

One Initialization to Rule Them All?

Stefan Wijnja, Ellis Wierstra, Thomas Hamburger

January 28, 2020

Neural networks are very cool

- ▶ TODO: Images of cool applications.

But they can be very demanding

- ▶ TODO: Numbers for huge networks and how long it takes to train them.

The Lottery Ticket Hypothesis

A randomly initialized, dense neural-network contains a subnetwork that is initialized such that – when trained in isolation – it can match the test accuracy of the original network after training for at most the same number of iterations.

(Frankle and Carbin 2019)

The Lottery Ticket Hypothesis (redux)

A big net contains a small net that can match the big net.

Finding winning tickets by pruning

- ▶ Pruning
- ▶ Magnitude-based pruning
- ▶ Iterative magnitude-based pruning

Status quo on pruning

TODO

- ▶ (Frankle and Carbin 2019)
- ▶ (Morcos et al. 2019)

Our question

How do initialization algorithms compare when looking for winning tickets?

Method

Plan

1. Pick dataset.
2. Pick model.
3. Pick initialization methods.
4. **Write the code.**
5. **Do lots of experiments.**
6. Answer the question.

Method

Dataset: MNIST



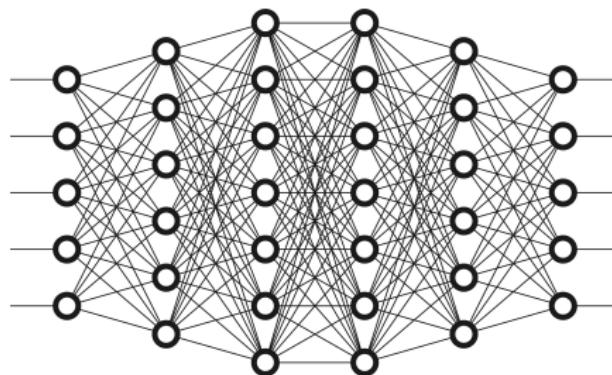
A 4x10 grid of handwritten digits. The digits are arranged in four rows. The first row contains ten '0's. The second row contains ten '1's. The third row contains ten '2's. The fourth row contains ten '3's. The digits are written in a cursive style and are slightly irregular.

0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1	1
2	2	2	2	2	2	2	2	2	2
3	3	3	3	3	3	3	3	3	3

- ▶ Handwritten digits
- ▶ 28×28 pixels
- ▶ Training set: 60k
- ▶ Test set: 10k

Method

Model: LeNet (Lecun et al. 1998)



- ▶ Fully connected
- ▶ Two hidden layers: 300 & 100 neurons → 266k weights.
- ▶ Leaky ReLU (negative slope: 0.05)

Method

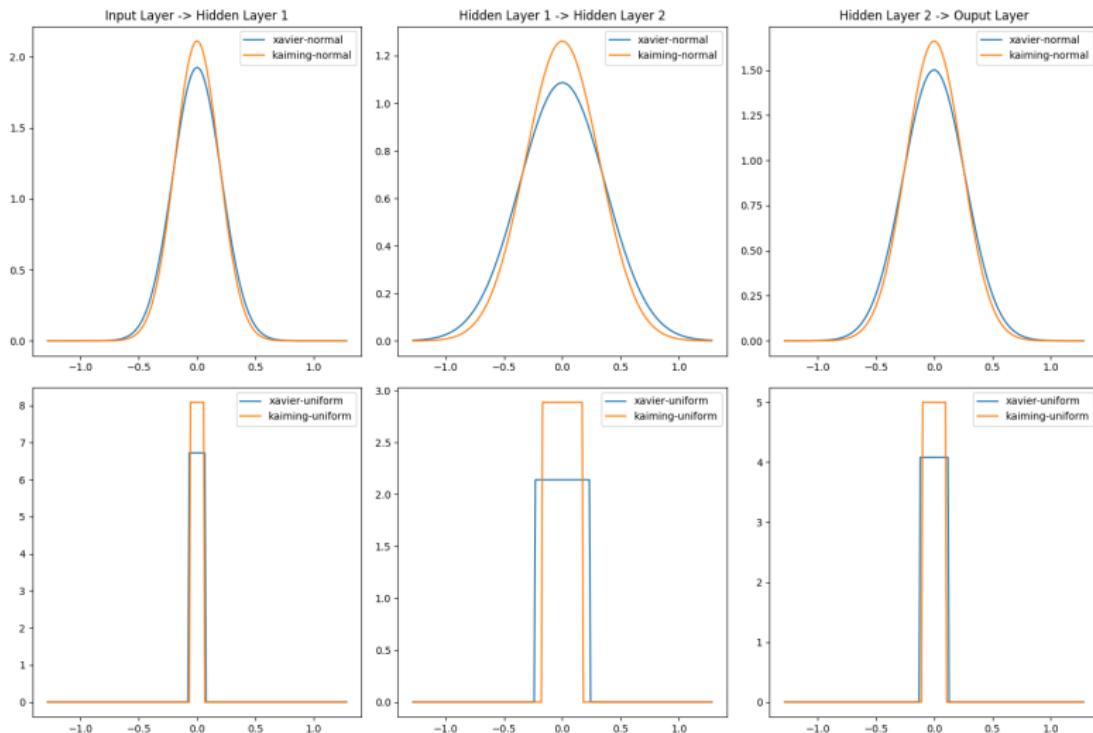
Initialization Methods

	Xavier	Kaiming
$\mathcal{U}(-a, a)$	$\sqrt{\frac{6}{\text{fan_in}+\text{fan_out}}}$	$\sqrt{\frac{3}{\text{fan_mode}}}$
$\mathcal{N}(0, \text{var})$	$\sqrt{\frac{2}{\text{fan_in}+\text{fan_out}}}$	$\frac{1}{\sqrt{\text{fan_mode}}}$

- ▶ $\text{fan_in}/\text{fan_out}$: number of inputs to/outputs from a neuron.
 fan_mode : fan_in or fan_out .

Method

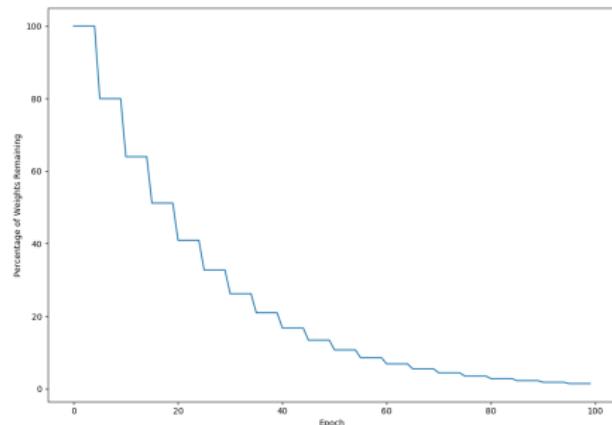
Initialization Methods' Probability Density Functions



Method

Training

1. Initialize network and save weights.
2. Train for 5 epochs.
3. Trim smallest 20% of the weights.
4. Reset the weights to the saved ones.
5. GO TO 2



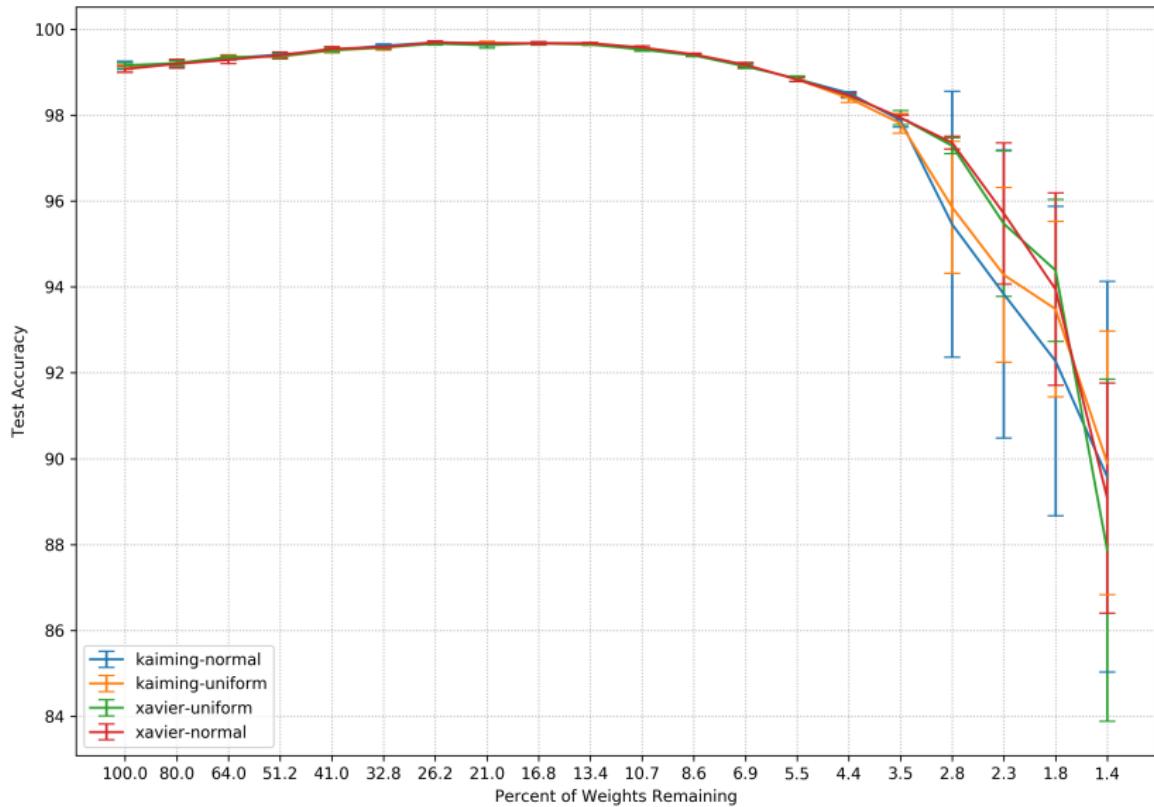
Method

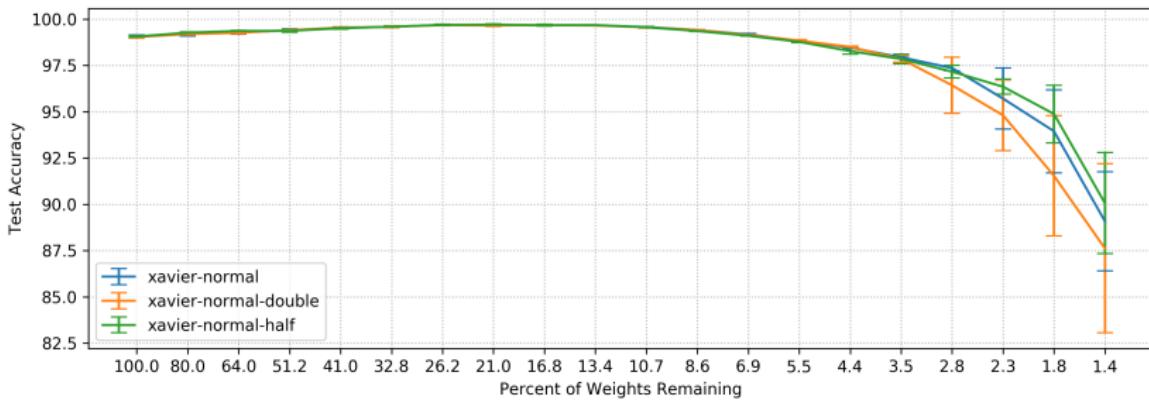
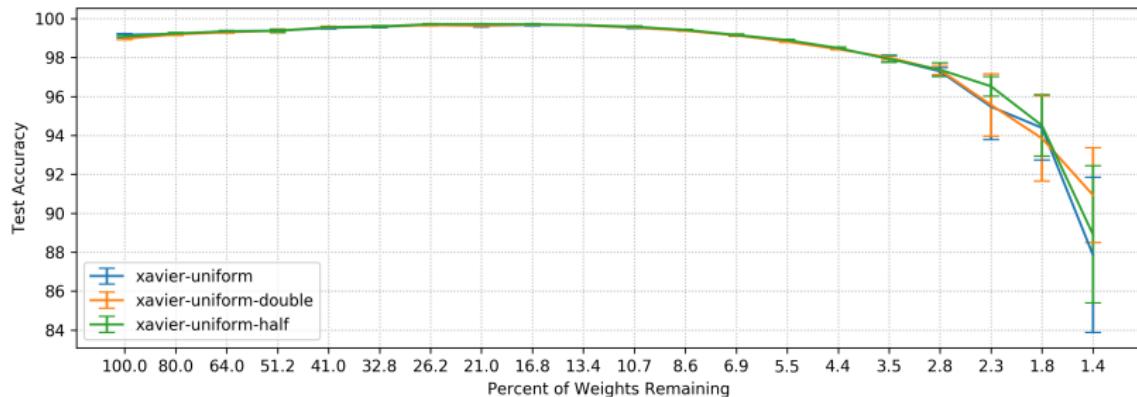
Testing

- ▶ Save loss and accuracy on the training and testing set after every epoch.



Results





Question

How do initialization algorithms compare when looking for winning tickets?

Answer*

Xavier > Kaiming

Uniform \approx Normal

Narrow > wide

Further Research

- ▶ Grid search other datasets/models/pruning settings.
- ▶ Are pruning method and initialization method connected?

Questions?

- ▶ Thanks
 - ▶  Andrei Apostol
 - ▶ Putri van der Linden and Lieuwe Rekker
 - ▶ BrainCreators
- ▶ Stats
 - ▶ ~599 lines of code
 - ▶ 122 training runs
- ▶ Frameworks
 - ▶ PyTorch
- ▶ Can I reproduce your results?
 - ▶ Please do. Email s.wijnja@me.com



References

- Frankle, Jonathan, and Michael Carbin. 2019. “The Lottery Ticket Hypothesis: Finding Sparse, Trainable Neural Networks.”
- Lecun, Y, L Bottou, Y Bengio, and P Haffner. 1998. “Gradient-Based Learning Applied to Document Recognition.” *Proceedings of the IEEE* 86 (11): 2278, 2324.
- Morcos, Ari S., Haonan Yu, Michela Paganini, and Yuandong Tian. 2019. “One Ticket to Win Them All: Generalizing Lottery Ticket Initializations Across Datasets and Optimizers.”