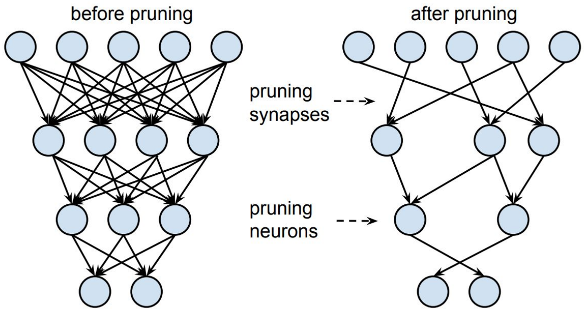
A signal propagation perspective for pruning neural networks at initialization

Namhoon Lee[1], Thalaiyasingam Ajanthan[2], Stephen Gould[2], Philip Torr[1]

[1]University of Oxford, [2]Australian National University

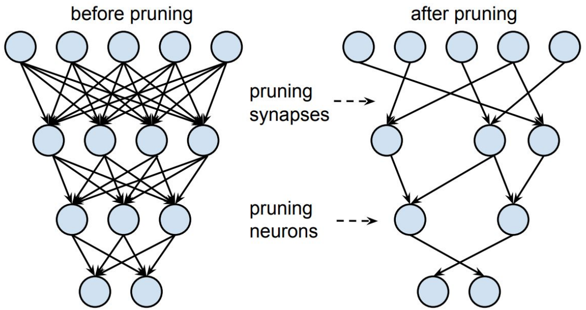
ICLR 2020 *Spotlight presentation*

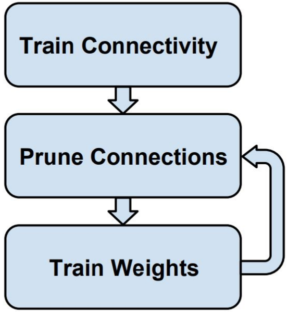
Motivation



Han et al. 2015

Motivation





Han et al. 2015

A typical pruning approach requires training steps

(Han *et al.* 2015, Liu *et al.* 2019).

Motivation

Published as a conference paper at ICLR 2019

SNIP: SINGLE-SHOT NETWORK PRUNING BASED ON CONNECTION SENSITIVITY

Namhoon Lee, Thalaiyasingam Ajanthan & Philip H. S. Torr
University of Oxford
{namhoon,ajanthan,phst}@robots.ox.ac.uk

ABSTRACT

Pruning large neural networks while maintaining their performance is often desirable due to the reduced space and time complexity. In existing methods, pruning is done within an iterative optimization procedure with either heuristically designed pruning schedules or additional hyperparameters, undermining their utility. In this work, we present a new approach that prunes a given network once at initialization prior to training. To achieve this, we introduce a saliency criterion based on connection sensitivity that identifies structurally important connections in the network for the given task. This eliminates the need for both pretraining and the complex pruning schedule while making it robust to architecture variations. After pruning, the sparse network is trained in the standard way. Our method obtains extremely sparse networks with virtually the same accuracy as the reference network on the MNIST, CIFAR-10, and Tiny-ImageNet classification tasks and is broadly applicable to various architectures including convolutional, residual and recurrent networks. Unlike existing methods, our approach enables us to demonstrate that the retained connections are indeed relevant to the given task.

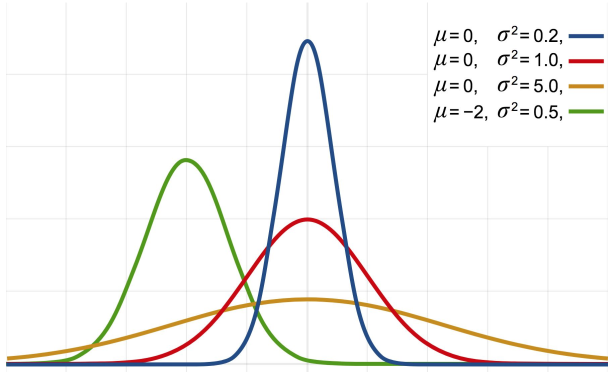
A typical pruning approach requires training steps

(Han *et al.* 2015, Liu *et al.* 2019).

Pruning can be done efficiently at initialization prior to training based on connection sensitivity

(Lee *et al.*, 2019).

Motivation



A typical pruning approach requires training steps

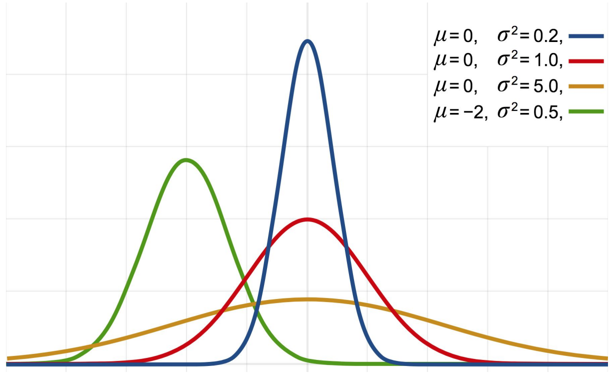
(Han *et al.* 2015, Liu *et al.* 2019).

Pruning can be done efficiently at initialization prior to training based on connection sensitivity

(Lee *et al.*, 2019).

The initial random weights are drawn from appropriately scaled Gaussians (Glorot & Bengio, 2010).

Motivation



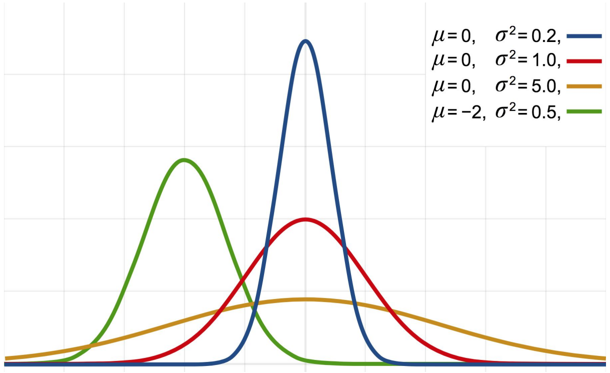
A typical pruning approach requires training steps (Han *et al.* 2015, Liu *et al.* 2019).

Pruning can be done efficiently at initialization prior to training based on connection sensitivity (Lee *et al.*, 2019).

The initial random weights are drawn from appropriately scaled Gaussians (Glorot & Bengio, 2010).

It remains unclear exactly why pruning at initialization is effective.

Motivation



A typical pruning approach requires training steps (Han *et al.* 2015, Liu *et al.* 2019).

Pruning can be done efficiently at initialization prior to training based on connection sensitivity (Lee *et al.*, 2019).

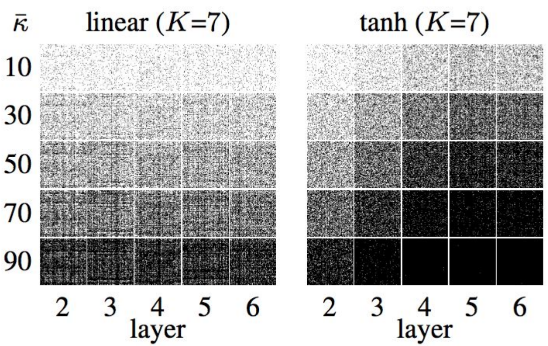
The initial random weights are drawn from appropriately scaled Gaussians (Glorot & Bengio, 2010).

It remains unclear exactly why pruning at initialization is effective.

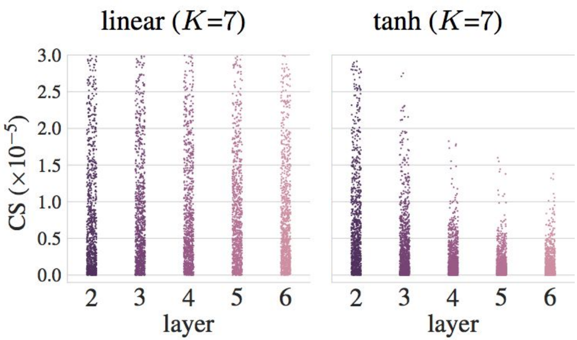
Our take ⇒ ***Signal Propagation Perspective***.

Initialization & connection sensitivity

Sparsity pattern

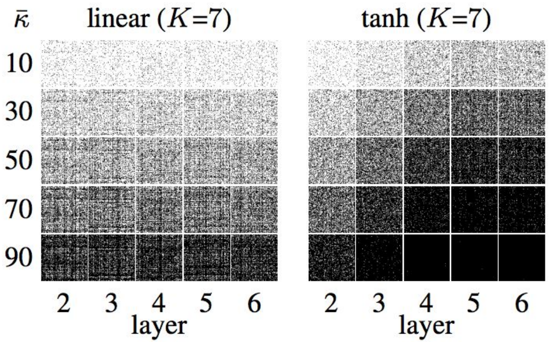


Sensitivity scores

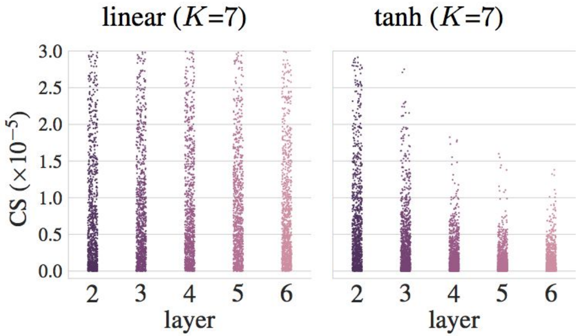


Initialization & connection sensitivity

Sparsity pattern



Sensitivity scores

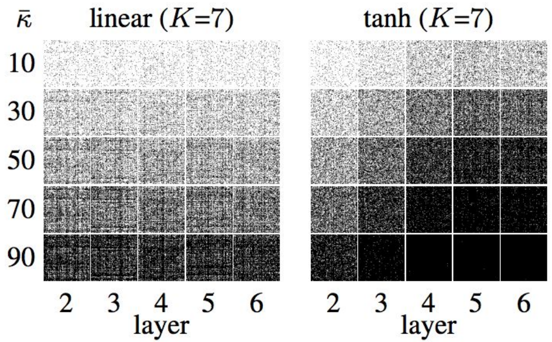


(Linear) uniformly pruned throughout the network.

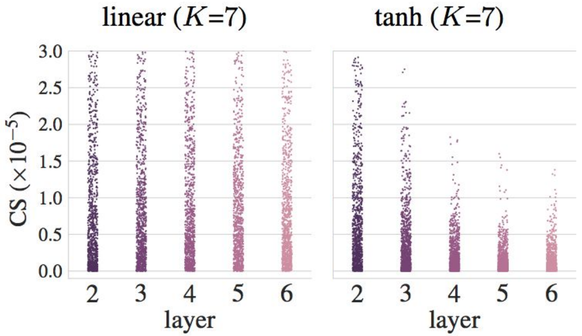
→ *learning capability secured*.

Initialization & connection sensitivity

Sparsity pattern



Sensitivity scores



(Linear) uniformly pruned throughout the network.

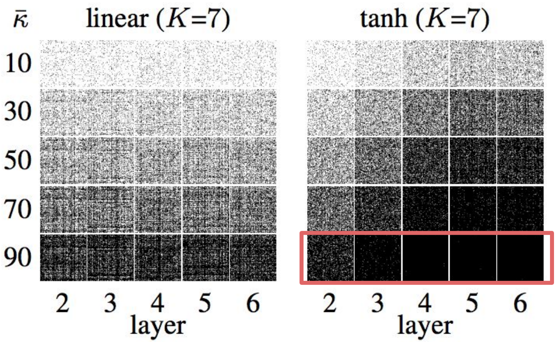
→ *learning capability secured*.

(tanh) more parameters pruned in the later layers.

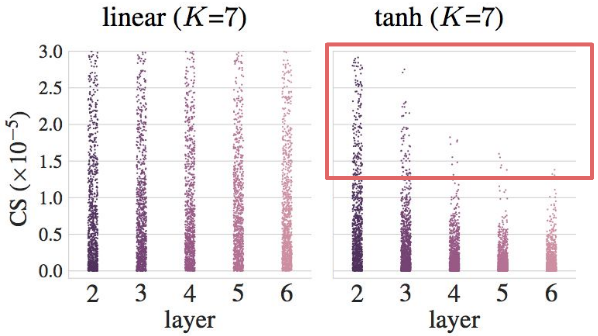
→ *critical for high sparsity pruning*.

Initialization & connection sensitivity

Sparsity pattern



Sensitivity scores

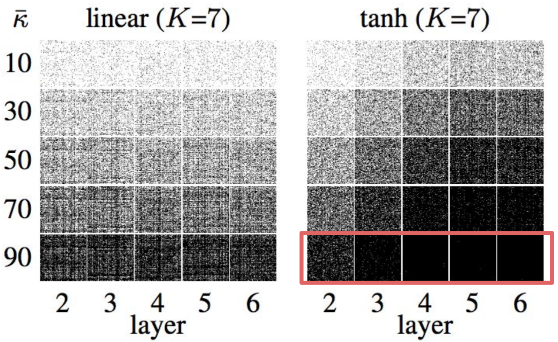


(Linear) uniformly pruned throughout the network.
$\longrightarrow$ *learning capability secured*.

(tanh) more parameters pruned in the later layers.
$\longrightarrow$ *critical for high sparsity pruning*.

Initialization & connection sensitivity

Sparsity pattern



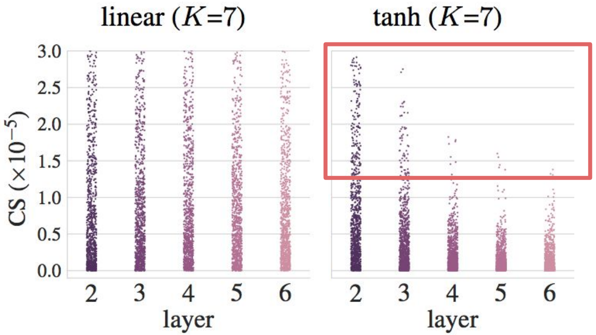
(Linear) uniformly pruned throughout the network.

→ *learning capability secured*.

(tanh) more parameters pruned in the later layers.

→ *critical for high sparsity pruning*.

Sensitivity scores

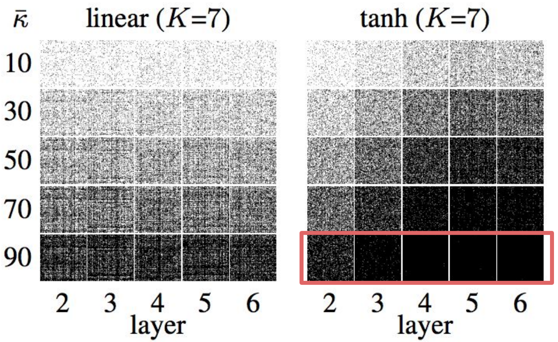


CS scores decrease towards the later layers.

→ Choosing top salient parameters globally results in *a network, in which parameters are distributed non-uniformly and sparsely towards the end*.

Initialization & connection sensitivity

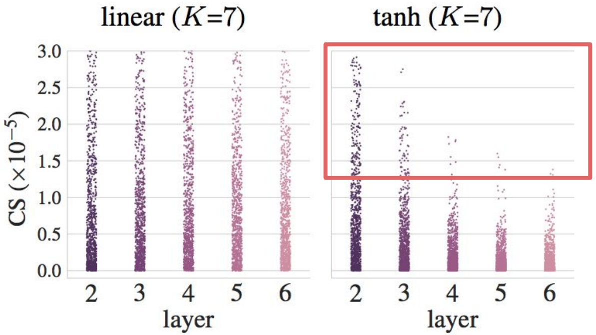
Sparsity pattern



(Linear) uniformly pruned throughout the network.
→ *learning capability secured*.

(tanh) more parameters pruned in the later layers.
→ *critical for high sparsity pruning*.

Sensitivity scores



CS scores decrease towards the later layers.
→ Choosing top salient parameters globally results in *a network, in which parameters are distributed non-uniformly and sparsely towards the end*.

CS metric can be decomposed as $\frac{\partial L(\mathbf{w};\mathcal{D})}{\partial \mathbf{w}} \odot \mathbf{w}$.
→ *necessary to ensure reliable gradient!*

Layerwise dynamical isometry for faithful gradients

Proposition 1 (*Gradients in terms of Jacobians*).

For a feed-forward network, the gradients satisfy: $\mathbf{g}_{\mathbf{w}^l}^T = \epsilon \mathbf{J}^{l,K} \mathbf{D}^l \otimes \mathbf{x}^{l-1}$, where $\epsilon = \partial L / \partial \mathbf{x}^K$ denotes the error signal, $\mathbf{J}^{l,K} = \partial \mathbf{x}^K / \partial \mathbf{x}^l$ is the Jacobian from layer l to the output layer K, and $\mathbf{D}^l \in \mathbb{R}^{N \times N}$ refers to the derivative of nonlinearity.

Layerwise dynamical isometry for faithful gradients

Proposition 1 (*Gradients in terms of Jacobians*).

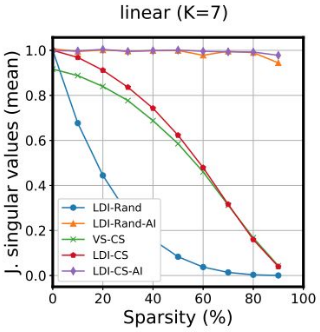
For a feed-forward network, the gradients satisfy: $\mathbf{g}_{\mathbf{w}^l}^T = \epsilon \mathbf{J}^{l,K} \mathbf{D}^l \otimes \mathbf{x}^{l-1}$, where $\epsilon = \partial L / \partial \mathbf{x}^K$ denotes the error signal, $\mathbf{J}^{l,K} = \partial \mathbf{x}^K / \partial \mathbf{x}^l$ is the Jacobian from layer l to the output layer K, and $\mathbf{D}^l \in \mathbb{R}^{N \times N}$ refers to the derivative of nonlinearity.

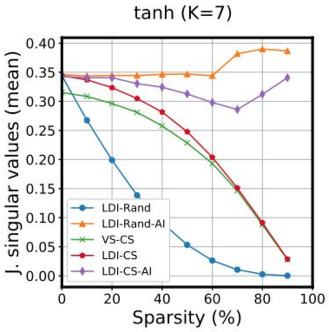
Definition 1 (*Layerwise dynamical isometry*).

Let $\mathbf{J}^{l-1,l} = \frac{\partial \mathbf{x}^l}{\partial \mathbf{x}^{l-1}} \in \mathbb{R}^{N_l \times N_{l-1}}$ be the Jacobian matrix of layer l. The network is said to satisfy layerwise dynamical isometry if the singular values of $\mathbf{J}^{l-1,l}$ are concentrated near 1 for all layers; *i.e.*, for a given $\epsilon > 0$, the singular value σ_j satisfies $|1 - \sigma_j| \leq \epsilon$ for all j.

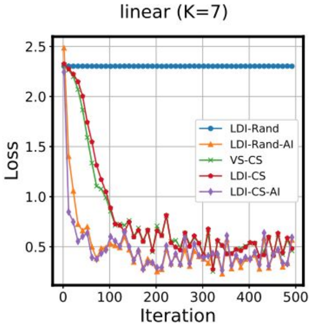
Signal propagation and trainability

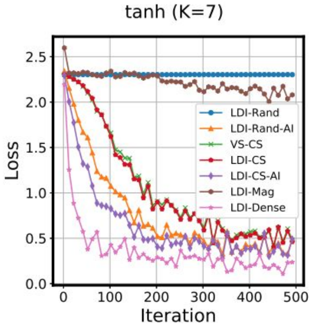
Signal propagation





Trainability (sparsity: 90%)

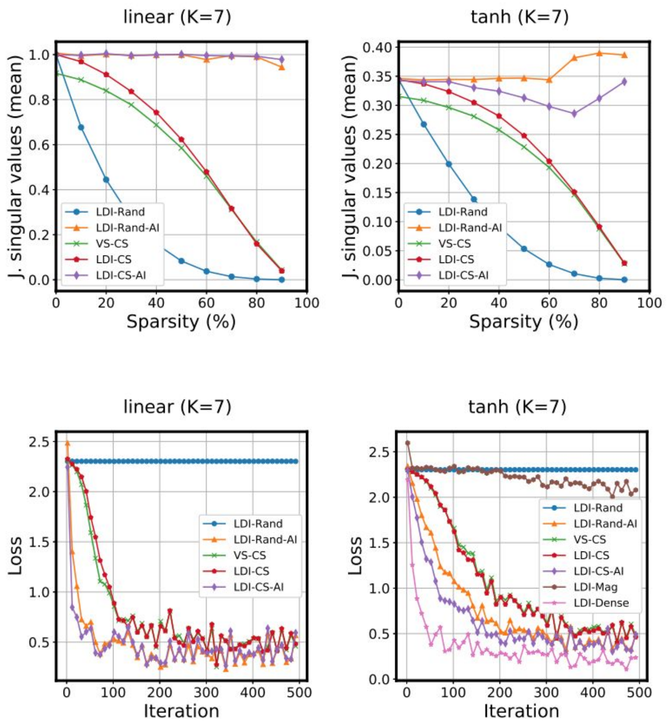




Signal propagation and trainability

Signal propagation

Trainability (sparsity: 90%)

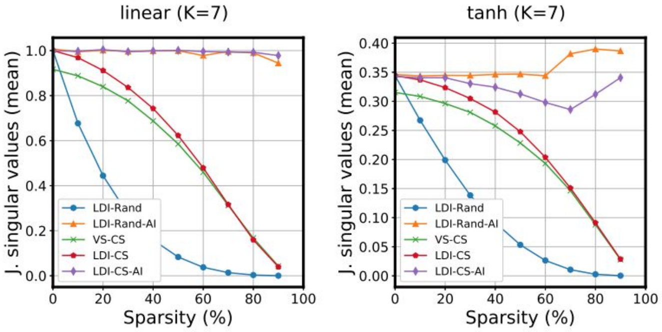


Jacobian singular values (JSV) decrease as per increasing sparsity.
→ *Pruning weakens signal propagation*.

JSV drop rapidly with random pruning, compared to connection sensitivity (CS) based pruning.
→ *CS pruning preserves signal propagation better*.

Signal propagation and trainability

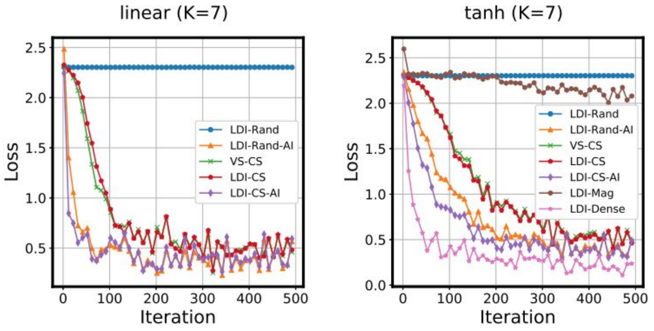
Signal propagation



Jacobian singular values (JSV) decrease as per increasing sparsity.
→ *Pruning weakens signal propagation*.

JSV drop rapidly with random pruning, compared to connection sensitivity (CS) based pruning.
→ *CS pruning preserves signal propagation better*.

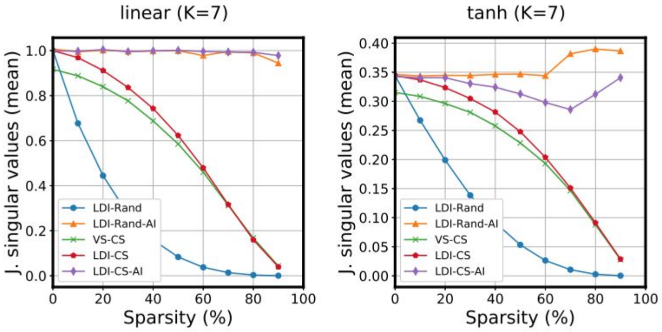
Trainability (sparsity: 90%)



Correlation between signal propagation and trainability.
→ *The better a network propagates signals,*
the faster it converges during training.

Signal propagation and trainability

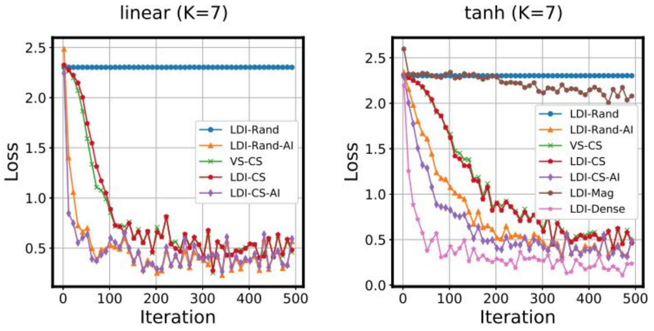
Signal propagation



Jacobian singular values (JSV) decrease as per increasing sparsity.
→ *Pruning weakens signal propagation*.

JSV drop rapidly with random pruning, compared to connection sensitivity (CS) based pruning.
→ *CS pruning preserves signal propagation better*.

Trainability (sparsity: 90%)



Correlation between signal propagation and trainability.
→ *The better a network propagates signals, the faster it converges during training*.

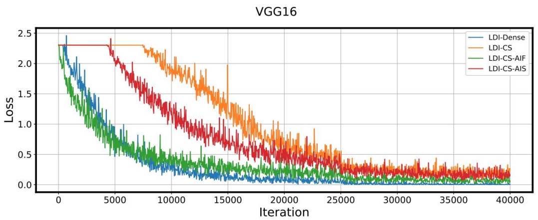
Enforce ***Approximate Isometry***:

$$\min_{\mathbf{W}^l} \|(\mathbf{C}^l \odot \mathbf{W}^l)^T(\mathbf{C}^l \odot \mathbf{W}^l) - \mathbf{I}^l\|_F$$

→ ***Restore signal propagation and improve training!***

Validations and extensions

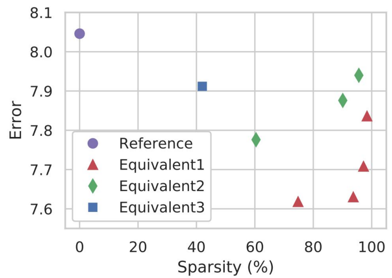
Modern networks



Pruning without supervision

Loss	Superv.	K=3	K=5	K=7
GT	✓	2.46	2.43	2.61
Pred. (raw)	✗	3.31	3.38	3.60
Pred. (softmax)	✗	3.11	3.37	3.56
Unif.	✗	2.77	2.77	2.94

Architecture sculpting



Non-linearities

	VGG16			ResNet32		
Initialization	tanh	l-relu	selu	tanh	l-relu	selu
VS-L	9.07	7.78	8.70	13.41	12.04	12.26
VS-G	9.06	7.84	8.82	13.44	12.02	12.32
VS-H	9.99	8.43	9.09	**13.12**	11.66	12.21
LDI	8.76	7.53	8.21	13.22	11.58	11.98
LDI-AI	**8.72**	**7.47**	**8.20**	13.14	**11.51**	**11.68**

Transfer of sparsity

	Dataset		Error		Error
Category	prune	train&test	sup. → unsup.	(Δ)	rand
Standard	MNIST	MNIST	2.42 → 2.94	+0.52	15.56
Transfer	F-MNIST	MNIST	2.66 → 2.80	**+0.14**	18.03
Standard	F-MNIST	F-MNIST	11.90 → 13.01	+1.11	24.72
Transfer	MNIST	F-MNIST	14.17 → 13.39	**-0.78**	24.89

Summary

- *The initial random weights have critical impact on pruning.*
- *Layerwise dynamical isometry ensures faithful signal propagation.*
- *Pruning breaks dynamical isometry and degrades trainability of a neural network.*
  Yet, enforcing approximate isometry can recover signal propagation and enhance trainability.
- *A range of experiments verify the effectiveness of signal propagation perspective.*