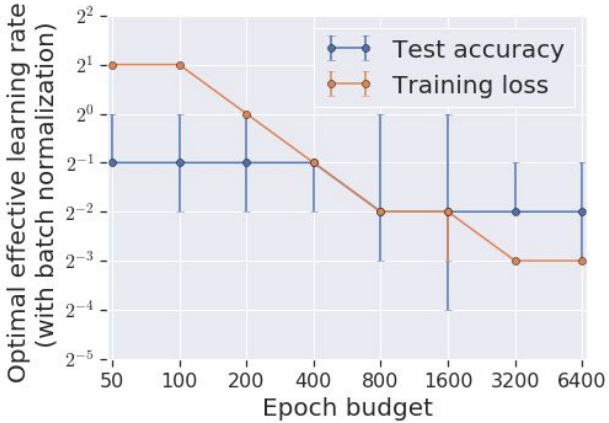


As expected, test accuracy saturates after finite epoch budget

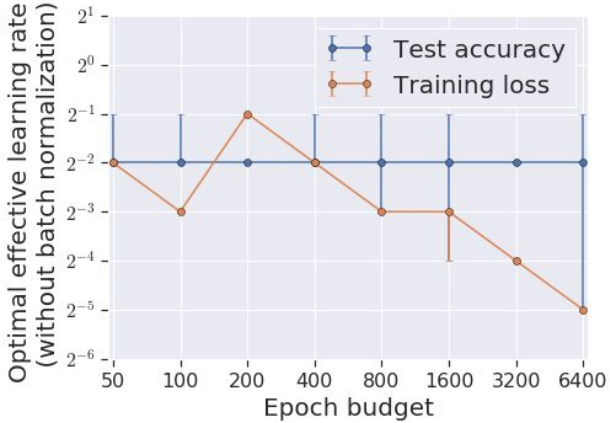
w/out batch normalization uses "SkipInit".
See: https://arxiv.org/pdf/2002.10444.pdf

Wide-ResNet on CIFAR-10 at batch size 64:

w/ Batch Normalization



w/o Batch Normalization



Training set:
Optimal learning rate decays as epoch budget increases

Test set:
Optimal learning rate almost independent of epoch budget

Supports notion that large learning rates generalize well early in training

Why is SGD so hard to beat?

Stochastic optimization has two big (fr)enemies:

1) Gradient noise
2) Curvature (maximum stable learning rate)

Under constant epoch budgets, we can ignore curvature by reducing the batch size

Why is SGD so hard to beat?

Stochastic optimization has two big (fr)enemies:

1) Gradient noise
2) Curvature (maximum stable learning rate)

Under constant epoch budgets, we can ignore curvature by reducing the batch size

Methods designed for curvature probably only help under constant step budgets/large batch training

1) Momentum
2) Adam
3) KFAC/Natural Gradient Descent

The Marginal Value of Adaptive Gradient Methods in Machine Learning

Ashia C. Wilson[♯], Rebecca Roelofs[♯], Mitchell Stern[♯], Nathan Srebro[†], and Benjamin Recht[♯]
{ashia,roelofs,mitchell}@berkeley.edu, nati@ttic.edu, brecht@berkeley.edu
[♯]University of California, Berkeley
[†]Toyota Technological Institute at Chicago

Which Algorithmic Choices Matter at Which Batch Sizes? Insights From a Noisy Quadratic Model

Guodong Zhang[1,2,3]*, Lala Li[3], Zachary Nado[3], James Martens[4],
Sushant Sachdeva[1], George E. Dahl[3], Christopher J. Shallue[3], Roger Grosse[1,2]
[1]University of Toronto, [2]Vector Institute, [3]Google Research, Brain Team, [4]DeepMind

Why is SGD so hard to beat?

Stochastic optimization has two big (fr)enemies:

1) Gradient noise
2) Curvature (maximum stable learning rate)

Under constant epoch budgets, we can ignore curvature by reducing the batch size

Methods designed for curvature probably only help under constant step budgets/large batch training

1) Momentum
2) Adam
3) KFAC/Natural Gradient Descent

There are methods designed to tackle gradient noise (eg. SVRG), but currently these do not work well on neural networks
(need to preserve generalization benefit of SGD?)

The Marginal Value of Adaptive Gradient Methods in Machine Learning

Ashia C. Wilson[♯], Rebecca Roelofs[♯], Mitchell Stern[♯], Nathan Srebro[†], and Benjamin Recht[♯]
{ashia,roelofs,mitchell}@berkeley.edu, nati@ttic.edu, brecht@berkeley.edu
[♯]University of California, Berkeley
[†]Toyota Technological Institute at Chicago

Which Algorithmic Choices Matter at Which Batch Sizes? Insights From a Noisy Quadratic Model

Guodong Zhang[1,2,3]*, Lala Li[3], Zachary Nado[3], James Martens[4],
Sushant Sachdeva[1], George E. Dahl[3], Christopher J. Shallue[3], Roger Grosse[1,2]
[1]University of Toronto, [2]Vector Institute, [3]Google Research, Brain Team, [4]DeepMind

Conclusions

Thank you for listening!

1) How does SGD behave at different batch sizes?

Small batch sizes	Large batch sizes
"Noise dominated"	"Curvature dominated"

2) Do large batch sizes generalize poorly?

Yes
(may require very large batches)

3) What is the optimal learning rate for train vs. test performance?

Optimal learning rate on train	Optimal learning rate on test
governed by epoch budget	near-independent of epoch budget