JHU vision lab

Mathematics of Deep Learning

René Vidal
Herschel Seder Professor of Biomedical Engineering
Director of the Mathematical Institute for Data Science
Johns Hopkins University

JOHNS HOPKINS MATHEMATICAL INSTITUTE for DATA SCIENCE
THE DEPARTMENT OF BIOMEDICAL ENGINEERING
The Whitaker Institute at Johns Hopkins
Center for Imaging Science

Brief History of Neural Networks

Beginnings			1st Neural Winter				2nd Neural Winter	GPU Era	
Thresholded Logic Unit 1943	Perceptron 1957	Adaline 1960	XOR Problem 1969	Multilayer Backprop 1982 1986	CNNs 1989	LSTMs 1997	SVMs 1995	Deep Nets 2006	Alex Net 2012

1940 **1950** **1960** **1970** **1980** **1990** **2000** **2010**

S. McCulloch - W. Pitts

R. Rosenblatt

B. Widrow - M. Hoff

M. Minsky - S. Papert

P. Werbos

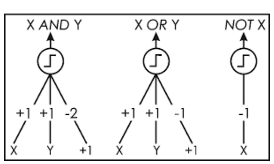
D. Rumelhart - G. Hinton - R. Williams

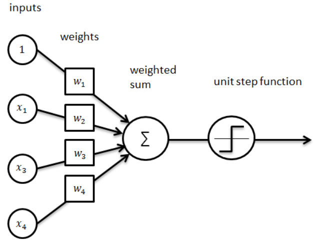
Y. Lecun

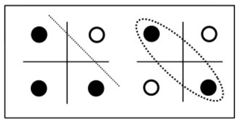
J. Schmidhuber

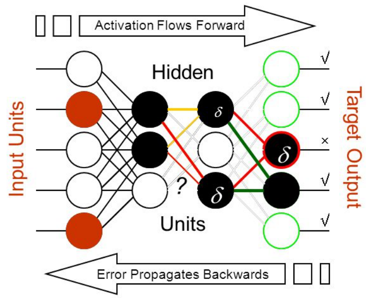
C. Cortes - V. Vapnik

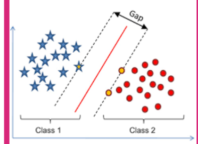
R. Salakhutdinov - J. Hinton - A. Krizhevsky - I. Sutskever

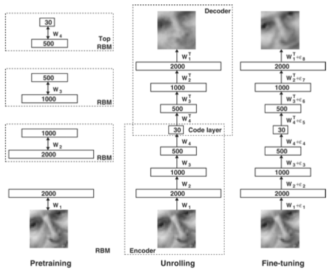












Impact of Deep Learning in Computer Vision

- 2012-2014 classification results in ImageNet

CNN
non-CNN

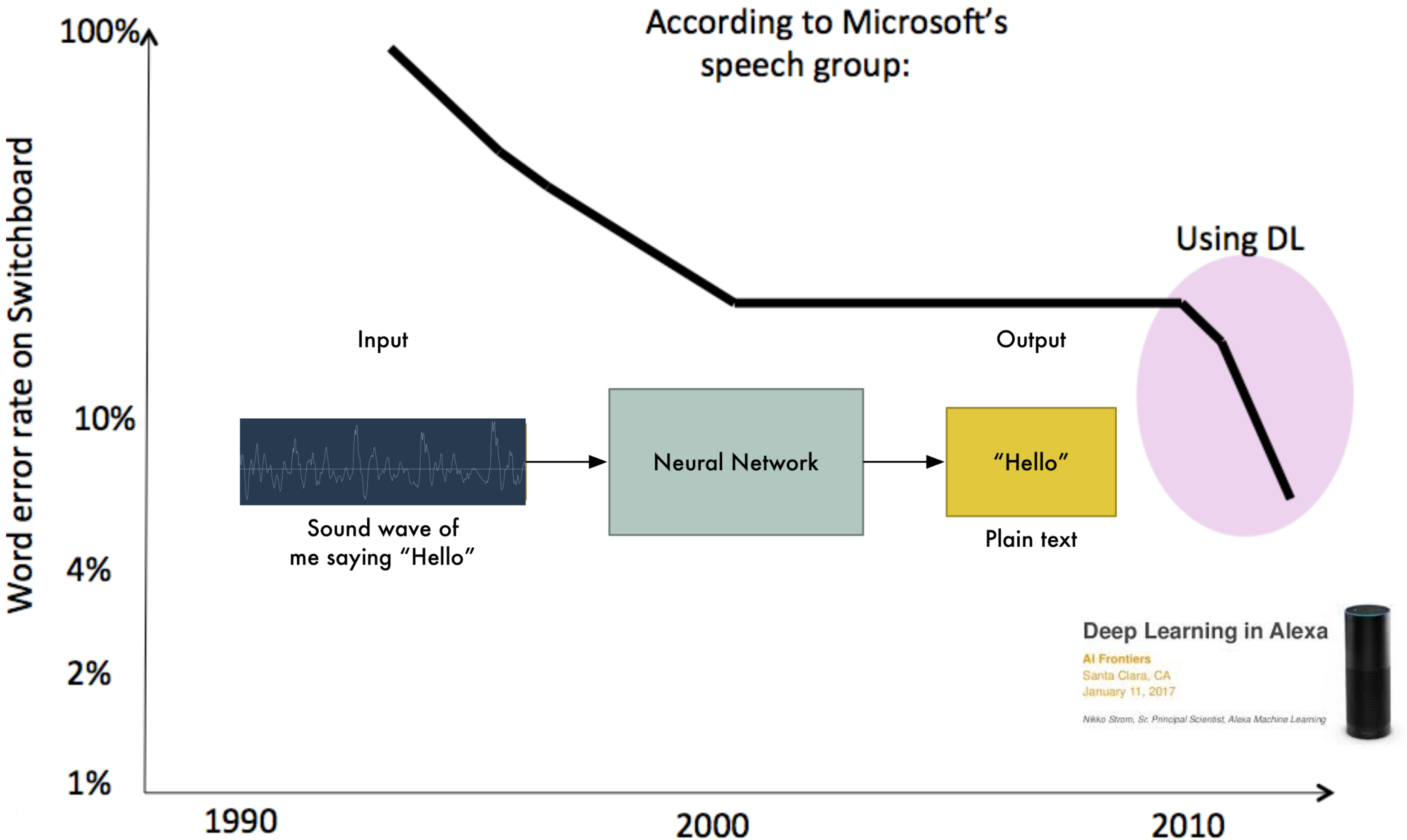
2012 Teams	%error
Supervision (Toronto)	15.3
ISI (Tokyo)	26.1
VGG (Oxford)	26.9
XRCE/INRIA	27.0
UvA (Amsterdam)	29.6
INRIA/LEAR	33.4

2013 Teams	%error
Clarifai (NYU spinoff)	11.7
NUS (singapore)	12.9
Zeiler-Fergus (NYU)	13.5
A. Howard	13.5
OverFeat (NYU)	14.1
UvA (Amsterdam)	14.2
Adobe	15.2
VGG (Oxford)	15.2
VGG (Oxford)	23.0

2014 Teams	%error
GoogLeNet	6.6
VGG (Oxford)	7.3
MSRA	8.0
A. Howard	8.1
DeeperVision	9.5
NUS-BST	9.7
TTIC-ECP	10.2
XYZ	11.2
UvA	12.1

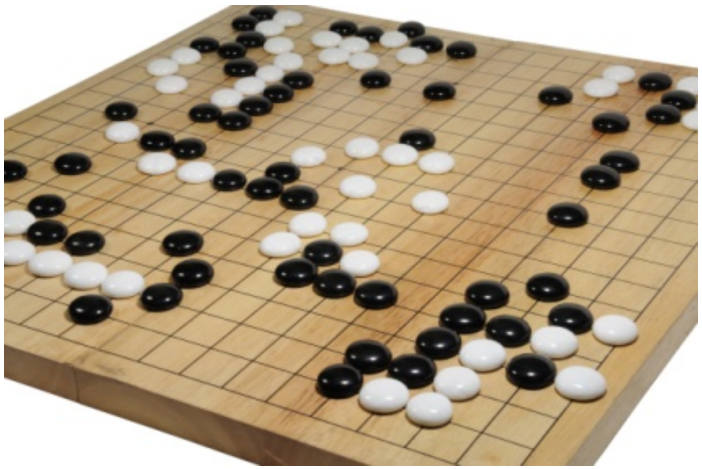
- 2015 results: ResNet under 3.5% error using 150 layers!

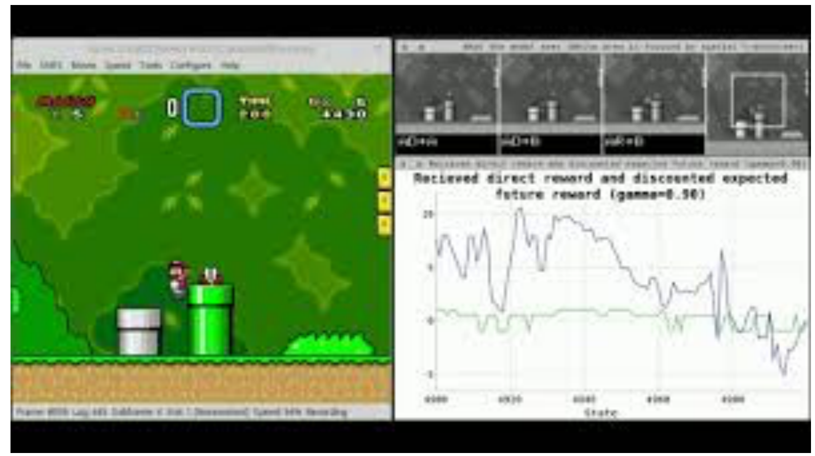
Impact of Deep Learning in Speech Recognition



Impact of Deep Learning in Game Playing

- **AlphaGo**: the first computer program to ever beat a professional player at the game of Go [1]



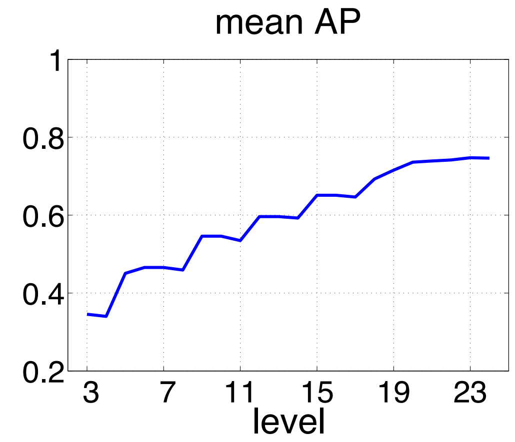


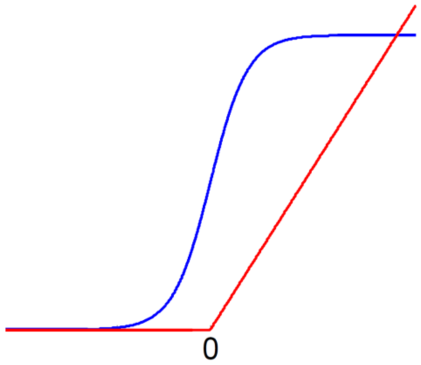
- Similar deep reinforcement learning strategies developed to play **Atari Breakout**, **Super Mario**

Silver et al. Mastering the game of Go with deep neural networks and tree search, Nature 2016
Artificial intelligence learns Mario level in just 34 attempts, https://www.engadget.com/2015/06/17/super-mario-world-self-learning-ai/, https://github.com/aleju/mario-ai

Why These Improvements in Performance?

- Features are learned rather than hand-crafted
- More layers capture more invariances [1]
- More data to train deeper networks
- More computing (GPUs)
- Better regularization: Dropout
- New nonlinearities
  - Max pooling, Rectified linear units (ReLU) [2]
- Theoretical understanding of deep networks remains shallow

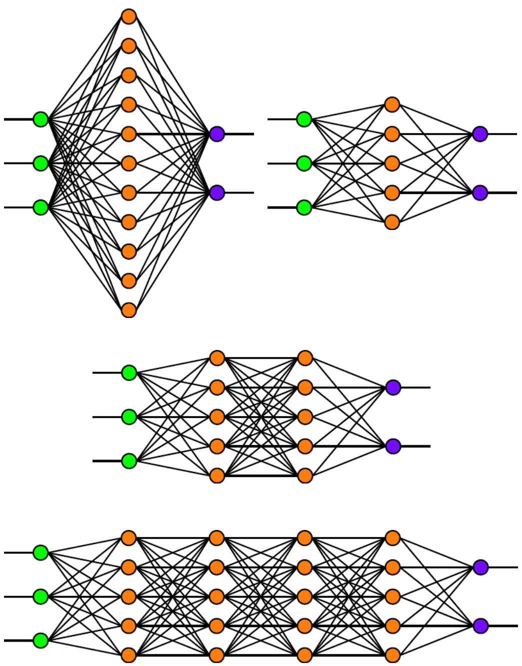




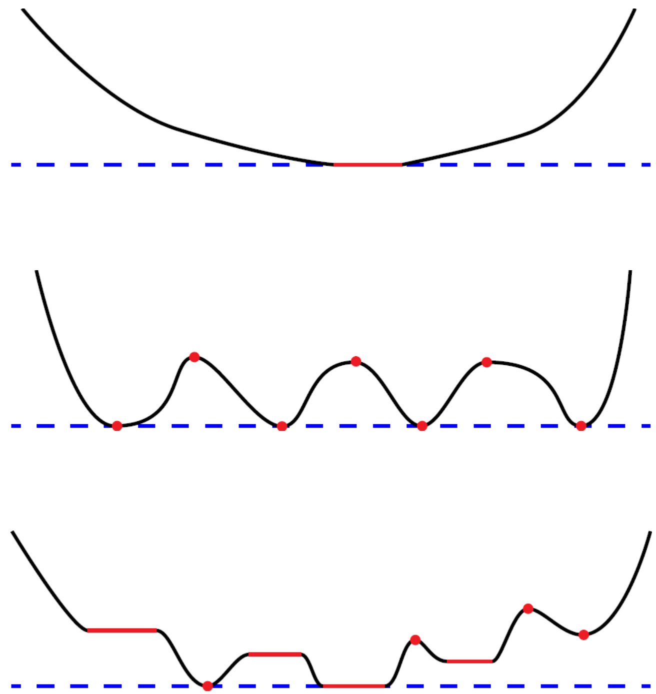
[1] Razavian, Azizpour, Sullivan, Carlsson, CNN Features off-the-shelf: an Astounding Baseline for Recognition. CVPRW'14.
[2] Hahnloser, Sarpeshkar, Mahowald, Douglas, Seung. Digital selection and analogue amplification coexist in a cortex-inspired silicon circuit. Nature, 405(6789):947–951, 2000.

Key Theoretical Questions in Deep Learning

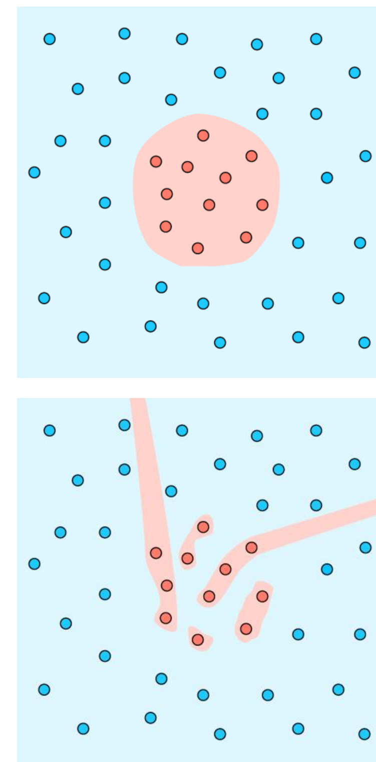
Architecture Design



Optimization

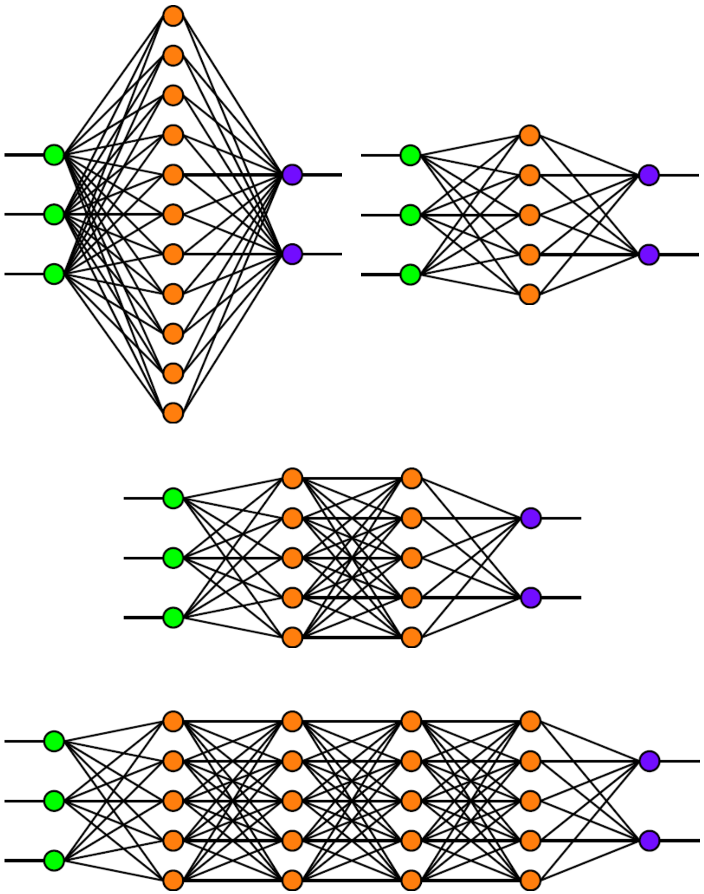


Generalization



Key Theoretical Questions: Architecture

- **Are there principled ways to design networks?**
  - How many layers?
  - Size of layers?
  - Choice of layer types?
  - What classes of functions can be approximated by a feedforward neural network?
  - How does the architecture impact expressiveness? [1]



[1] Cohen, et al., “On the expressive power of deep learning: A tensor analysis.” COLT. (2016)

Key Theoretical Questions: Architecture

- **Approximation, depth, width and invariance: earlier work**
  - Perceptrons and multilayer feedforward networks are universal approximators [Cybenko '89, Hornik '89, Hornik '91, Barron '93]

Theorem [C'89, H'91] Let $\rho()$ be a bounded, non-constant continuous function. Let I_m denote the m-dimensional hypercube, and $C(I_m)$ denote the space of continuous functions on I_m. Given any $f \in C(I_m)$ and $\epsilon > 0$, there exists $N > 0$ and v_i, w_i, b_i, $i = 1 \ldots, N$ such that

$$F(x) = \sum_{i \leq N} v_i \rho(w_i^T x + b_i) \quad \text{satisfies}$$

$$\sup_{x \in I_m} |f(x) - F(x)| < \epsilon \; .$$

Key Theoretical Questions: Architecture

- **Approximation, depth, width and invariance: earlier work**
  - Perceptrons and multilayer feedforward networks are universal approximators [Cybenko '89, Hornik '89, Hornik '91, Barron '93]

Theorem [Barron'92] The mean integrated square error between the estimated network $\hat{F}$ and the target function f is bounded by

$$O\left(\frac{C_f^2}{N}\right) + O\left(\frac{Nm}{K}\log K\right) \quad ,$$

where K is the number of training points, N is the number of neurons, m is the input dimension, and C_f measures the global smoothness of f.

Key Theoretical Questions: Architecture

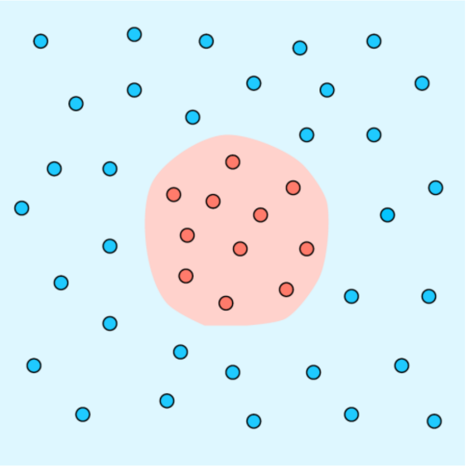
- **Approximation, depth, width and invariance: earlier work**
  - Perceptrons and multilayer feedforward networks are universal approximators [Cybenko '89, Hornik '89, Hornik '91, Barron '93]
- **Approximation, depth, width and invariance: recent work**
  - Gaps between deep and shallow networks [Montufar'14, Mhaskar'16]
  - Deep Boltzmann machines are universal approximators [Montufar'15]
  - Design of CNNs via hierarchical tensor decompositions [Cohen '17]
  - Scattering networks are deformation stable for Lipschitz non-linearities [Bruna-Mallat '13, Wiatowski '15, Mallat '16]
  - Exponential # of units needed to approximate deep net [Telgarsky'16]
  - Memory-optimal neural network approximation [Bölcskei '17]

[1] Cybenko. Approximations by superpositions of sigmoidal functions, Mathematics of Control, Signals, and Systems, 2 (4), 303-314, 1989.
[2] Hornik, Stinchcombe and White. Multilayer feedforward networks are universal approximators, Neural Networks, 2(3), 359-366, 1989.
[3] Hornik. Approximation Capabilities of Multilayer Feedforward Networks, Neural Networks, 4(2), 251–257, 1991.
[4] Barron. Universal approximation bounds for superpositions of a sigmoidal function. IEEE Transactions on Information Theory, 39(3):930–945, 1993.
[5] Cohen et al. Analysis and Design of Convolutional Networks via Hierarchical Tensor Decompositions arXiv preprint arXiv:1705.02302
[6] Montúfar, Pascanu, Cho, Bengio, On the number of linear regions of deep neural networks, NIPS, 2014
[7] Mhaskar, Poggio. Deep vs. shallow networks: An approximation theory perspective. Analysis and Applications, 2016.
[8] Montúfar et al, Deep narrow Boltzmann machines are universal approximators, ICLR 2015, arXiv:1411.3784v3
[9] Bruna and Mallat. Invariant scattering convolution networks. Trans. PAMI, 35(8):1872–1886, 2013.
[10] Wiatowski, Bölcskei. A mathematical theory of deep convolutional neural networks for feature extraction. arXiv2015.
[11] Mallat. Understanding deep convolutional networks. Phil. Trans. R. Soc. A, 374(2065), 2016.
[12] Telgarsky, Benefits of depth in neural networks. COLT 2016.
[13] Bölcskei, Grohs, Kutyniok, Petersen. Memory-optimal neural network approximation. Wavelets and Sparsity 2017.

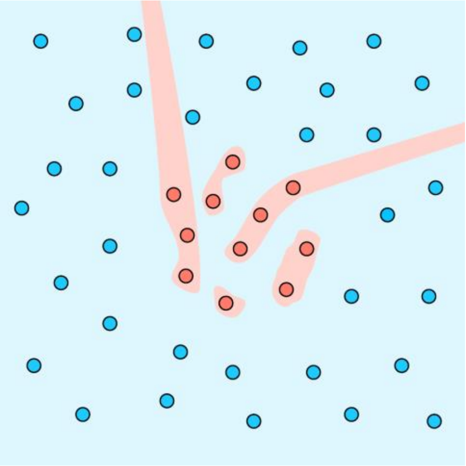
Key Theoretical Questions: Generalization

- **Classification performance guarantees?**
  - How well do deep networks generalize?
  - How should networks be regularized?
  - How to prevent under or over fitting?

✓ Simple



✗ Complex



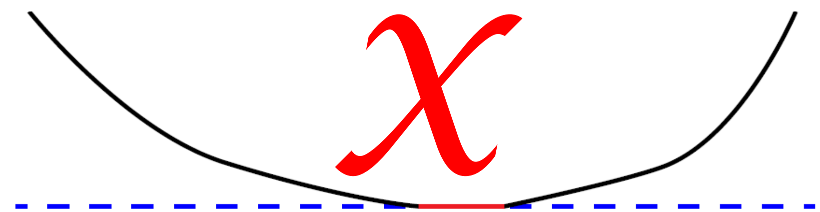
Key Theoretical Questions: Generalization

- **Generalization and regularization theory: earlier work**
  - # training examples grows polynomially with network size [1,2]
- **Regularization methods: earlier and recent work**
  - Early stopping [3]
  - Dropout, Dropconnect, and extensions (adaptive, annealed) [4,5]
- **Generalization and regularization theory: recent work**
  - Distance and margin-preserving embeddings [6,7]
  - Path SGD/implicit regularization & generalization bounds [8,9]
  - Product of norms regularization & generalization bounds [10,11]
  - Information theory: info bottleneck, info dropout, Fisher-Rao [12,13,14]
  - Rethinking generalization: [15]

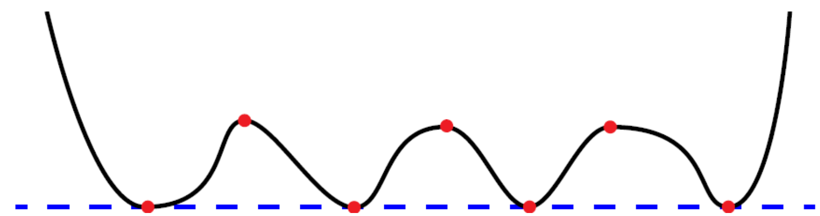
[1] Sontag. VC Dimension of Neural Networks. Neural Networks and Machine Learning, 1998.
[2] Bartlett, Maass. VC dimension of neural nets. The handbook of brain theory and neural networks, 2003.
[3] Caruana, Lawrence, Giles. Overfitting in neural nets: Backpropagation, conjugate gradient & early stopping. NIPS01.
[4] Srivastava. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. JMLR, 2014.
[5] Wan. Regularization of neural networks using dropconnect. ICML, 2013.
[6] Giryes, Sapiro, Bronstein. Deep Neural Networks with Random Gaussian Weights. arXiv:1504.08291.
[7] Sokolic. Margin Preservation of Deep Neural Networks, 2015
[8] Neyshabur. Path-SGD: Path-Normalized Optimization in Deep Neural Networks. NIPS 2015
[9] Behnam Neyshabur. Implicit Regularization in Deep Learning. PhD Thesis 2017
[10] Sokolic, Giryes, Sapiro, Rodrigues. Generalization error of invariant classifiers. In AISTATS, 2017.
[11 Sokolić, Giryes, Sapiro, Rodrigues. Robust Large Margin Deep Neural Networks. IEEE Transactions on Signal Processing, 2017.
[12] Shwartz-Ziv, Tishby. Opening the black box of deep neural networks via information. arXiv:1703.00810, 2017.
[13] Achille, Soatto. Information dropout: Learning optimal representations through noisy computation. *arXiv:* 2016.
[14] Liang, Poggio, Rakhlin, Stokes. Fisher-Rao Metric, Geometry and Complexity of Neural Networks. arXiv: 2017.
[15] Zhang, Bengio, Hardt, Recht, Vinyals. Understanding deep learning requires rethinking generalization. ICLR 2017.

Key Theoretical Questions: Optimization

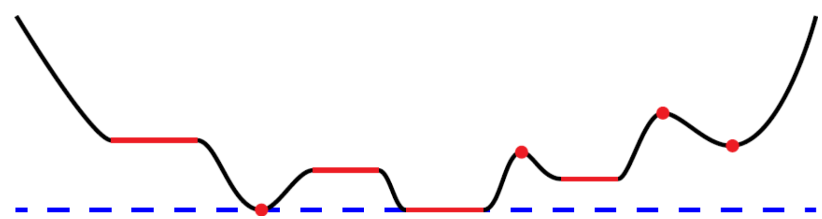
- **How to train neural networks?**
  - Problem is non-convex



  - What does the error surface look like?



  - How to guarantee optimality?
  - When does local descent succeed?

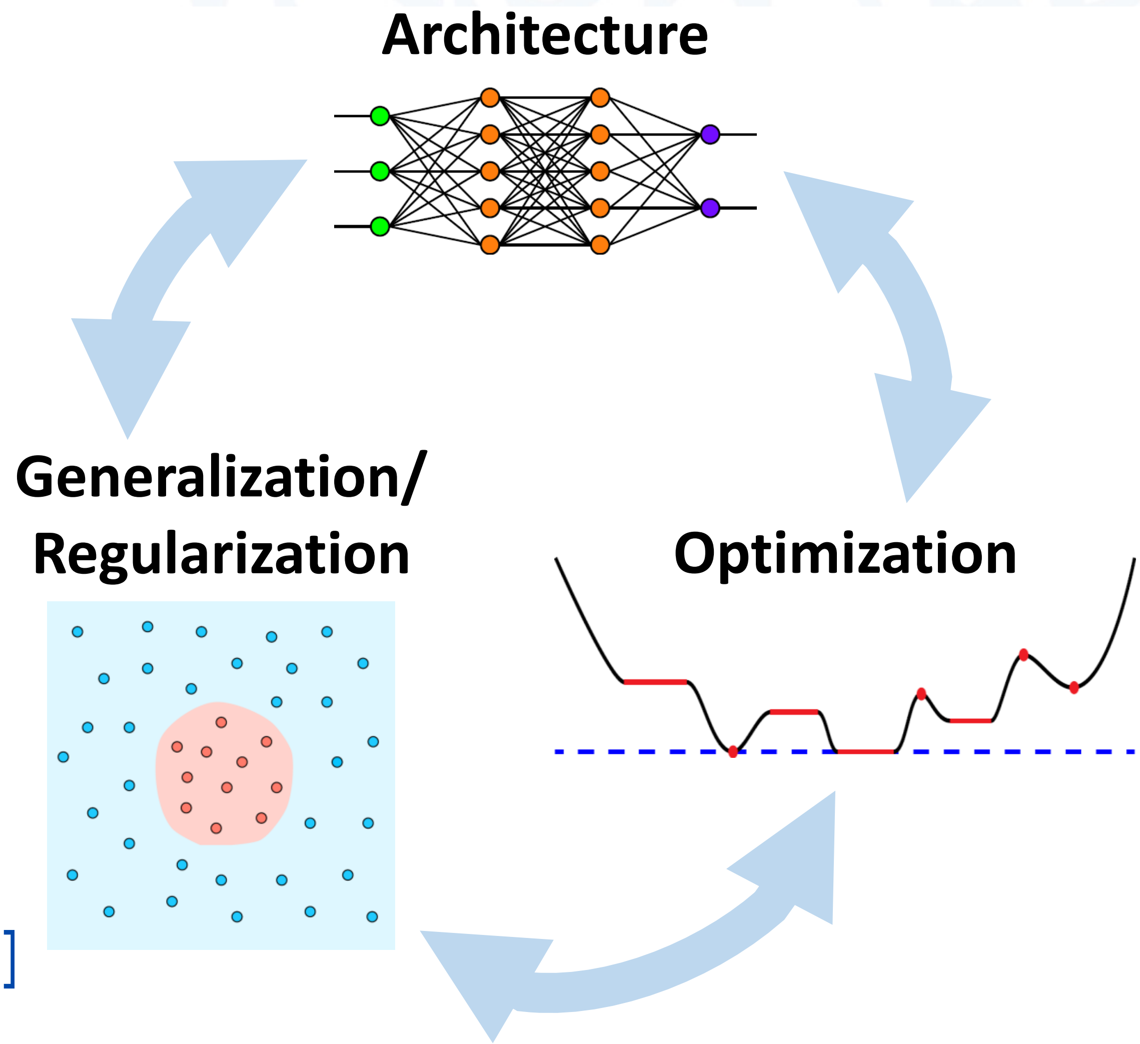


Key Theoretical Questions: Optimization

- **Optimization theory: earlier work**
  - No spurious local minima for linear networks [Baldi-Hornik ’89]
  - Backprop fails to converge for nonlinear networks [Brady’89], converges for linearly separable data [Gori-Tesi’91-’92], or it gets stuck [Frasconi’97]
  - Local minima and plateaus in multilayer perceptrons [Fukumizu-Amari’00]
- **Optimization theory: recent work**
  - Convex neural networks in infinite number of variables [Bengio ’05]
  - Networks with many hidden units can learn polynomials [Andoni ’14]
  - The loss surface of multilayer networks [Choromanska ’15]
  - Attacking the saddle point problem [Dauphin ’14]
  - Effect of gradient noise on the energy landscape: [Chaudhari ‘15]
  - Entropy-SGD is biased toward wide valleys: [Chaudhari ‘17]
  - Deep relaxation: PDEs for optimizing deep nets [Chaudhari ‘17]
  - Guaranteed training of NNs using tensor methods [Janzamin ’15]
  - No spurious local minima for large networks [Haeffele-Vidal’15 Soudry’16]

Key Theoretical Questions are Interrelated

- Optimization can impact generalization [1,2]
- Architecture has strong effect on generalization [3]
- Some architectures could be easier to optimize than others [4]

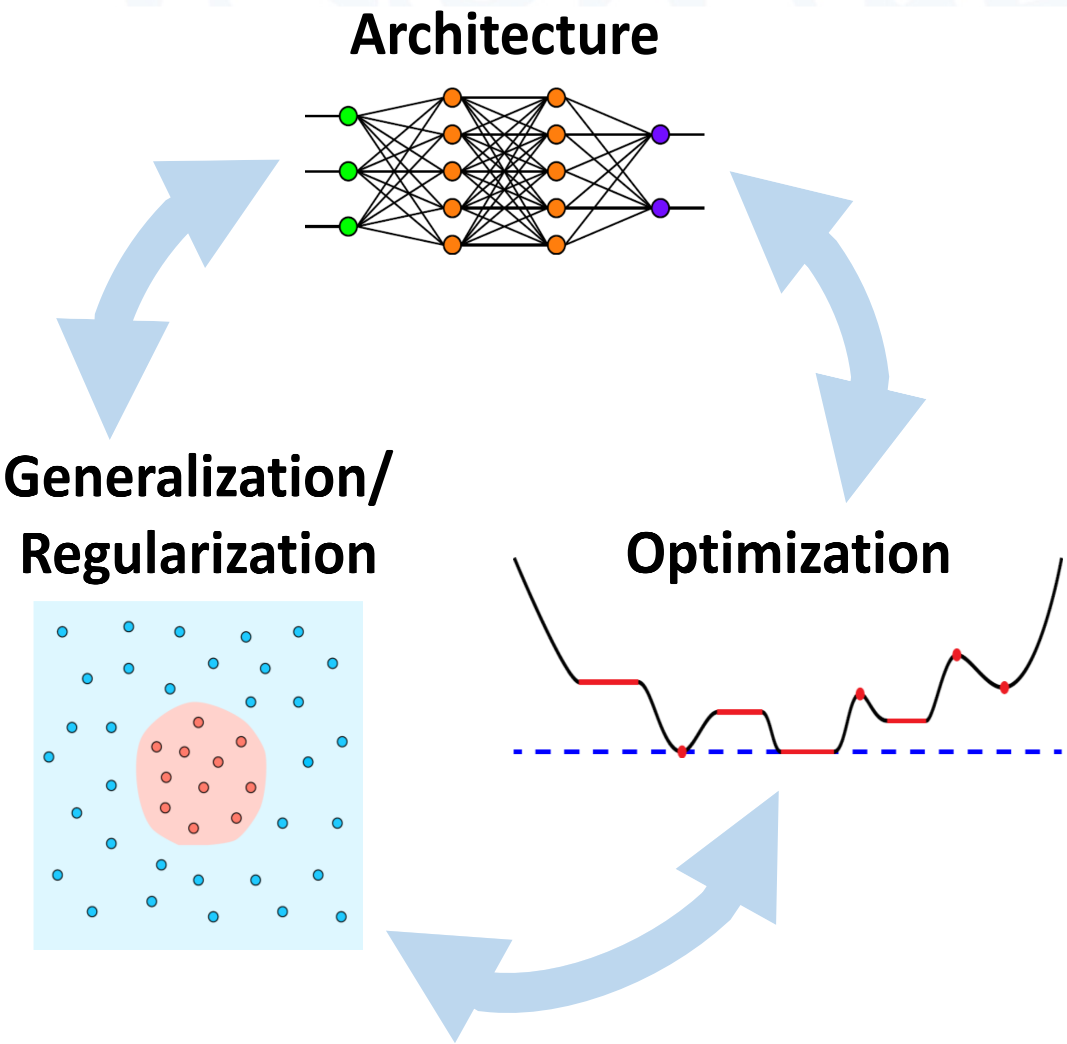


[1] Neyshabur et. al. In Search of the Real Inductive Bias: On the Role of Implicit Regularization in Deep Learning." ICLR workshop. (2015).
[2] P. Zhou, J. Feng. The Landscape of Deep Learning Algorithms. 1705.07038, 2017
[3] Zhang, et al., "Understanding deep learning requires rethinking generalization." ICLR. (2017).
[4] Haeffele, Vidal. Global optimality in neural network training. CVPR 2017.

JOHNS HOPKINS
MATHEMATICAL INSTITUTE
for DATA SCIENCE

Toward a Unified Theory?

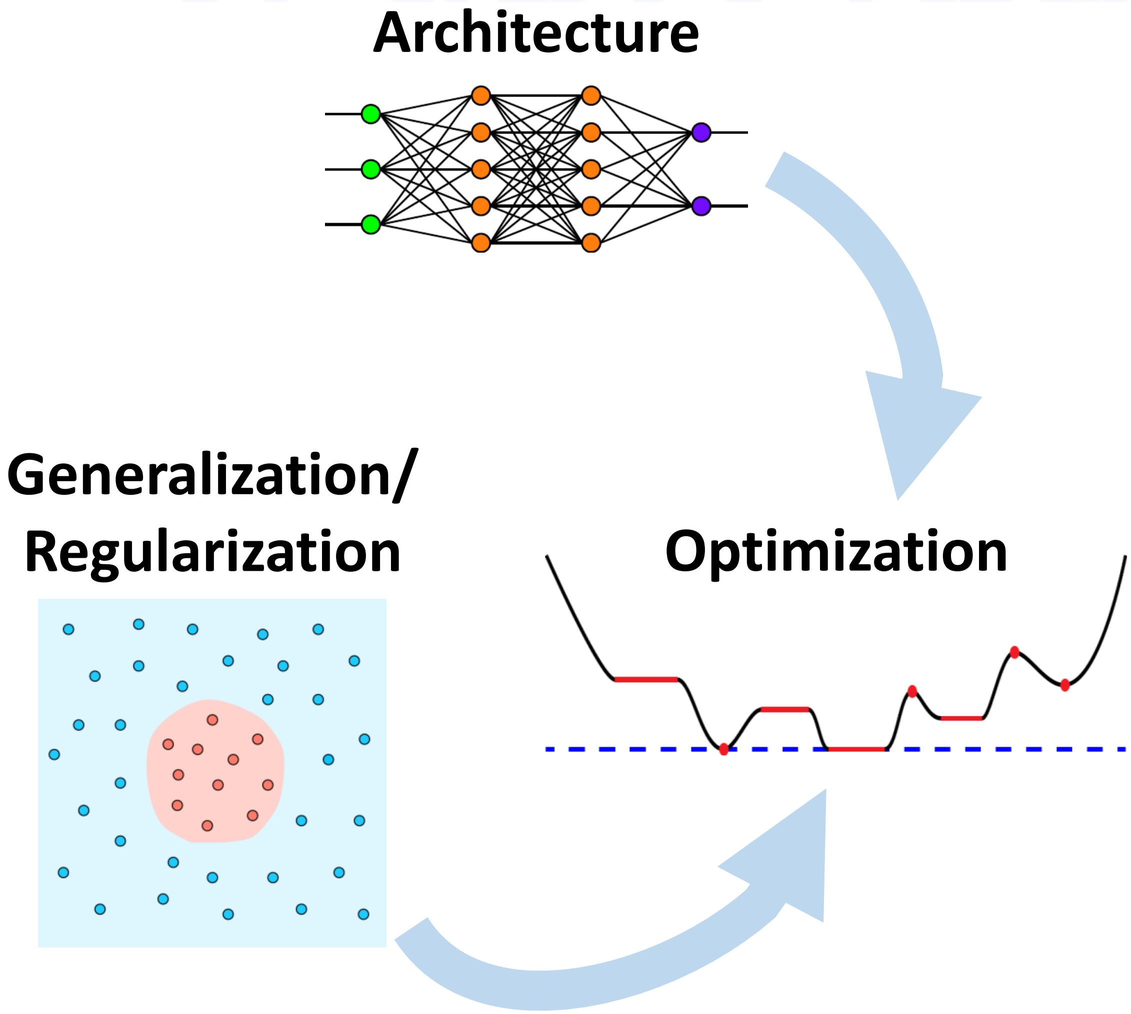
- Dropout regularization is equivalent to regularization with products of weights [1,2]
- Regularization with product of weights generalizes well [3,4]
- No spurious local minima for product of weight regularizers [5]



[1] Cavazza, Lane, Moreiro, Haeffele, Murino, Vidal. An Analysis of Dropout for Matrix Factorization, AISTATS 2018.
[2] Poorya Mianjy, Raman Arora, Rene Vidal. On the Implicit Bias of Dropout. ICML 2018.
[3] Neyshabur, Salakhutdinov, Srebro. Path-SGD: Path-Normalized Optimization in Deep Neural Networks. NIPS 2015
[4] Sokolic, Giryes, Sapiro, Rodrigues. Generalization error of Invariant Classifiers. AISTATS, 2017.
[5] Haeffele, Vidal. Global optimality in neural network training. CVPR 2017.

Part I: Analysis of Optimization

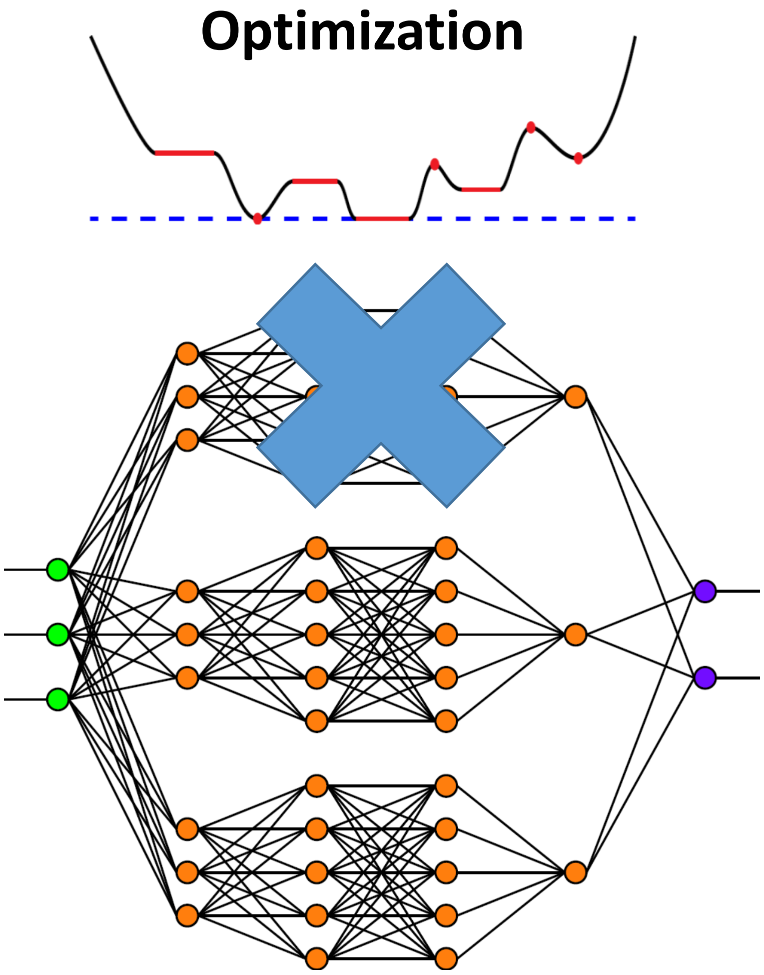
- What properties of the network architecture facilitate optimization?
  - Positive homogeneity
  - Parallel subnetwork structure
- What properties of the regularization function facilitate optimization?
  - Positive homogeneity
  - Adapt network structure to the data [1]



Picture courtesy of Ben Haeffele

[1] Bengio, et al., “Convex neural networks.” NIPS. (2005)

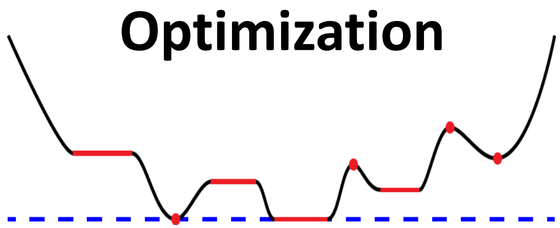
Main Results



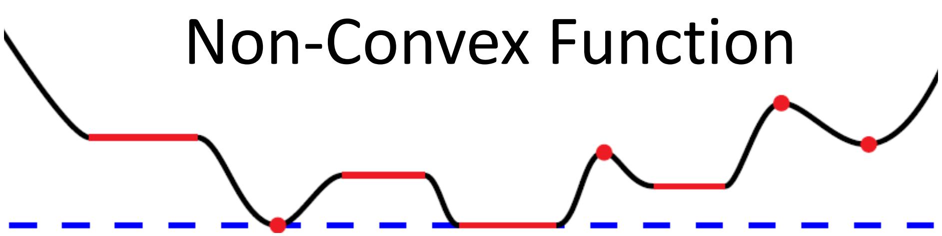
Theorem 1:
A local minimum such that all the weights from one subnetwork are zero is a global minimum

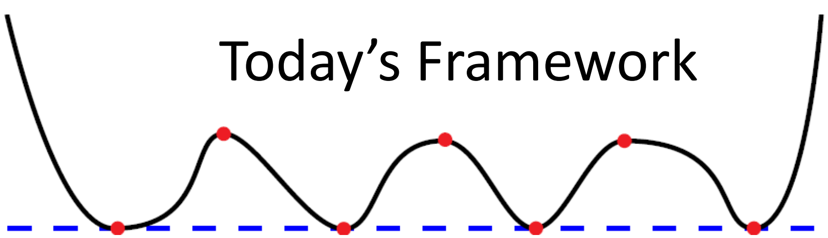
[1] Haeffele, Young, Vidal. Structured Low-Rank Matrix Factorization: Optimality, Algorithm, and Applications to Image Processing, ICML '14
[2] Haeffele, Vidal. Global Optimality in Tensor Factorization, Deep Learning and Beyond, arXiv, '15
[3] Haeffele, Vidal. Global optimality in neural network training. CVPR 2017.

Main Results



Theorem 2:
If the size of the network is large enough, local descent can reach a global minimizer from any initialization

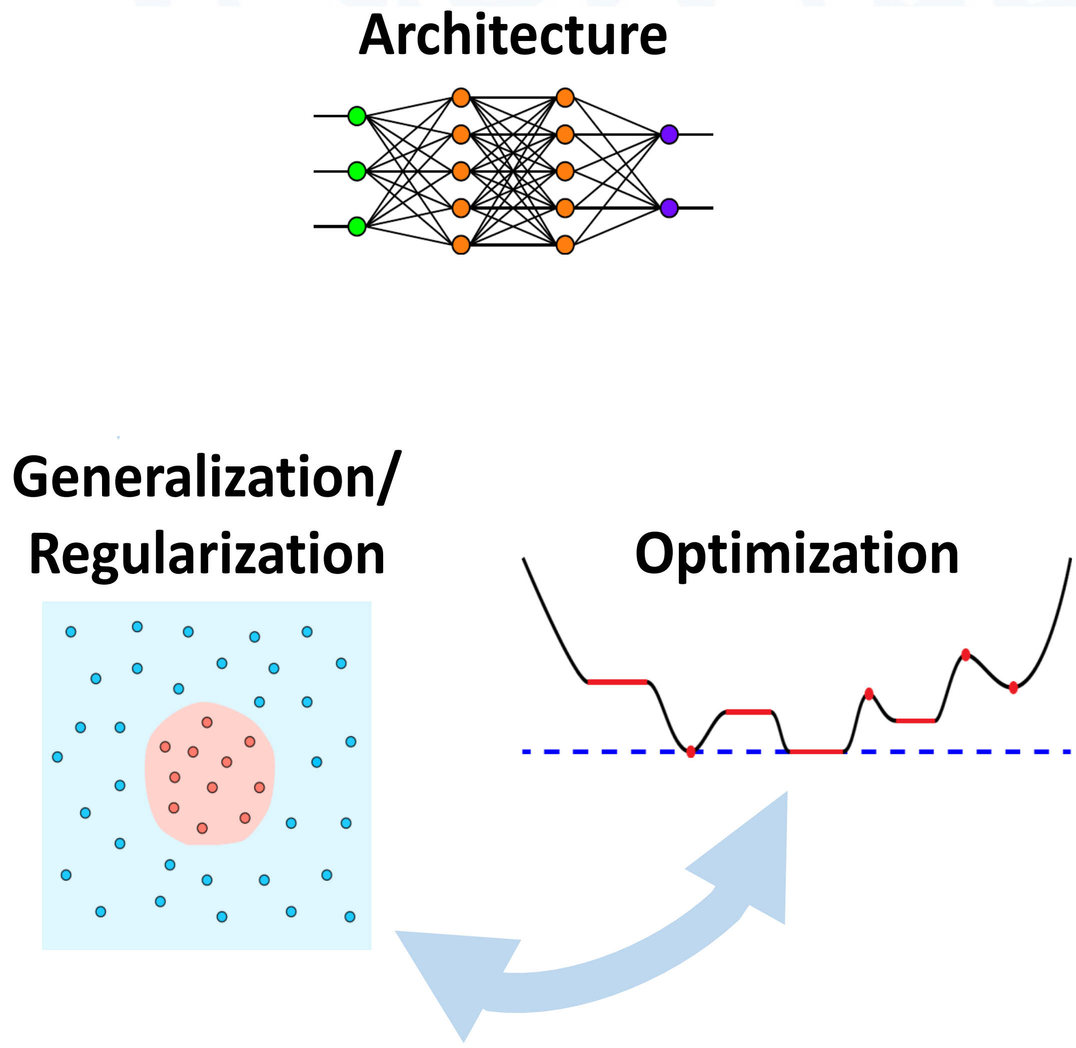




[1] Haeffele, Young, Vidal. Structured Low-Rank Matrix Factorization: Optimality, Algorithm, and Applications to Image Processing, ICML '14
[2] Haeffele, Vidal. Global Optimality in Tensor Factorization, Deep Learning and Beyond, arXiv, '15
[3] Haeffele, Vidal. Global optimality in neural network training. CVPR 2017.

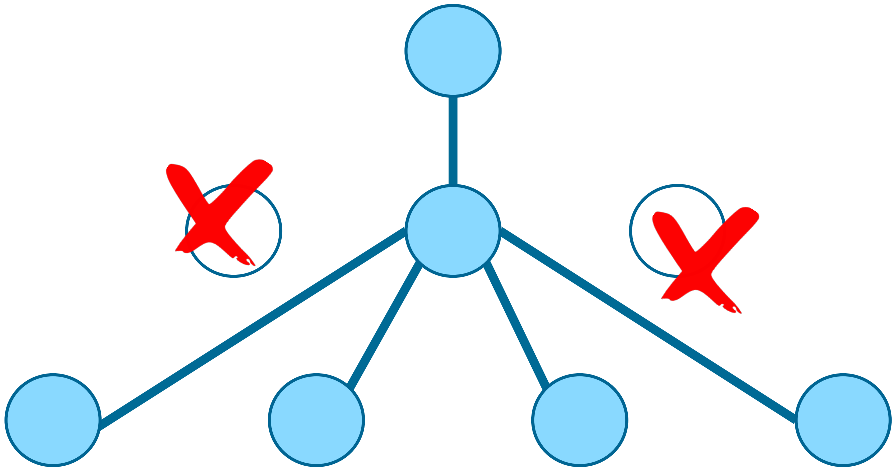
Part II: Analysis of Dropout for Linear Nets

- What objective function is being minimized by dropout?
- What type of regularization is induced by dropout?
- What are the properties of the optimal weights?



Picture courtesy of Ben Haeffele

Main Results for Linear Nets



Theorem 3:
Dropout is SGD applied to a stochastic objective.

Theorem 4:
Dropout induces explicit low-rank regularization (nuclear norm squared).

Theorem 5:
Dropout induces balanced weights.

JHU vision lab

Global Optimality in Matrix and Tensor Factorization, Deep Learning & Beyond

Ben Haeffele and René Vidal
Center for Imaging Science
Mathematical Institute for Data Science
Johns Hopkins University

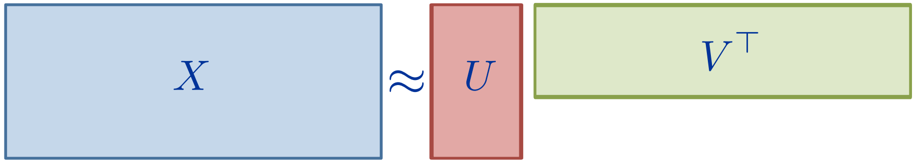
JOHNS HOPKINS MATHEMATICAL INSTITUTE *for* DATA SCIENCE

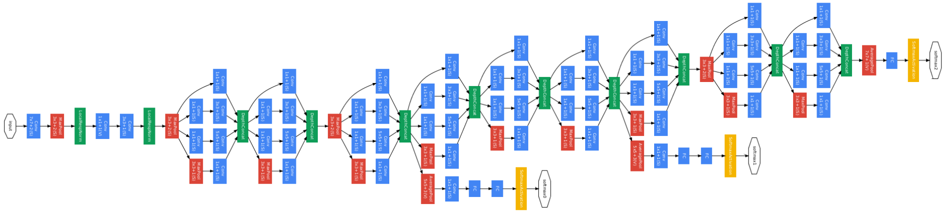
THE DEPARTMENT OF BIOMEDICAL ENGINEERING
The Whitaker Institute at Johns Hopkins

Center for IMAGING SCIENCE

Outline

- **Architecture properties that facilitate optimization**
  - Positive homogeneity
  - Parallel subnetwork structure
- **Regularization properties that facilitate optimization**
  - Positive homogeneity
  - Adapt network structure to the data
- **Theoretical guarantees**
  - Sufficient conditions for global optimality
  - Local descent can reach global minimizers

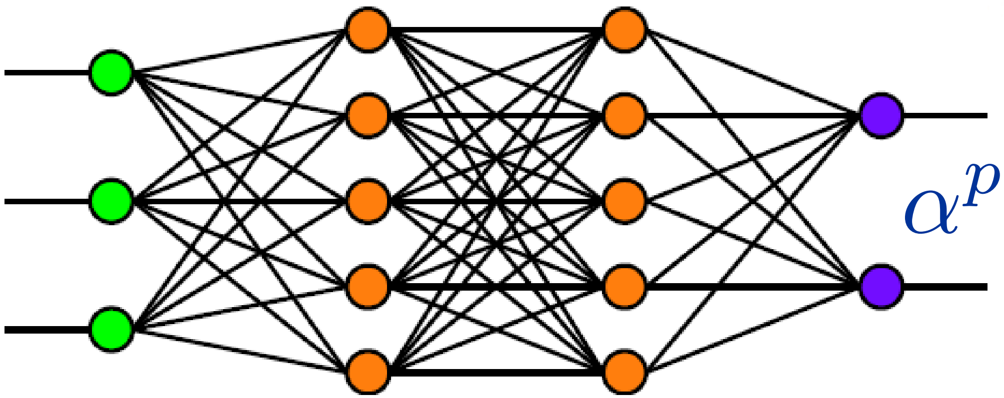




[1] Haeffele, Young, Vidal. Structured Low-Rank Matrix Factorization: Optimality, Algorithm, and Applications to Image Processing, ICML '14
[2] Haeffele, Vidal. Global Optimality in Tensor Factorization, Deep Learning and Beyond, arXiv, '15
[3] Haeffele, Vidal. Global optimality in neural network training. CVPR 2017.

Key Property #1: Positive Homogeneity

- Start with a network



- Scale the weights by

$$\alpha \geq 0$$

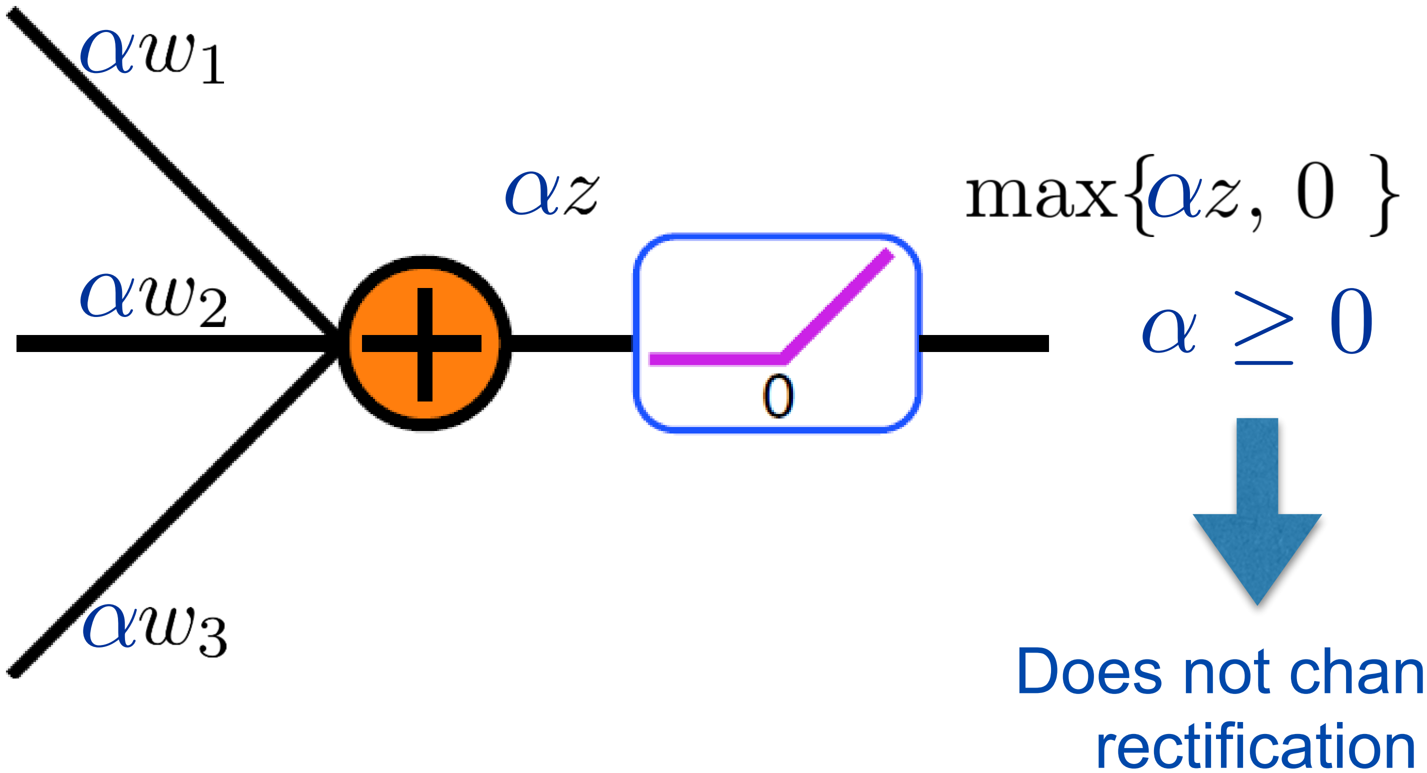
- Output is scaled by α^p, where p = degree of homogeneity

$$\Phi(W^1, W^2, W^3) = Y$$

$$\Phi(\alpha W^1, \alpha W^2, \alpha W^3) = \alpha^p Y$$

Examples of Positively Homogeneous Maps

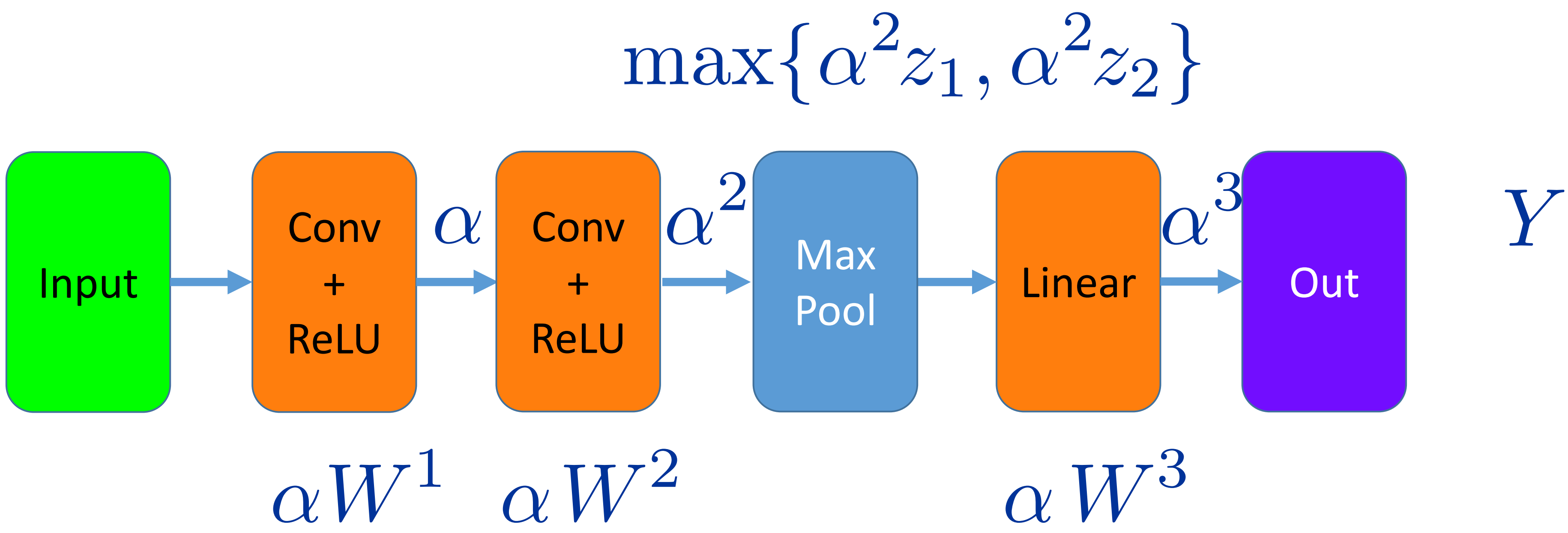
- **Example 1**: Rectified Linear Units (ReLU)



- Linear + ReLU layer is positively homogeneous of degree 1

Examples of Positively Homogeneous Maps

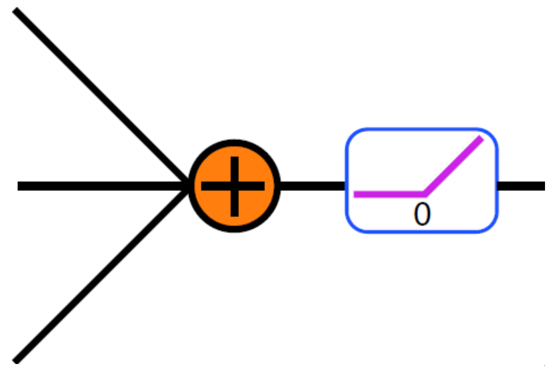
- **Example 2**: Simple networks with convolutional layers, ReLU, max pooling and fully connected layers

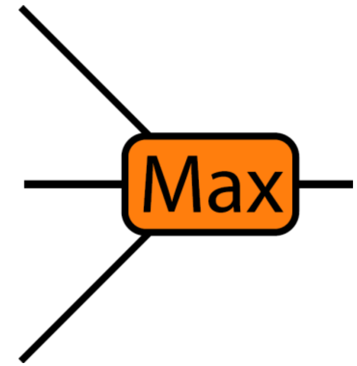


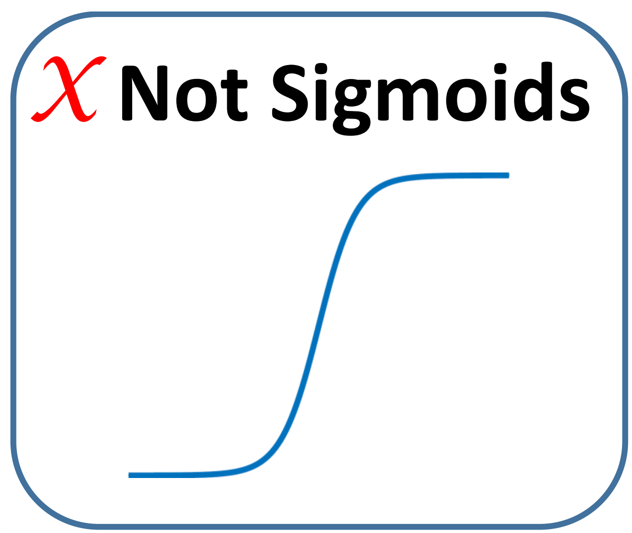
- Typically each weight layer increases degree of homogeneity by 1

Examples of Positively Homogeneous Maps

- Some Common Positively Homogeneous Layers
  - Fully Connected + ReLU
  - Convolution + ReLU
  - Max Pooling
  - Linear Layers
  - Mean Pooling
  - Max Out
  - Many possibilities...





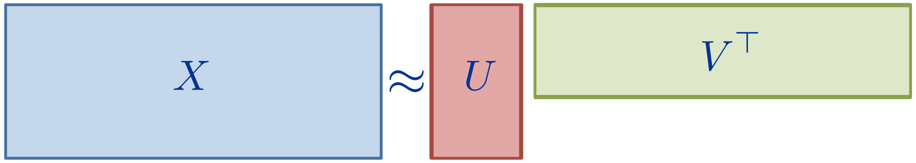


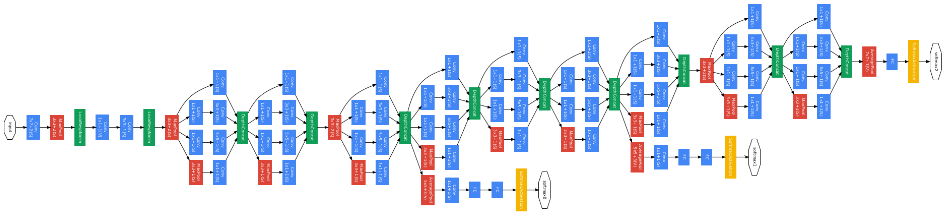
Outline

- **Architecture properties that facilitate optimization**
  - Positive homogeneity
  - Parallel subnetwork structure

- **Regularization properties that facilitate optimization**
  - Positive homogeneity
  - Adapt network structure to the data

- **Theoretical guarantees**
  - Sufficient conditions for global optimality
  - Local descent can reach global minimizers

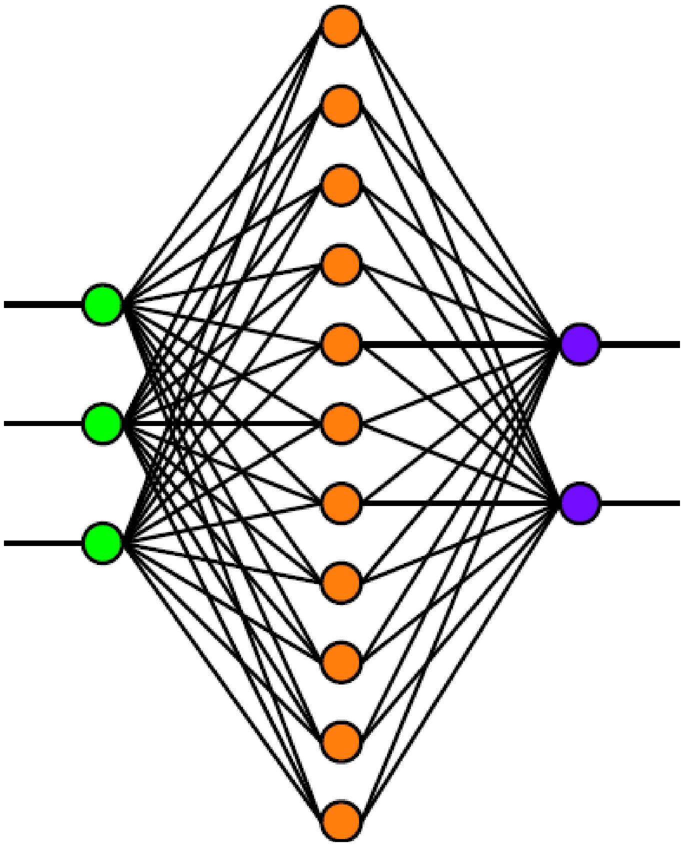




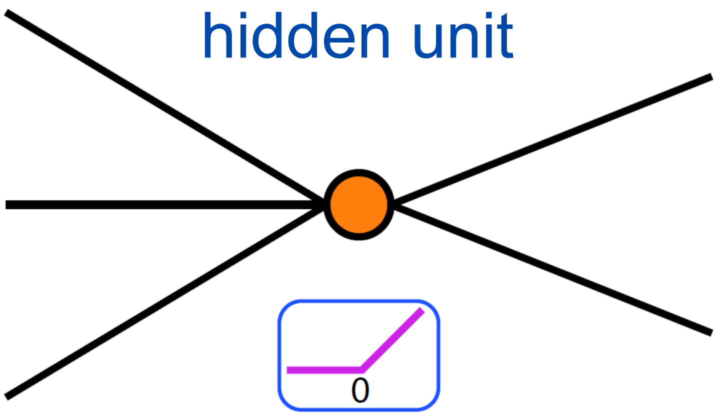
[1] Haeffele, Young, Vidal. Structured Low-Rank Matrix Factorization: Optimality, Algorithm, and Applications to Image Processing, ICML '14
[2] Haeffele, Vidal. Global Optimality in Tensor Factorization, Deep Learning and Beyond, arXiv, '15
[3] Haeffele, Vidal. Global optimality in neural network training. CVPR 2017.

Key Property #2: Parallel Subnetworks

- Subnetworks with identical structure connected in parallel
- **Simple example**: single hidden network

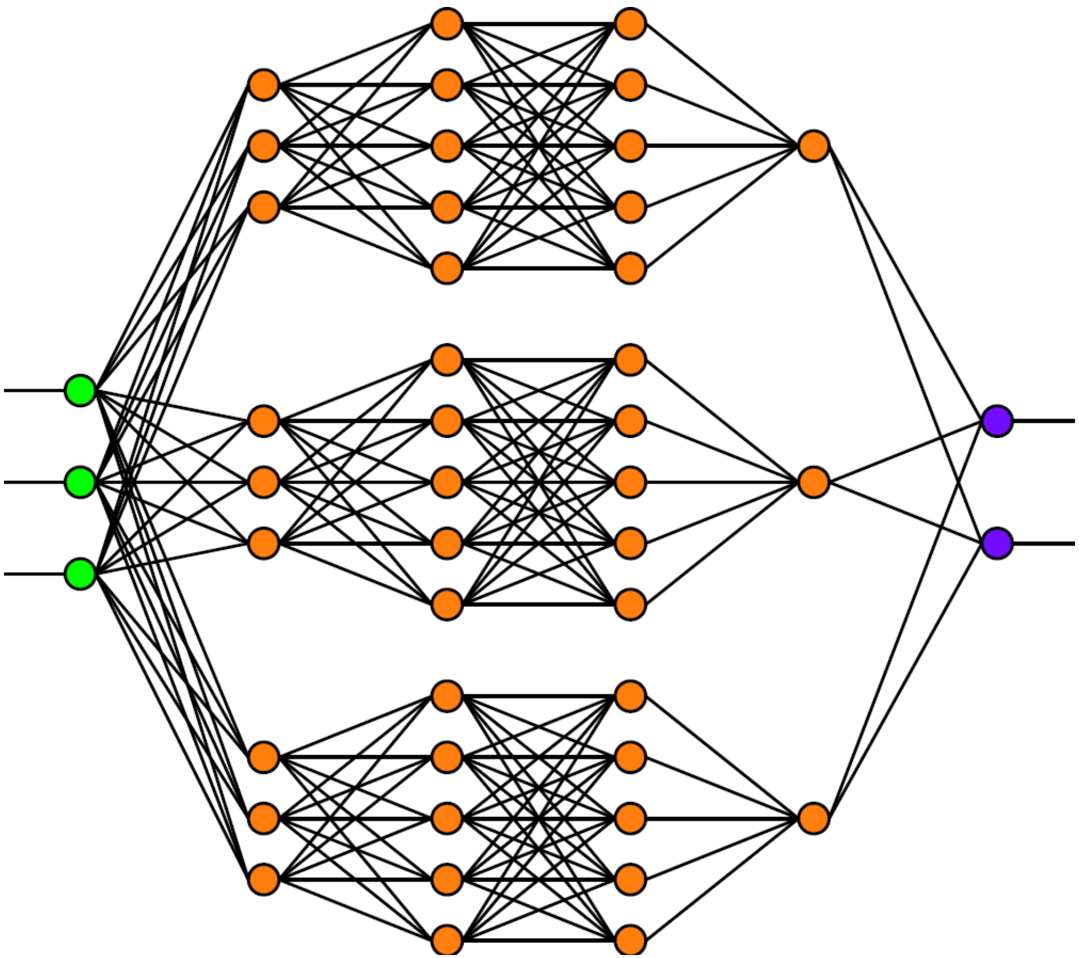


Subnetwork:
one ReLU
hidden unit

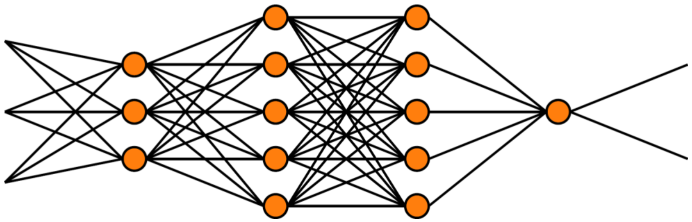


Key Property #2: Parallel Subnetworks

- Any positively homogeneous network can be used

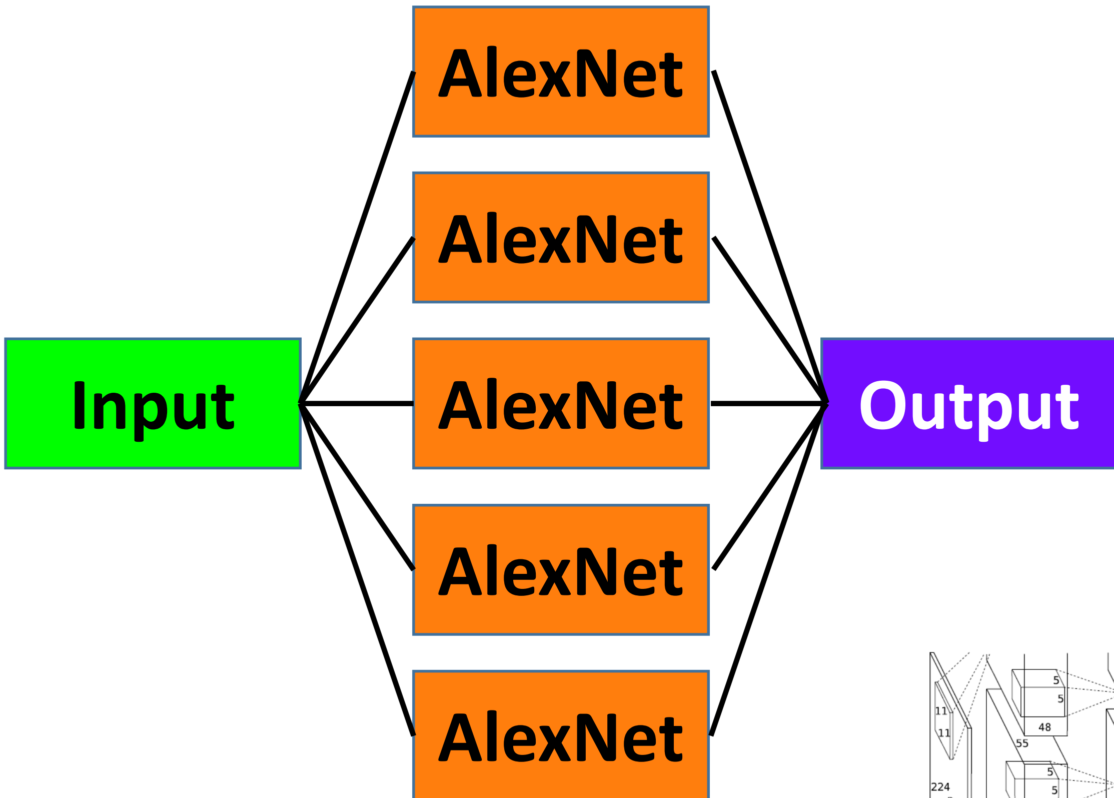


Subnetwork: multiple ReLU layers

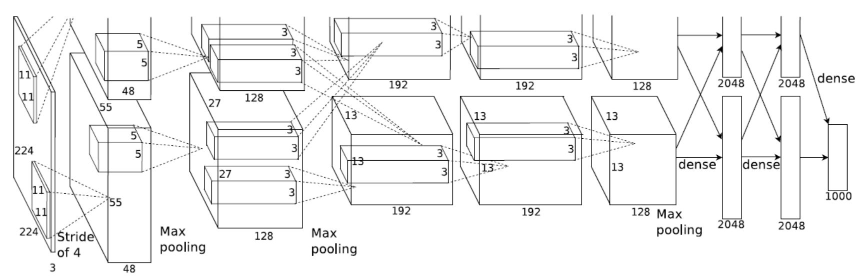


Key Property #2: Parallel Subnetworks

- **Example**: Parallel AlexNets [1]



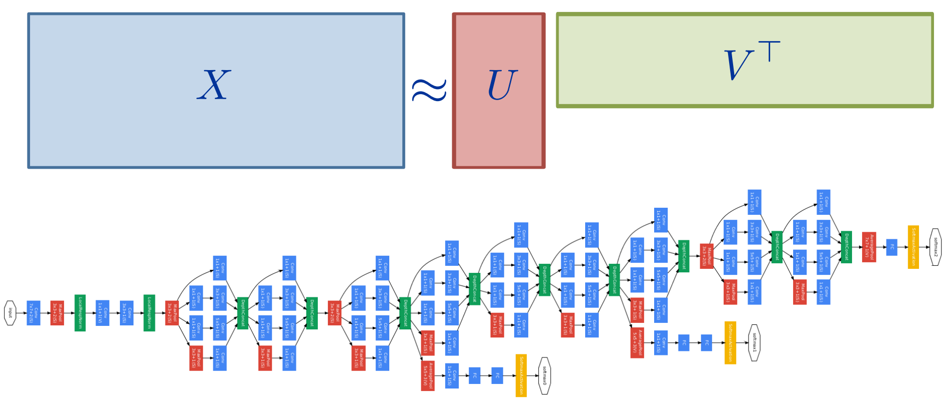
Subnetwork:
AlexNet



[1] Krizhevsky, Sutskever, and Hinton. "Imagenet classification with deep convolutional neural networks." NIPS, 2012

Outline

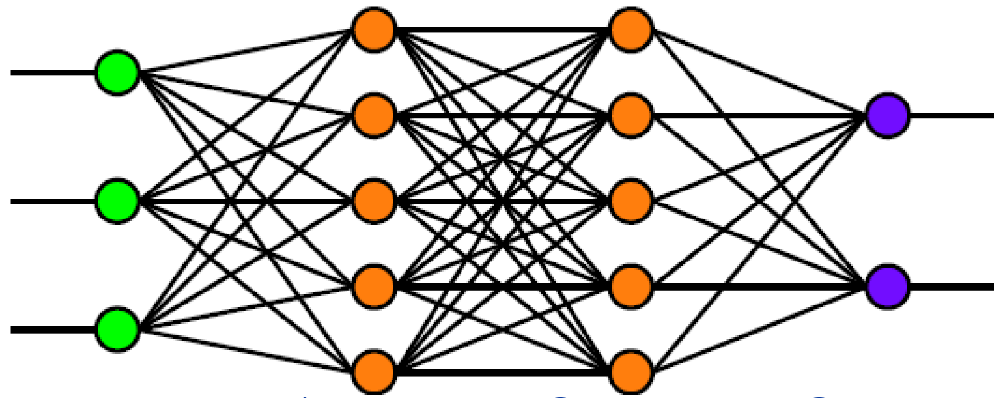
- **Architecture properties that facilitate optimization**
  - Positive homogeneity
  - Parallel subnetwork structure
- **Regularization properties that facilitate optimization**
  - Positive homogeneity
  - Adapt network structure to the data
- **Theoretical guarantees**
  - Sufficient conditions for global optimality
  - Local descent can reach global minimizers



[1] Haeffele, Young, Vidal. Structured Low-Rank Matrix Factorization: Optimality, Algorithm, and Applications to Image Processing, ICML ’14
[2] Haeffele, Vidal. Global Optimality in Tensor Factorization, Deep Learning and Beyond, arXiv, ’15
[3] Haeffele, Vidal. Global optimality in neural network training. CVPR 2017.

Basic Regularization: Weight Decay

$$\Theta(W^1, W^2, W^3) = \|W^1\|_F^2 + \|W^2\|_F^2 + \|W^3\|_F^2$$



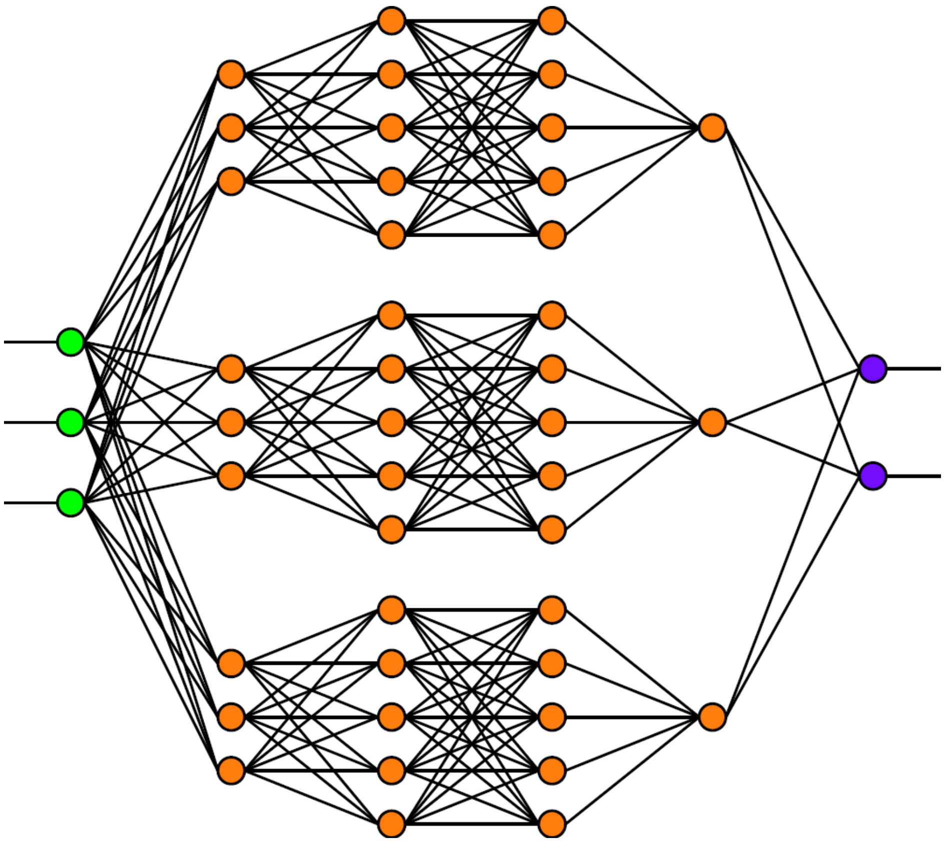
$$W^1 \qquad W^2 \qquad W^3$$

$$\Theta(\alpha W^1, \alpha W^2, \alpha W^3) = \alpha^2 \Theta(W^1, W^2, W^3)$$
$$\Phi(\alpha W^1, \alpha W^2, \alpha W^3) = \alpha^3 \Phi(W^1, W^2, W^3)$$

- **Proposition** non-matching degrees => spurious local minima

Regularizer Adapted to Network Size

- Start with a positively homogeneous network with parallel structure



Regularizer Adapted to Network Size

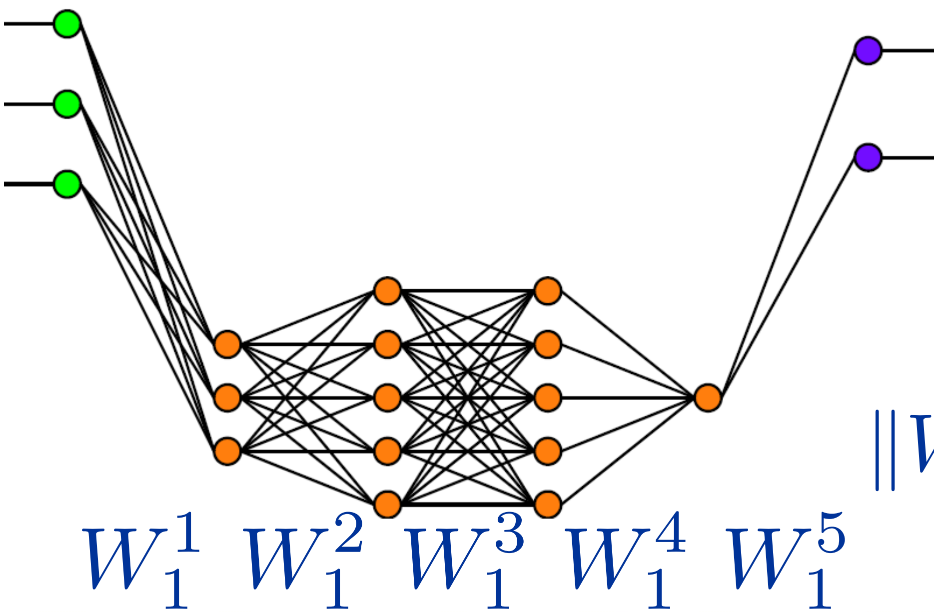
- Take the weights of one subnetwork and define a regularizer as $\theta(W_1^1, W_1^2, W_1^3, W_1^4, W_1^5)$ with the properties:
  - Positive semi-definite
  - Positively homogeneous with the same degree as network

$$\Phi(\alpha W) = \alpha^p \Phi(W)$$

$$\theta(\alpha W) = \alpha^p \theta(W)$$

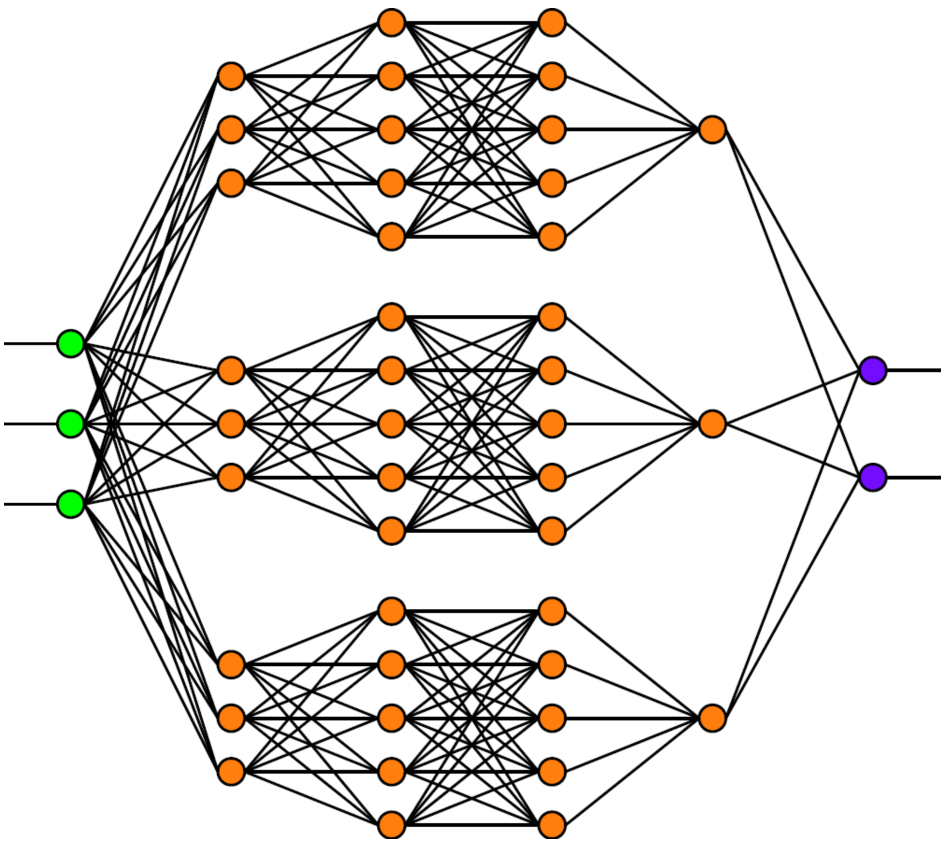
  - **Example:** product of norms

$$\|W_1^1\| \|W_1^2\| \|W_1^3\| \|W_1^4\| \|W_1^5\|$$



Regularizer Adapted to Network Size

- Sum over all subnetworks



$$\Theta(W) = \sum_{i=1}^{r} \theta(W^i)$$

$$r = \#\ \text{subnets}$$

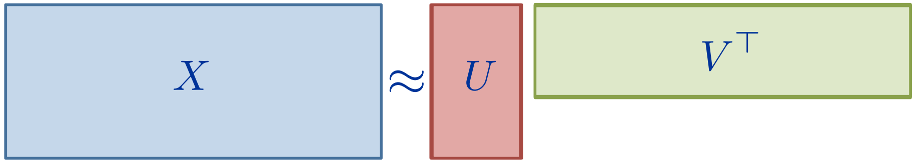
- Allow r to vary
- Adding a subnetwork is penalized by an additional term in the sum
- Regularizer constraints number of subnetworks

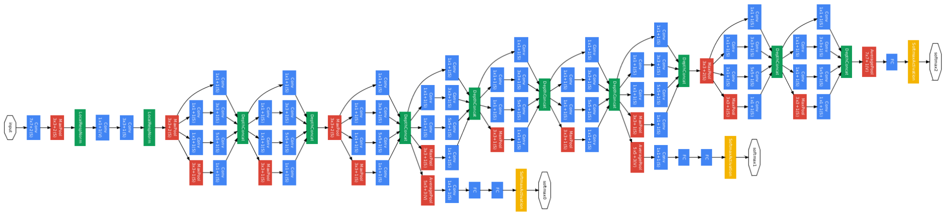
Outline

- **Architecture properties that facilitate optimization**
  - Positive homogeneity
  - Parallel subnetwork structure

- **Regularization properties that facilitate optimization**
  - Positive homogeneity
  - Adapt network structure to the data

- **Theoretical guarantees**
  - Sufficient conditions for global optimality
  - Local descent can reach global minimizers





[1] Haeffele, Young, Vidal. Structured Low-Rank Matrix Factorization: Optimality, Algorithm, and Applications to Image Processing, ICML '14
[2] Haeffele, Vidal. Global Optimality in Tensor Factorization, Deep Learning and Beyond, arXiv, '15
[3] Haeffele, Vidal. Global optimality in neural network training. CVPR 2017.

JOHNS HOPKINS
MATHEMATICAL INSTITUTE
for DATA SCIENCE

Main Results: Matrix Factorization

- **Convex formulations:** **Factorized formulations**

$$\min_X \ell(Y, X) + \lambda \|X\|_* \qquad \min_{U,V} \ell(Y, UV^\top) + \lambda \boxed{\Theta(U, V)}$$

- Variational form of the nuclear norm [1,2]

$$\boxed{\|X\|_*} = \min_{U,V} \boxed{\sum_{i=1}^{r} \|U_i\|_2 \|V_i\|_2} \quad \text{s.t.} \quad UV^\top = X$$

$$\boxed{\|X\|_* = \sum \sigma_i(X)}$$

- A natural generalization is the projective tensor norm [3,4]

$$\|X\|_{u,v} = \min_{U,V} \sum_{i=1}^{r} \|U_i\|_u \|V_i\|_v \quad \text{s.t.} \quad UV^\top = X$$

[1] Burer, Monteiro. Local minima and convergence in low- rank semidefinite programming. Math. Prog., 2005.
[2] Cabral, De la Torre, Costeira, Bernardino, "Unifying nuclear norm and bilinear factorization approaches for low-rank matrix decomposition," CVPR, 2013, pp. 2488–2495.
[3] Bach, Mairal, Ponce, Convex sparse matrix factorizations, arXiv 2008.
[4] Bach. Convex relaxations of structured matrix factorizations, arXiv 2013.

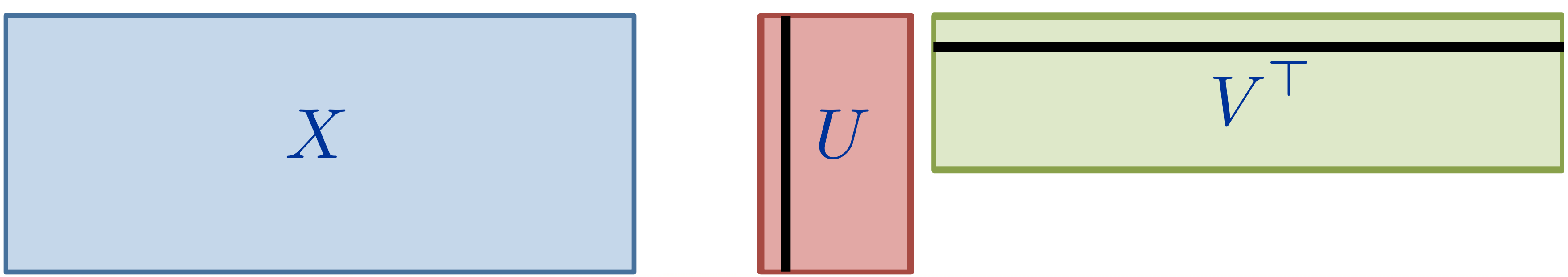
Main Results: Matrix Factorization

- **Theorem 1**: Assume ℓ is convex and once differentiable in X. A local minimizer (U, V) of the non-convex factorized problem

$$\min_{U,V} \ell(Y, UV^\top) + \lambda \sum_{i=1}^{r} \|U_i\|_u \|V_i\|_v$$

such that for some i $U_i = V_i = 0$, is a global minimizer. Moreover, $UV^\top$ is a global minimizer of the convex problem

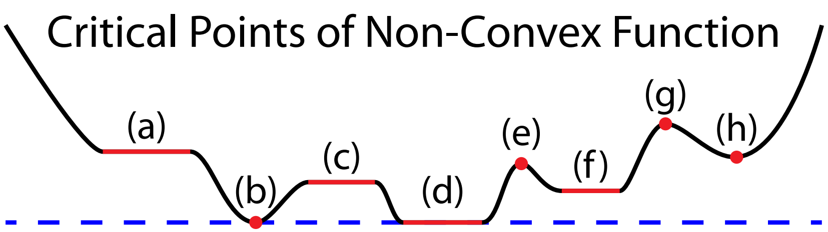
$$\min_{X} \ell(Y, X) + \lambda \|X\|_{u,v}$$

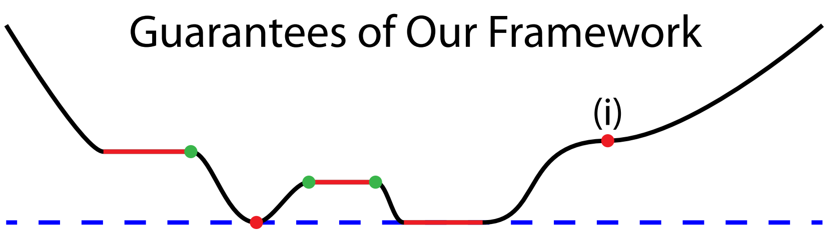


[1] Haeffele, Young, Vidal. Structured Low-Rank Matrix Factorization: Optimality, Algorithm, and Applications to Image Processing, ICML '14
[2] Haeffele, Vidal. Global Optimality in Tensor Factorization, Deep Learning and Beyond, arXiv '15

Main Results: Matrix Factorization

- **Theorem 2:** If the number of columns is large enough, local descent can reach a global minimizer from any initialization





- **Meta-Algorithm:**
  - If not at a local minima, perform local descent
  - At local minima, test if Theorem 1 is satisfied. If yes => global minima
  - If not, increase size of factorization and find descent direction (u,v)

$$r \leftarrow r+1 \quad U \leftarrow \begin{bmatrix} U & u \end{bmatrix} \quad V \leftarrow \begin{bmatrix} V & v \end{bmatrix}$$

[1] Haeffele, Vidal. Global Optimality in Tensor Factorization, Deep Learning and Beyond, arXiv '15

Example: Low-Rank Matrix Factorization

- **Convex formulation** of low-rank matrix approximation based on nuclear norm minimization admits closed form solution

$$\min_X \frac{1}{2}\|Y - X\|_F^2 + \lambda \|X\|_*$$

$$\Downarrow$$

$$Y = U\Sigma V^\top$$

$$X^* = U\mathcal{S}_\lambda(\Sigma)V^\top$$

Shrink singular values by lambda

- r = rank (X*) = number of singular values above lambda

Example: Low-Rank Matrix Factorization

- **Factorized formulation** of low-rank matrix approximation

$$\min_{U,V,r} \frac{1}{2}\|Y - UV^\top\|_F^2 + \lambda \sum_{i=1}^{r} \|U_i\|_2 \|V_i\|_2$$

- **For fixed r:** perform alternating proximal gradient

$$U_i \leftarrow U_i - \eta_u \mathcal{S}_{\lambda \|V_i\|_2}\left(\nabla_{U_i} \ell(Y, UV^\top)\right)$$
$$V_i \leftarrow V_i - \eta_v \mathcal{S}_{\lambda \|U_i\|_2}\left(\nabla_{V_i} \ell(Y, UV^\top)\right)$$

- **Check if r needs to be increased:** solve polar problem

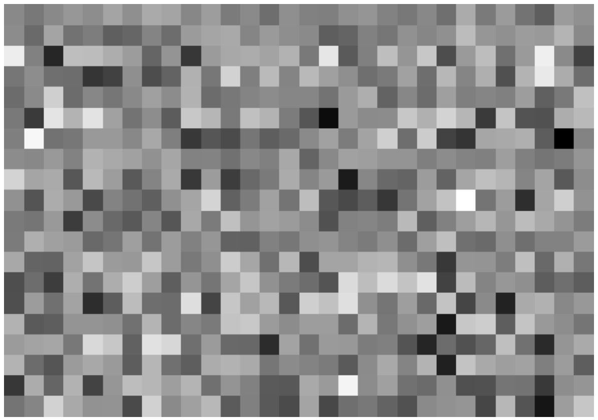
$$\min_{u,v} u^\top \nabla_X \ell(Y, UV^\top) v \quad \text{s.t.} \quad \|u\|_2 \|v\|_2 \leq 1$$

Shrink columns by lambda

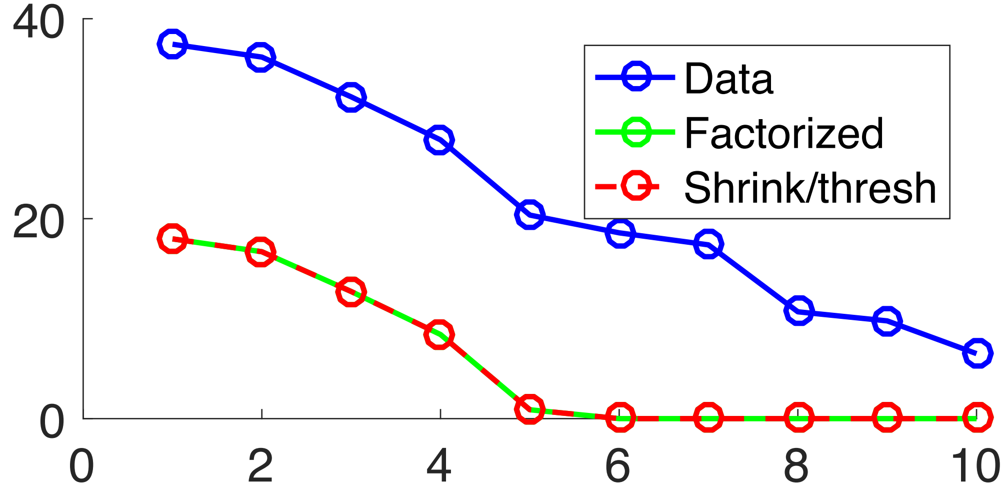
  - IF polar >= - lambda THEN stop; ELSE (u,v) gives descent direction

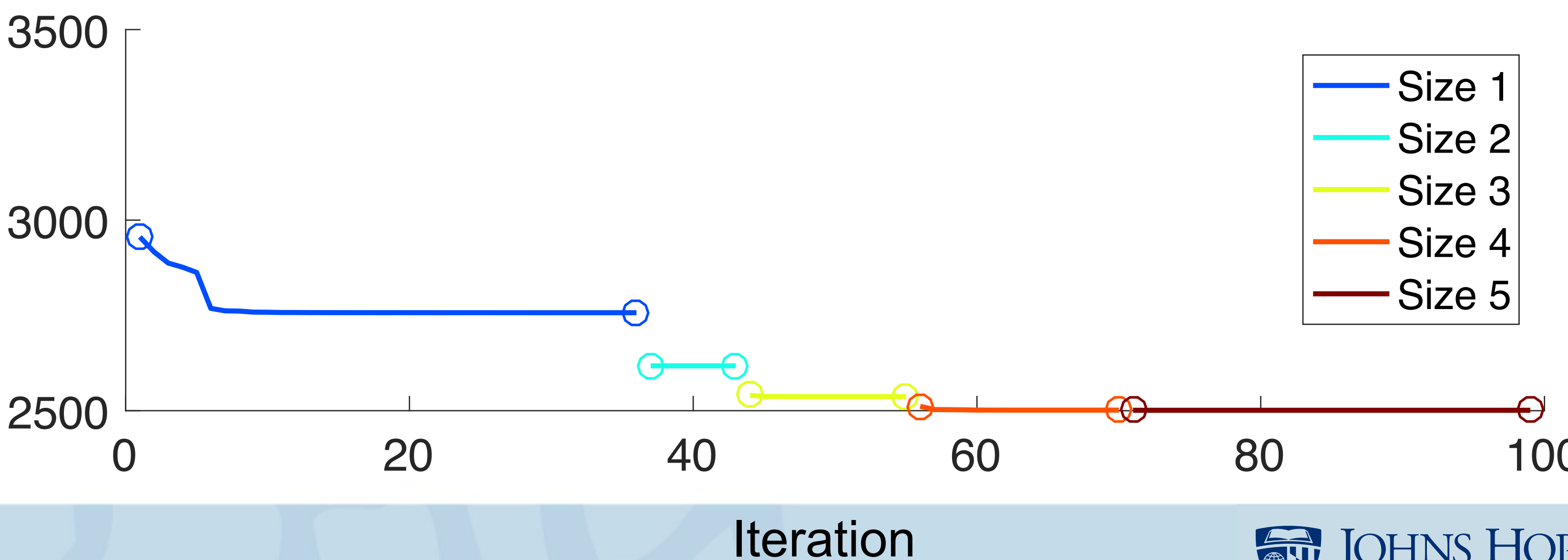
Example: Low-Rank Matrix Factorization

Synthetic data

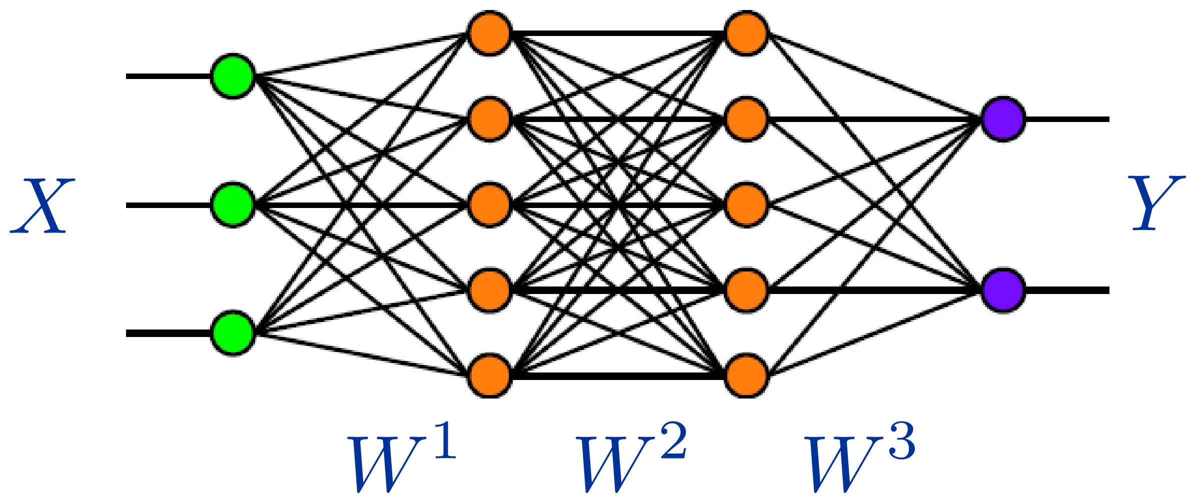


Singular values





From Matrix Factorization to Deep Learning



$$\Phi(X, W^1, \ldots, W^K) = \psi_K(\cdots \psi_2(\psi_1(XW^1)W^2) \cdots W^K)$$

output activation input weights

Main Results: Tensor Fact. & Deep Learning

- In matrix factorization we had "generalized nuclear norm"

$$\|Z\|_{u,v} = \min_{U,V,r} \sum_{i=1}^{r} \|U_i\|_u \|V_i\|_v \quad \text{s.t.} \quad UV^\top = Z$$

- By analogy we define "nuclear deep net regularizer"

$$\Omega_{\phi,\theta}(Z) = \min_{\{W^k\},r} \sum_{i=1}^{r} \theta(W_i^1, \dots, W_i^K) \text{ s.t. } \Phi(W_i^1, \dots, W_i^K) = Z$$

where θ is positively homogeneous of the same degree as ϕ

- **Proposition:** $\Omega_{\phi,\theta}$ is convex
- **Intuition:** regularizer Θ "comes from a convex function"

Main Results: Tensor Fact. & Deep Learning

$$\min_{\{W^k\}_{k=1}^K} \ell(Y, \Phi(X, W^1, \dots, W^K)) + \lambda \Theta(W^1, \dots, W^K)$$

- **Assumptions:**
  - $\ell(Y, Z)$: convex and once differentiable in Z
  - Φ and Θ: sums of positively homogeneous functions of same degree

$$\phi(\alpha W_i^1, \dots, \alpha W_i^K) = \alpha^p \phi(W_i^1, \dots, W_i^K) \quad \forall \alpha \geq 0$$

- **Theorem 1:** A local minimizer such that for some *i* and all k $W_i^k = 0$ is a global minimizer

- **Theorem 2:** If the size of the network is large enough, local descent can reach a global minimizer from any initialization

[1] Haeffele, Young, Vidal. Structured Low-Rank Matrix Factorization: Optimality, Algorithm, and Applications to Image Processing, ICML '14
[2] Haeffele, Vidal. Global Optimality in Tensor Factorization, Deep Learning and Beyond, arXiv, '15
[3] Haeffele, Vidal. Global optimality in neural network training. CVPR 2017.

Conclusions and Future Directions

- **Size matters**
  - Optimize not only the network weights, but also the network size
  - Today: size = number of neurons or number of parallel networks
  - Tomorrow: size = number of layers + number of neurons per layer
- **Regularization matters**
  - Use “positively homogeneous regularizer” of same degree as network
  - How to build a regularizer that controls number of layers + number of neurons per layer
- **Not done yet**
  - Checking if we are at a local minimum or finding a descent direction can be NP hard
  - Need “computationally tractable” regularizers

JHU vision lab

Global Optimality in Structured Matrix Factorization

Ben Haeffele and René Vidal
Center for Imaging Science
Mathematical Institute for Data Science
Johns Hopkins University

JOHNS HOPKINS MATHEMATICAL INSTITUTE for DATA SCIENCE

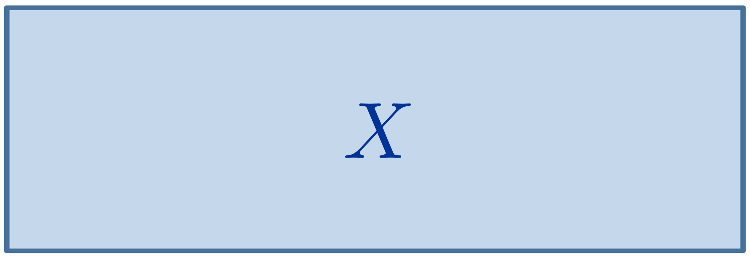
THE DEPARTMENT OF BIOMEDICAL ENGINEERING
The Whitaker Institute at Johns Hopkins

Center for IMAGING SCIENCE

Typical Low-Rank Formulations

- **Convex formulations**:

$$\min_{X} \ell(Y, X) + \lambda\Theta(X)$$



  - Low-rank matrix approximation
  - Low-rank matrix completion
  - Robust PCA

✓ Convex
✳ Large problem size
✳ Unstructured factors

- **Factorized formulations**:

$$\min_{U,V} \ell(Y, UV^\top) + \lambda\Theta(U, V)$$

  - Principal component analysis
  - Nonnegative matrix factorization
  - Sparse dictionary learning

✳ Non-Convex
✓ Small problem size
✓ Structured factors

Convex Formulations of Matrix Factorization

- **Convex formulations**:
  - ℓ, Θ : convex in X

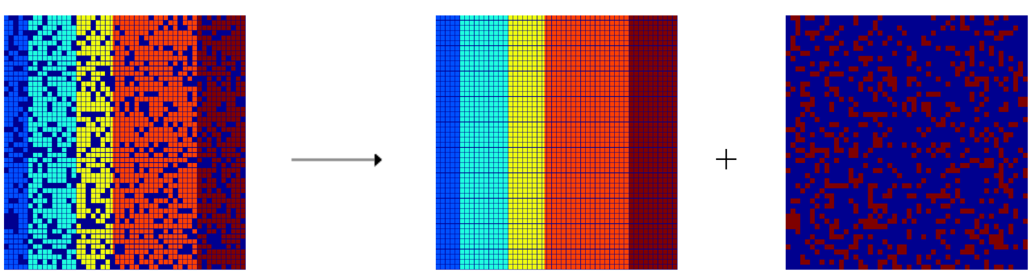
$$\boxed{\min_X \; \ell(Y, X) + \lambda \, \Theta(X)}$$

- Low-rank matrix approximation:

$$\min_X \; \frac{1}{2}\|Y - X\|_F^2 + \lambda \boxed{\|X\|_*} \qquad \boxed{\|X\|_* = \sum \sigma_i(X)}$$

- Robust PCA:

$$\min_X \; \|Y - X\|_1 + \lambda \, \|X\|_*$$



✓ Convex

✳ Large problem size

✳ Unstructured factors

Candès, Recht. Exact matrix completion via convex optimization. Foundations of Computational Mathematics, 2009.
Keshavan, Montanari, Oh. Matrix completion from a few entries. IEEE Transactions on Information Theory, 2010.
Candès, Tao. The power of convex relaxation: Near-optimal matrix completion. IEEE Transactions Information Theory, 2010
Candes, Li, Ma, Wright. Robust Principal Component Analysis? Journal of the ACM, 2011.
Xu, Caramanis, Sanghavi. Robust PCA via outlier pursuit. NIPS 2010

Factorized Formulations Matrix Factorization

- **Factorized formulations**:
  - $\ell(Y, X)$: convex in X

$$\min_{U,V} \ell(Y, UV^\top) + \lambda\Theta(U, V)$$

- PCA [1]: $\min_{U,V} \|Y - UV^\top\|_F^2 \quad \text{s.t.} \quad U^\top U = I$
- NMF [2]: $\min_{U,V} \|Y - UV^\top\|_F^2 \quad \text{s.t.} \quad U \geq 0, V \geq 0$
- SDL [3-5]: $\min_{U,V} \|Y - UV^\top\|_F^2 \quad \text{s.t.} \quad \|U_i\|_2 \leq 1, \|V_i\|_0 \leq r$

✓ Small problem size

✓ Structured factors

✳ Need to specify size a priori

✳ Non-convex optimization problem

[1] Jolliffe. Principal component analysis. Springer, 1986
[2] Lee, Seung. "Learning the parts of objects by non−negative matrix factorization." Nature, 1999
[3] Olshausen, Field. "Sparse coding with an overcomplete basis set: A strategy employed by v1?," Vision Research, 1997
[4] Engan, Aase, Hakon-Husoy, "Method of optimal directions for frame design," ICASSP 1999
[5] Aharon, Elad, Bruckstein, "K-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation", TSP 2006

Relating Convex & Factorized Formulations

- **Convex formulations:** **Factorized formulations**

$$\min_X \ell(Y,X) + \lambda\|X\|_* \qquad \min_{U,V} \ell(Y,UV^\top) + \lambda\Theta(U,V)$$

- Variational form of the nuclear norm [1,2]

$$\|X\|_* = \min_{U,V} \sum_{i=1}^r \|U_i\|_2\|V_i\|_2 \quad \text{s.t.} \quad UV^\top = X$$

$$\|X\|_* = \sum \sigma_i(X)$$

- A natural generalization is the projective tensor norm [3,4]

$$\|X\|_{u,v} = \min_{U,V} \sum_{i=1}^r \|U_i\|_u\|V_i\|_v \quad \text{s.t.} \quad UV^\top = X$$

[1] Burer and Monteiro. Local minima and convergence in low- rank semidefinite programming. Math. Prog., 103(3):427–444, 2005.
[2] R. Cabral, F. De la Torre, J. P. Costeira, and A. Bernardino, “Unifying nuclear norm and bilinear factorization approaches for low-rank matrix decomposition,” in IEEE International Conference on Computer Vision, 2013, pp. 2488–2495.
[3] Bach, Mairal, Ponce, Convex sparse matrix factorizations, arXiv 2008.
[4] Bach. Convex relaxations of structured matrix factorizations, arXiv 2013.

Main Results: Projective Tensor Norm Case

- **Theorem 1**: Assume ℓ is convex and once differentiable in X. A local minimizer (U, V) of the non-convex factorized problem

$$\min_{U,V} \ell(Y, UV^\top) + \lambda \sum_{i=1}^{r} \|U_i\|_u \|V_i\|_v$$

such that for some i $U_i = V_i = 0$, is a global minimizer. Moreover, $UV^\top$ is a global minimizer of the convex problem

$$\min_{X} \ell(Y, X) + \lambda \|X\|_{u,v}$$

- **Proof sketch**:
  - Convex problem gives global lower bound for non-convex problem
  - If (U, V) local min. of non-convex, then $UV^\top$ global min. of convex

[1] Haeffele, Young, Vidal. Structured Low-Rank Matrix Factorization: Optimality, Algorithm, and Applications to Image Processing, ICML '14
[2] Haeffele, Vidal. Global Optimality in Tensor Factorization, Deep Learning and Beyond, arXiv '15

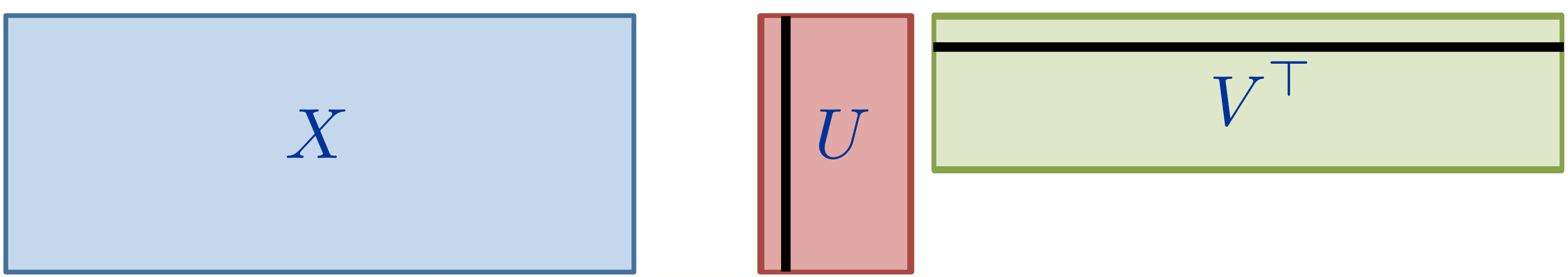
Main Results: Projective Tensor Norm Case

- **Theorem 1**: Assume ℓ is convex and once differentiable in X. A local minimizer (U, V) of the non-convex factorized problem

$$\min_{U,V} \ell(Y, UV^\top) + \lambda \sum_{i=1}^{r} \|U_i\|_u \|V_i\|_v$$

such that for some i $U_i = V_i = 0$, is a global minimizer. Moreover, $UV^\top$ is a global minimizer of the convex problem

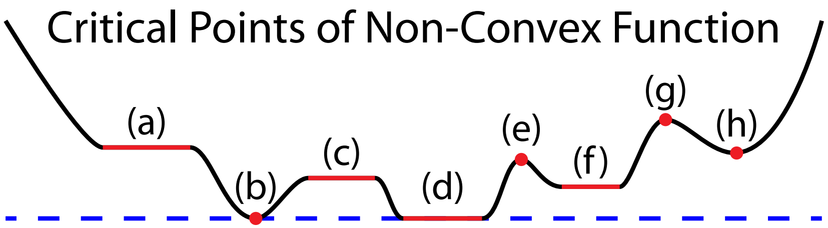
$$\min_{X} \ell(Y, X) + \lambda \|X\|_{u,v}$$

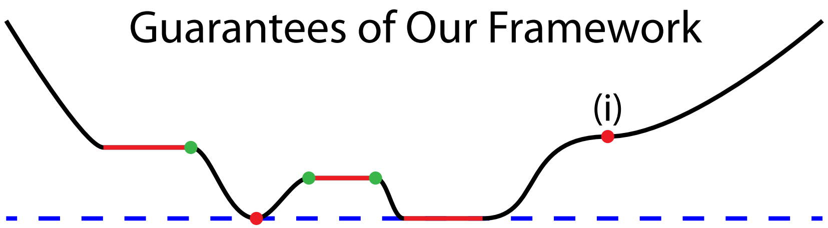


[1] Haeffele, Young, Vidal. Structured Low-Rank Matrix Factorization: Optimality, Algorithm, and Applications to Image Processing, ICML '14
[2] Haeffele, Vidal. Global Optimality in Tensor Factorization, Deep Learning and Beyond, arXiv '15

Main Results: Projective Tensor Norm Case

- **Theorem 2:** If the number of columns is large enough, local descent can reach a global minimizer from any initialization





- **Meta-Algorithm:**
  - If not at a local minima, perform local descent
  - At local minima, test if Theorem 1 is satisfied. If yes => global minima
  - If not, increase size of factorization and find descent direction (u,v)

$$r \leftarrow r+1 \quad U \leftarrow \begin{bmatrix} U & u \end{bmatrix} \quad V \leftarrow \begin{bmatrix} V & v \end{bmatrix}$$

[1] Haeffele, Vidal. Global Optimality in Tensor Factorization, Deep Learning and Beyond, arXiv '15

Example: Low-Rank Matrix Factorization

- **Convex formulation** of low-rank matrix approximation based on nuclear norm minimization admits closed form solution

$$\min_X \frac{1}{2}\|Y - X\|_F^2 + \lambda\|X\|_*$$

$$\Downarrow$$

$$Y = U\Sigma V^\top$$

$$X^* = U\mathcal{S}_\lambda(\Sigma)V^\top$$

Shrink singular values by lambda

- r = rank (X*) = number of singular values above lambda

Example: Low-Rank Matrix Factorization

- **Factorized formulation** of low-rank matrix approximation

$$\min_{U,V,r} \frac{1}{2}\|Y - UV^\top\|_F^2 + \lambda \sum_{i=1}^{r} \|U_i\|_2 \|V_i\|_2$$

- **For fixed r:** perform alternating proximal gradient

$$U_i \leftarrow U_i - \eta_u \mathcal{S}_{\lambda\|V_i\|_2}\left(\nabla_{U_i} \ell(Y, UV^\top)\right)$$
$$V_i \leftarrow V_i - \eta_v \mathcal{S}_{\lambda\|U_i\|_2}\left(\nabla_{V_i} \ell(Y, UV^\top)\right)$$

- **Check if r needs to be increased:** solve polar problem

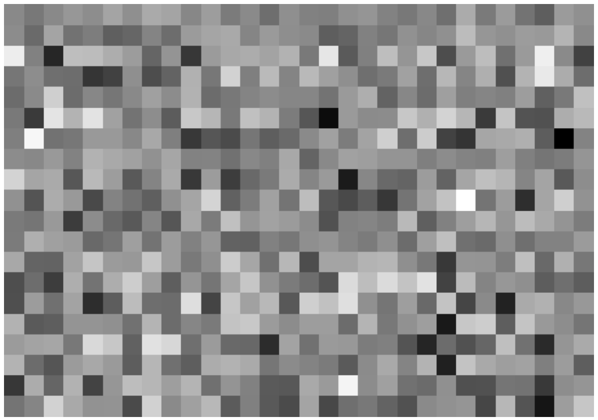
$$\min_{u,v} u^\top \nabla_X \ell(Y, UV^\top) v \quad \text{s.t.} \quad \|u\|_2 \|v\|_2 \leq 1$$

Shrink columns by lambda

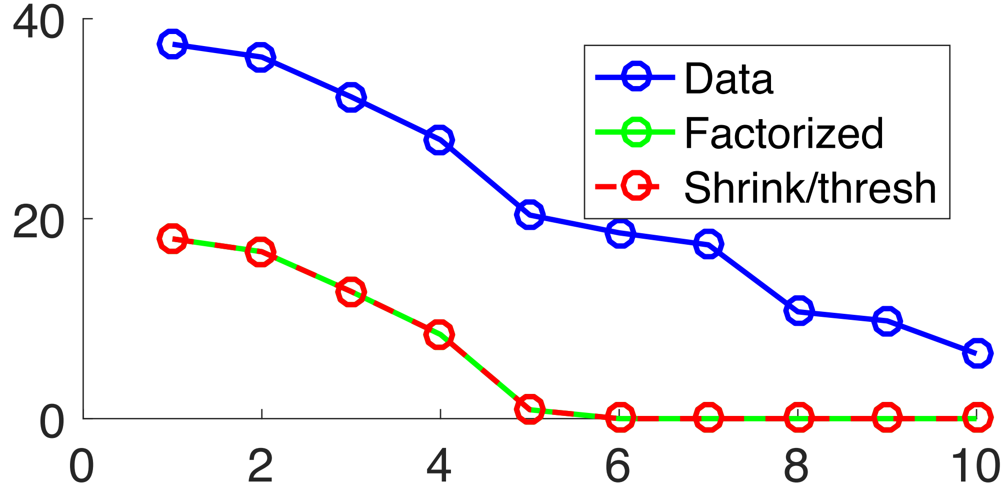
  – IF polar >= - lambda THEN stop; ELSE (u,v) gives descent direction

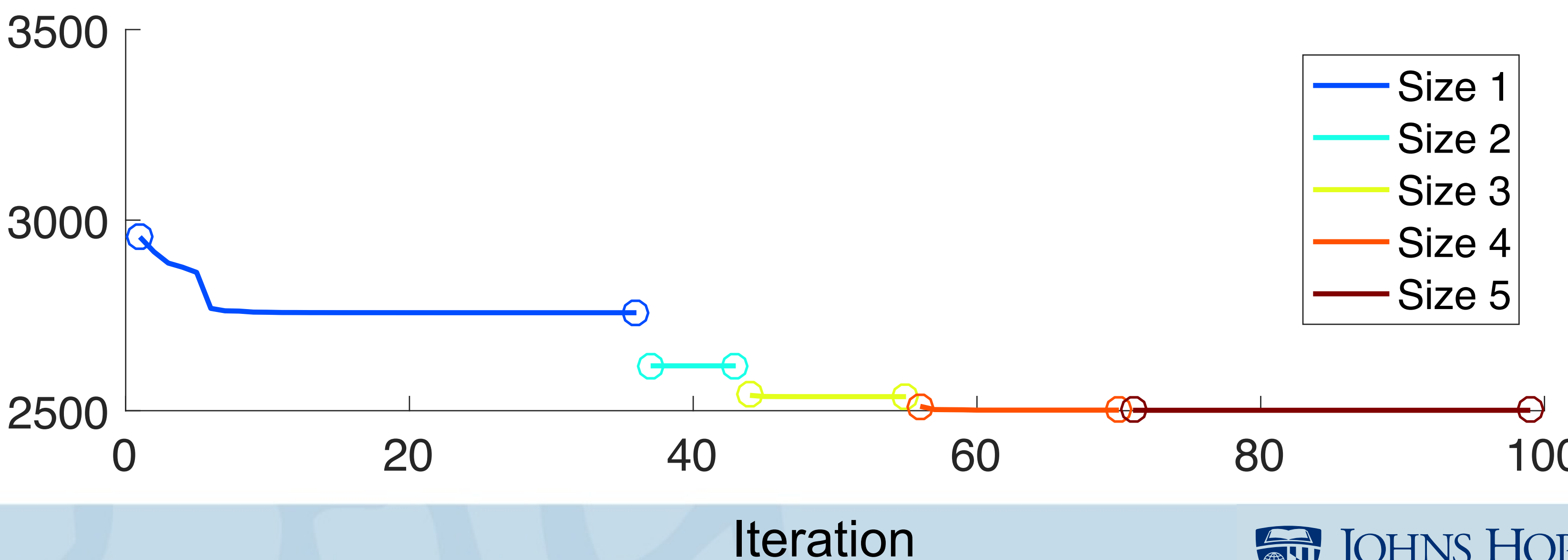
Example: Low-Rank Matrix Factorization

Synthetic data



Singular values





Main Results: Homogeneous Regularizers

$$\min_{U,V} \ell(Y, UV^\top) + \lambda\Theta(U,V)$$

- Theorems are also true for Θ = sum of positive semi-definite and positively homogeneous regularizers of degree 2

$$\Theta(U,V) = \sum_{i=1}^{r} \theta(U_i, V_i), \quad \theta(\alpha u, \alpha v) = \alpha^2 \theta(u,v), \forall \alpha \geq 0$$

- Examples

Product of norms

$$\theta(u,v) = \|u\| \|v\|$$

Conic constraints

$$u, v \geq 0$$

- Such regularizers on (U,V) induce a convex regularizer on X

$$\Omega_\theta(X) = \inf_{U,V} \ \Theta(U,V) \quad \text{s.t.} \quad X = UV^\top$$

B. Haeffele, E. Young, R. Vidal. Structured Low-Rank Matrix Factorization: Optimality, Algorithm, and Applications to Image Processing. ICML 2014
Benjamin D. Haeffele, Rene Vidal. Global Optimality in Tensor Factorization, Deep Learning, and Beyond. arXiv:1506.07540, 2015

Example: Nonnegative Matrix Factorization

- Original formulation

$$\min_{U,V} \|Y - UV^\top\|_F^2 \quad \text{s.t.} \quad U \geq 0, V \geq 0$$

- New factorized formulation

$$\min_{U,V,r} \|Y - UV^\top\|_F^2 + \lambda \sum_{i=1}^{r} \|U_i\|_2 \|V_i\|_2 \quad \text{s.t.} \quad U, V \geq 0$$

  – **Note:** regularization limits the number of columns in (U,V)

Example: Sparse Dictionary Learning

- Original formulation

$$\min_{U,V} \|Y - UV^\top\|_F^2 \quad \text{s.t.} \quad \|U_i\|_2 \leq 1, \|V_i\|_0 \leq r$$

- New factorized formulation

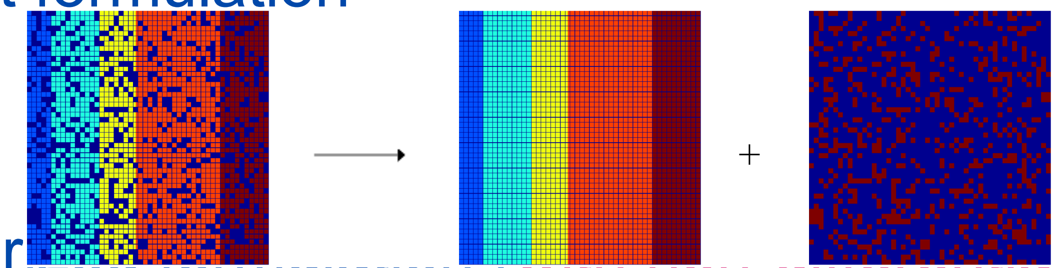
$$\min_{U,V} \|Y - UV^\top\|_F^2 + \lambda \sum_i |U_i|_2(|V_i|_2 + \gamma |V_i|_1)$$

Example: Robust PCA

- Original formulation [1]

$$\min_{X,E} \|E\|_1 + \lambda \|X\|_* \quad \text{s.t.} \quad Y = X + E$$

- Equivalent formulation



- New factorized formulation (with non-differentiable loss)

$$\min_{U,V} \|Y - UV^\top\|_1 + \lambda \sum_i |U_i|_2 |V_i|_2$$

- New factorized formulation (with differentiable loss)

$$\min_{U,V,E} \|E\|_1 + \lambda \sum_i |U_i|_2 |V_i|_2 + \frac{\gamma}{2} \|Y - UV - E\|_F^2$$

[1] Candes, Li, Ma, Wright. Robust Principal Component Analysis? Journal of the ACM, 2011.

JHU vision lab

Global Optimality in Positively Homogeneous Factorization

Ben Haeffele and René Vidal

Center for Imaging Science
Mathematical Institute for Data Science
Johns Hopkins University

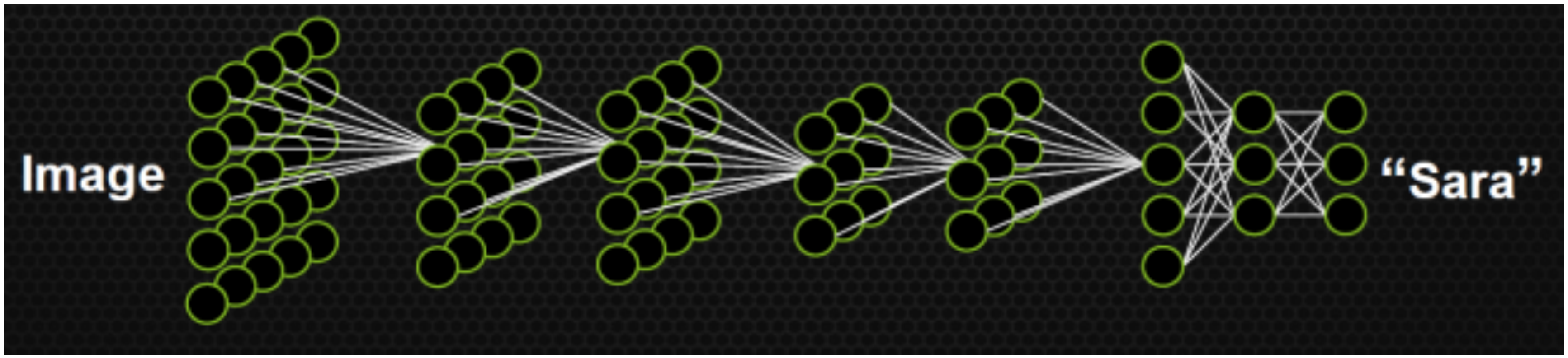
JOHNS HOPKINS
MATHEMATICAL INSTITUTE
for DATA SCIENCE

THE DEPARTMENT OF BIOMEDICAL ENGINEERING
The Whitaker Institute at Johns Hopkins

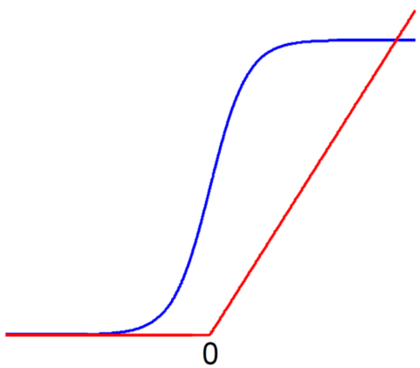
Center for IMAGING SCIENCE

Learning Problem for Neural Networks

- The learning problem is non-convex



$$\Phi(X^1, \quad \psi_K(\cdots \psi_2(\psi_1(VX^1)X^2)\cdots X^K)$$



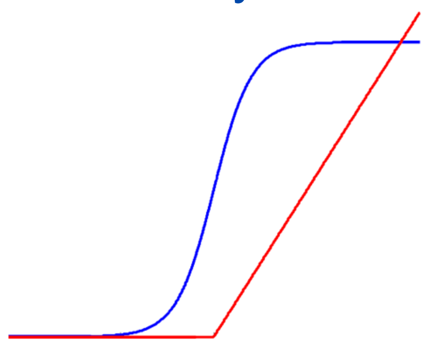
nonlinearity features weights

$$\min_{X^1,\ldots,X^K} \ell(Y, \Phi(X^1,\ldots,X^K)) + \lambda\Theta(X^1,\ldots,X^K)$$

loss labels regularizer

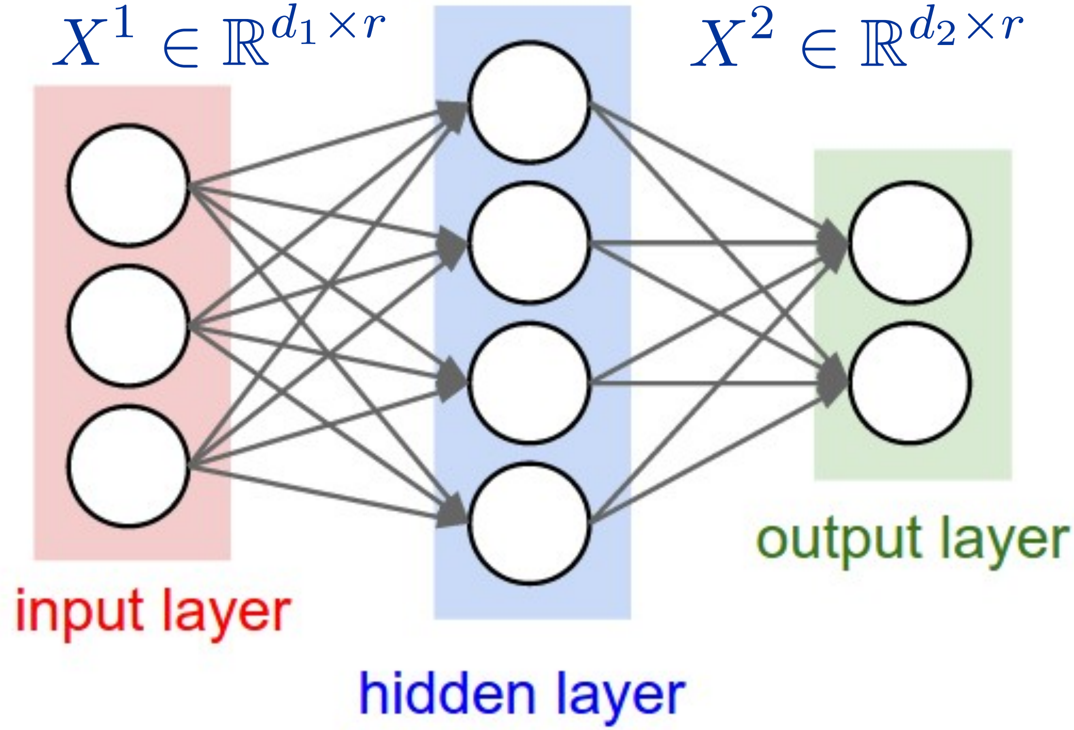
From Matrix Factorizations to Deep Learning

- Two-layer NN
  - Input: $V \in \mathbb{R}^{N \times d_1}$
  - Weights: $X^k \in \mathbb{R}^{d_k \times r}$
  - Nonlinearity: ReLU



$$\psi_1(x) = \max(x, 0)$$

$X^1 \in \mathbb{R}^{d_1 \times r}$ $X^2 \in \mathbb{R}^{d_2 \times r}$

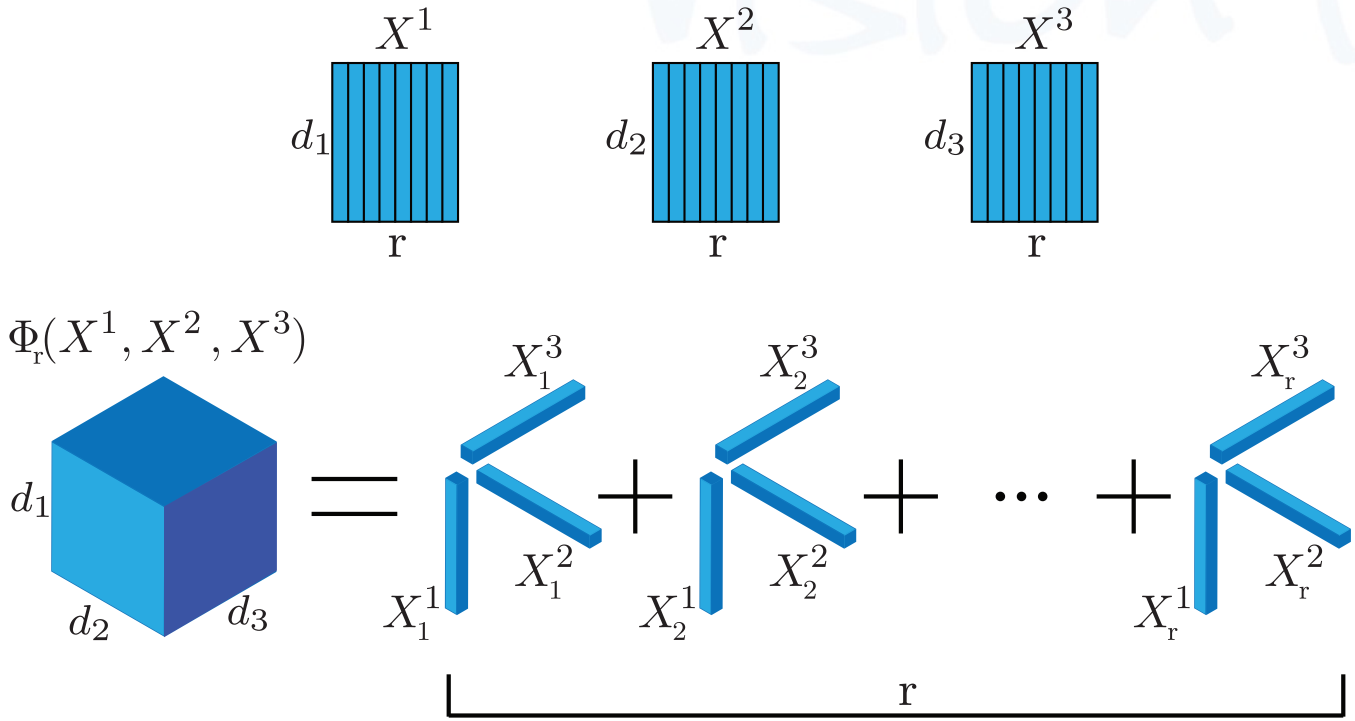


- "Almost" like matrix factorization
  - r = rank
  - r = #neurons in hidden layer
  - ReLU + max pooling is positively homogeneous of degree 1

$$\Phi(X^1, X^2) = \psi_1(V X^1)(X^2)^\top$$

JOHNS HOPKINS
MATHEMATICAL INSTITUTE
for DATA SCIENCE

From Matrix to Tensor Factorization



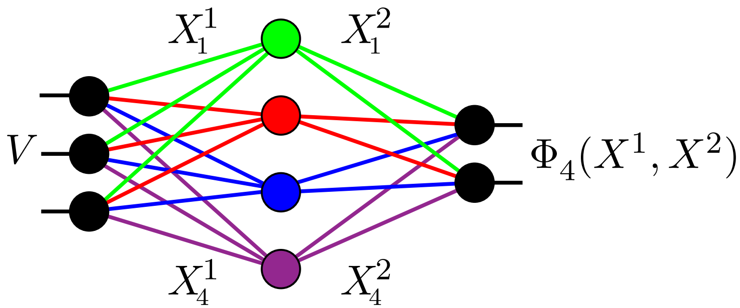
- Tensor product $\phi(X^1, \ldots, X^K) = X^1 \otimes \cdots \otimes X^K$ is positively homogeneous of degree K

$$\Phi(X^1, \ldots, X^K) = \sum_{i=1}^{r} \phi(X_i^1, \ldots, X_i^K)$$

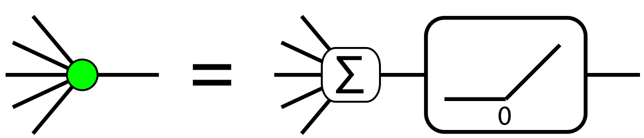
From Matrix Factorizations to Deep Learning

$$\Phi(X^1, \ldots, X^K) = \sum_{i=1}^{r} \phi(X_i^1, \ldots, X_i^K)$$

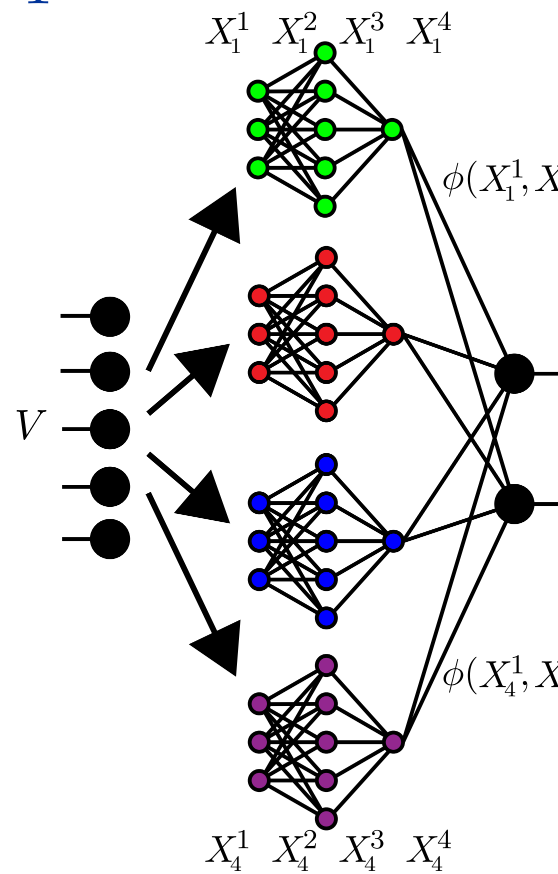
ReLU Network with One Hidden Layer



Rectified Linear Unit (ReLU)



Multilayer ReLU Parallel Network



Key Ingredient: Proper Regularization

- In matrix factorization we had "generalized nuclear norm"

$$\|X\|_{u,v} = \min_{U,V} \sum_{i=1}^{r} \|U_i\|_u \|V_i\|_v \quad \text{s.t.} \quad UV^\top = X$$

- By analogy we define "nuclear deep net regularizer"

$$\Omega_{\phi,\theta}(X) = \min_{\{X^k\}} \sum_{i=1}^{r} \theta(X_i^1, \ldots, X_i^K) \text{ s.t. } \Phi(X^1, \ldots, X^K) = X$$

where θ is positively homogeneous of the same degree as ϕ

- **Proposition:** $\Omega_{\phi,\theta}$ is convex
- **Intuition:** regularizer Θ "comes from a convex function"

JOHNS HOPKINS
MATHEMATICAL INSTITUTE
for DATA SCIENCE

Main Results

- **Theorem 1:** Assume ℓ is convex and once differentiable in X. A local minimizer $(X^1, \ldots, X^K)$ of the factorized formulation

$$\min_{\{X^k\}} \ell\Big(Y, \sum_{i=1}^{r} \phi(X_i^1, \ldots, X_i^K)\Big) + \lambda \sum_{i=1}^{r} \theta(X_i^1, \ldots, X_i^K)$$

such that for some i and all k $X_i^k = 0$ is a global minimizer. Moreover, $X = \Phi(X^1, \ldots, X^K)$ is a global minimizer of the convex problem

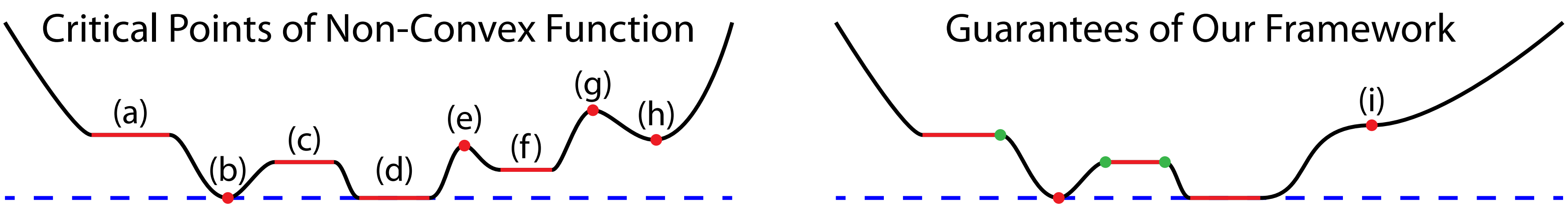
$$\min_{X} \ell(Y, X) + \lambda \Omega_{\phi,\theta}(X)$$

- Examples
  - Matrix factorization
  - Tensor factorization
  - Deep learning

[1] Haeffele, Young, Vidal. Structured Low-Rank Matrix Factorization: Optimality, Algorithm, and Applications to Image Processing, ICML '14
[2] Haeffele, Vidal. Global Optimality in Tensor Factorization, Deep Learning and Beyond, arXiv, '15
[3] Haeffele, Vidal. Global optimality in neural network training. CVPR 2017.

Main Results

- **Theorem 2:** If the size of the network is large enough, local descent can reach a global minimizer from any initialization



- **Meta-Algorithm:**
  - If not at a local minima, perform local descent
  - At a local minima, test if Theorem 1 is satisfied. If yes => global minima
  - If not, increase size by 1 (add network in parallel) and continue
  - Maximum r guaranteed to be bounded by the dimensions of the network output

[1] Haeffele, Vidal. Global Optimality in Tensor Factorization, Deep Learning and Beyond, arXiv '15

Experimental Results

- Better performance with less training examples [Sokolic, Giryes, Sapiro, Rodrigues, 2017]
  - WD = weight decay
  - LM = Jacobian regularizer ~ product of weights regularizer

		256 samples			512 samples			1024 samples		
loss	# layers	no reg.	WD	LM	no reg.	WD	LM	no reg.	WD	LM
hinge	2	88.37	89.88	**93.83**	93.99	94.62	95.49	95.79	96.57	97.45
hinge	3	87.22	89.31	93.22	93.41	93.97	**95.76**	95.46	96.45	**97.60**
CCE	2	88.45	88.45	92.77	92.29	93.14	95.25	95.38	95.79	96.89
CCE	3	89.05	89.05	93.10	91.81	93.02	95.32	95.11	95.86	97.14

[Sokolic, Giryes, Sapiro, Rodrigues, 2017]

JOHNS HOPKINS
MATHEMATICAL INSTITUTE
for DATA SCIENCE

Conclusions and Future Directions

- **Size matters**
  - Optimize not only the network weights, but also the network size
  - Today: size = number of neurons or number of parallel networks
  - Tomorrow: size = number of layers + number of neurons per layer
- **Regularization matters**
  - Use “positively homogeneous regularizer” of same degree as network
  - How to build a regularizer that controls number of layers + number of neurons per layer
- **Not done yet**
  - Checking if we are at a local minimum or finding a descent direction can be NP hard
  - Need “computationally tractable” regularizers

Dropout as a Low-Rank Regularizer for Matrix Factorization

J. Cavazza*, B. Haeffele*, C. Lane, P. Morerio, V. Murino, and R. Vidal
Mathematical Institute for Data Science, Johns Hopkins University, USA
Istituto Italiano di Tecnologia, Genoa, Italy

Backpropagation vs Dropout Training

- Minimize empirical loss

$$\min_W \frac{1}{N} \sum_{j=1}^{N} \ell(Y_j, \Phi(X_j, W))$$

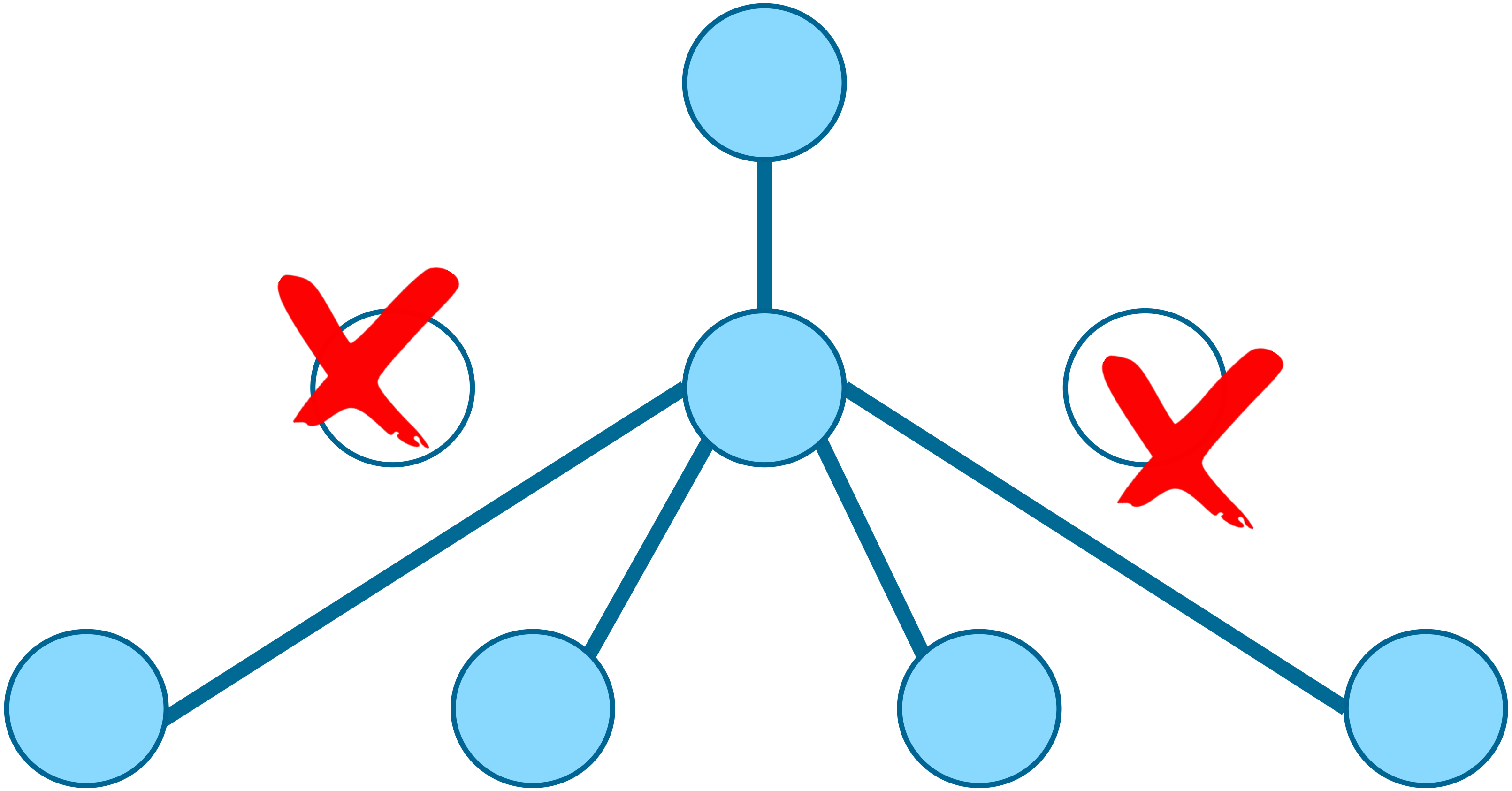
- Backpropagation with stochastic gradient descent (SGD)

$$W^{t+1} = W^t - \frac{\epsilon}{|\mathcal{B}|} \sum_{j \in \mathcal{B}} \nabla \ell\big(Y_j, \Phi(X_j, W^t)\big)$$

- Backpropagation with dropout $z_k \sim \mathrm{Ber}(\theta), \quad \theta \in (0,1)$

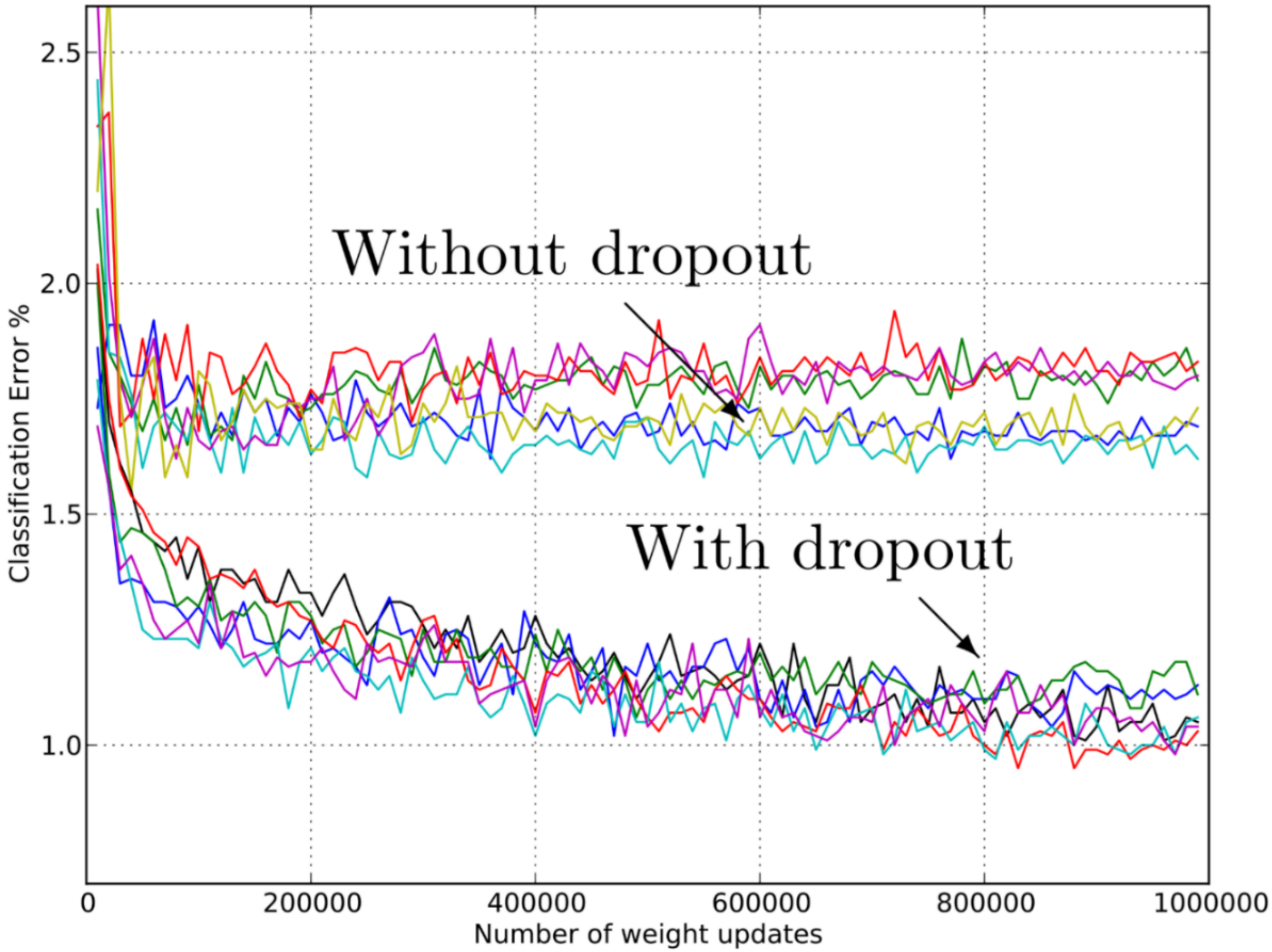
$$W^{t+1} = W^t - \frac{\epsilon}{|\mathcal{B}|} \sum_{j \in \mathcal{B}} \nabla \ell\big(Y_j, \underbrace{\Phi(X_j, W^t, \boldsymbol{z})}_{\text{set output of drop out neurons to 0}}\big) \otimes \underbrace{\boldsymbol{z}}_{\text{set gradient of dropout neurons to 0}}$$

Dropout Training



Srivastava et al. – Dropout: A simple way to prevent neural networks from overfitting – JMLR 2014

Dropout Training: Better Learning Curve



Srivastava et al. – Dropout: A simple way to prevent neural networks from overfitting – JMLR 2014

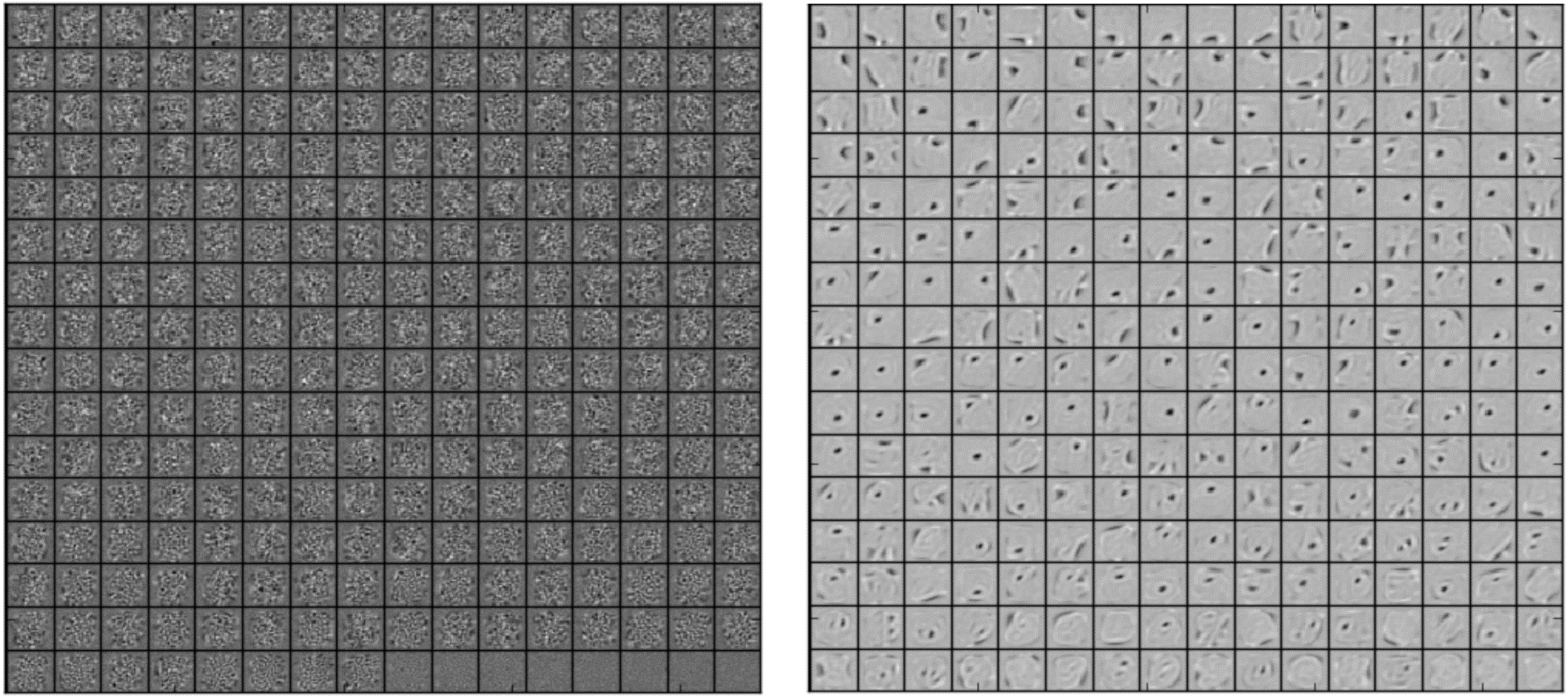
Dropout Training: Better Performance

Method	Test Classification error %
L2	1.62
L2 + L1 applied towards the end of training	1.60
L2 + KL-sparsity	1.55
Max-norm	1.35
Dropout + L2	1.25
Dropout + Max-norm	**1.05**

Table 9: Comparison of different regularization methods on MNIST.

Srivastava et al. – Dropout: A simple way to prevent neural networks from overfitting – JMLR 2014

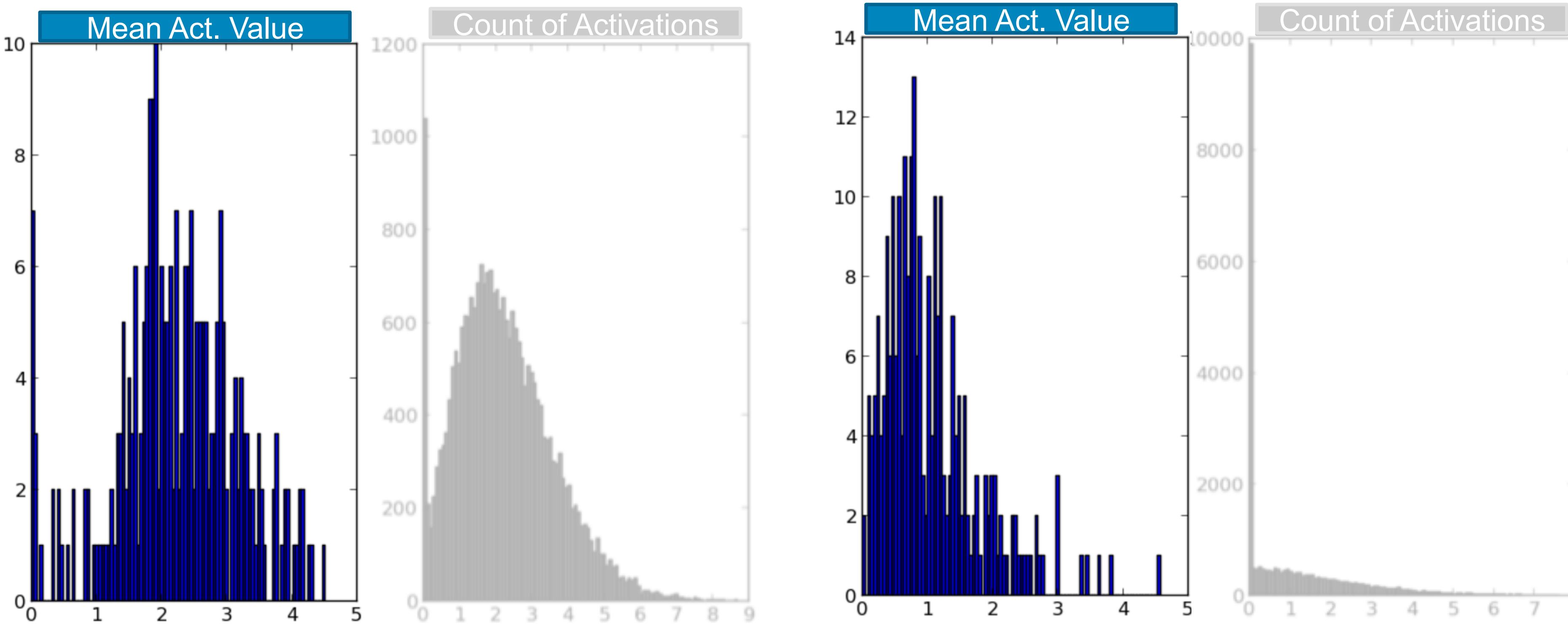
Dropout Training: More Structured Filters



(a) Without dropout

(b) Dropout with $p = 0.5$.

Dropout Training: More Compact Models



Srivastava et al. – Dropout: A simple way to prevent neural networks from overfitting – JMLR 2014

Toward a Theoretical Analysis of Dropout

- What kind of regularization does dropout induce?

- Can the regularized be characterized explicitly/analytically?

- **Theorem**: dropout with variable rate induces a low-rank regularizer (nuclear norm squared) for matrix factorization.

Deterministic vs Stochastic Factorization

- Deterministic Matrix Factorization (DMF)

$$\min_{U,V} \|Y - UV^\top\|_F^2$$

- Stochastic Matrix Factorization (SMF)

$$\min_{U,V} \mathbb{E}_{\boldsymbol{z}} \|Y - \frac{1}{\theta} \underbrace{U\mathrm{diag}(\boldsymbol{z})V^\top}_{\sum_{i=1}^{r} z_i U_i V_i^\top} \|_F^2, \quad z_i \sim \mathrm{Ber}(\theta), \quad \theta \in (0,1)$$

Dropout is SGD for SMF

- Stochastic matrix factorization objective

$$\min_{U,V} \mathbb{E}_{\boldsymbol{z}} \|Y - \frac{1}{\theta} U \operatorname{diag}(\boldsymbol{z}) V^\top\|_F^2$$

- Dropout is a stochastic gradient descent method for SMF

$$\begin{bmatrix} U^{t+1} \\ V^{t+1} \end{bmatrix} = \begin{bmatrix} U^{t} \\ V^{t} \end{bmatrix} + \frac{\epsilon}{\theta} \begin{bmatrix} (Y - \frac{1}{\theta} U^t \operatorname{diag}(\boldsymbol{z}^t) V^{t\top}) V^t \\ (Y - \frac{1}{\theta} U^t \operatorname{diag}(\boldsymbol{z}^t) V^{t\top}) U^t \end{bmatrix} \operatorname{diag}(\boldsymbol{z}^t)$$

- Compare to backpropagation with dropout

$$W^{t+1} = W^t - \frac{\epsilon}{|\mathcal{B}|} \sum_{j \in \mathcal{B}} \nabla \ell\big(Y_j, \Phi(X_j, W^t, \boldsymbol{z})\big) \otimes \boldsymbol{z}$$

Dropout as an Explicit Regularizer for SMF

- Using the definition of variance $\mathbb{E}(y^2) = \mathbb{E}(y)^2 + \mathrm{Var}(y)$ we can show that dropout induces an explicit regularizer

$$\mathbb{E}_{\boldsymbol{z}} \left\| Y - \tfrac{1}{\theta} U \mathrm{diag}(\boldsymbol{z}) V^\top \right\|_F^2 = \|Y - UV^\top\|_F^2 + \frac{1-\theta}{\theta} \sum_{i=1}^{r} \|U_i\|_2^2 \|V_i\|_2^2$$

- It really looks like the nuclear norm!!

$$\|X\|_* = \min_{U,V,r} \sum_{i=1}^{r} \|U_i\|_2 \|V_i\|_2 \quad \text{s.t.} \quad UV^\top = X$$

Dropout with Fixed Rate Fails to Regularize

- The dropout regularizer

$$\Theta(U, V) = \sum_{i=1}^{r} \|U_i\|_2^2 \|V_i\|_2^2$$

fails to regularize the size of the factorization because we can lower the objective by doubling the size of the factorization

$$\Theta\left(\frac{1}{\sqrt{2}}\begin{bmatrix}U & U\end{bmatrix}, \frac{1}{\sqrt{2}}\begin{bmatrix}V & V\end{bmatrix}\right) = \frac{1}{2}\Theta(U, V)$$

Dropout with Variable Rate Works

- Recall the dropout regularizer with regularization parameter

$$\lambda\Theta(U,V) = \frac{1-\theta}{\theta}\sum_{i=1}^{r}\|U_i\|_2^2\|V_i\|_2^2$$

- What if dropout rate varies?

$$\lambda_r = \frac{1-\theta_r}{\theta_r} = r\frac{1-\theta_1}{\theta_1} = r\lambda_1$$

- Then, pathological case disappears

$$\lambda_{2r}\Theta\left(\frac{1}{\sqrt{2}}\begin{bmatrix}U & U\end{bmatrix}, \frac{1}{\sqrt{2}}\begin{bmatrix}V & V\end{bmatrix}\right) = \lambda_r\Theta(U,V)$$

Dropout with Variable Rate Works

- **Proposition**: Dropout with variable rate induces a regularizer

$$\Omega(X) = \min_{U,V,r} \frac{1-\theta_r}{\theta_r} \sum_{i=1}^{r} \|U_i\|_2^2 \|V_i\|_2^2 \quad \text{s.t.} \quad UV^\top = X$$

whose convex envelope is the (nuclear norm)² $\frac{1-\theta_1}{\theta_1}\|X\|_*^2$

- **Theorem**: Let (U*,V*,r*) be a global minimum of

$$\min_{U,V,r} \|Y - UV^\top\|_F^2 + \frac{1-\theta_r}{\theta_r} \sum_{i=1}^{r} \|U_i\|_2^2 \|V_i\|_2^2$$

Then, $X^* = U^* {V^*}^\top$ is a global minimum of $\min_X \|Y - X\|_F^2 + \frac{1-\theta_1}{\theta_1}\|X\|_*^2$

JOHNS HOPKINS
MATHEMATICAL INSTITUTE
for DATA SCIENCE

Global Optima are Low Rank

$$\min_{U,V,r} \|Y - UV^\top\|_F^2 + \frac{1-\theta_r}{\theta_r} \sum_{i=1}^{r} \|U_i\|_2^2 \|V_i\|_2^2$$

- **Theorem**: (U*,V*,r*) is a global minimum iff

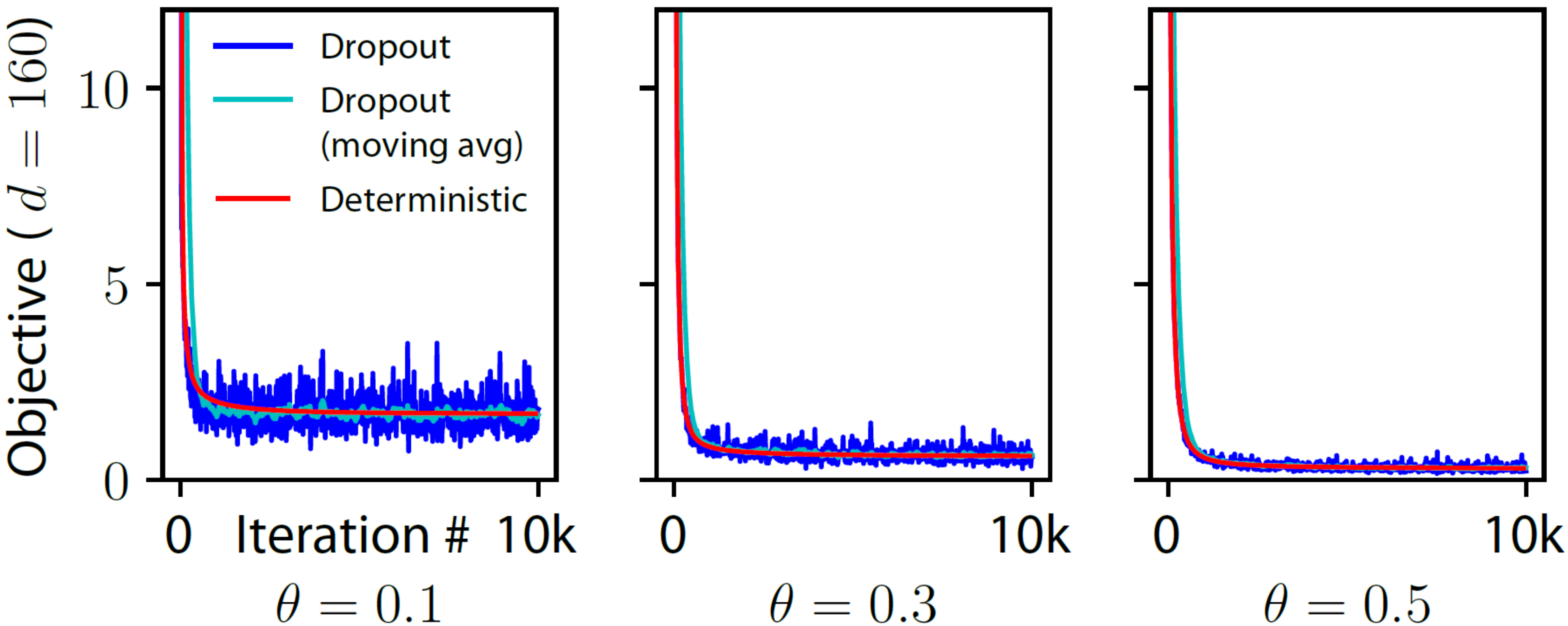
  $$U^* {V^*}^\top = \mathcal{S}_\tau(Y)$$

  where tau and r* depends on singular values of Y

- **Open issues:**
  - Results are valid for variable r, but not for a fixed r
  - How to find the optimal (U*, V*) ?

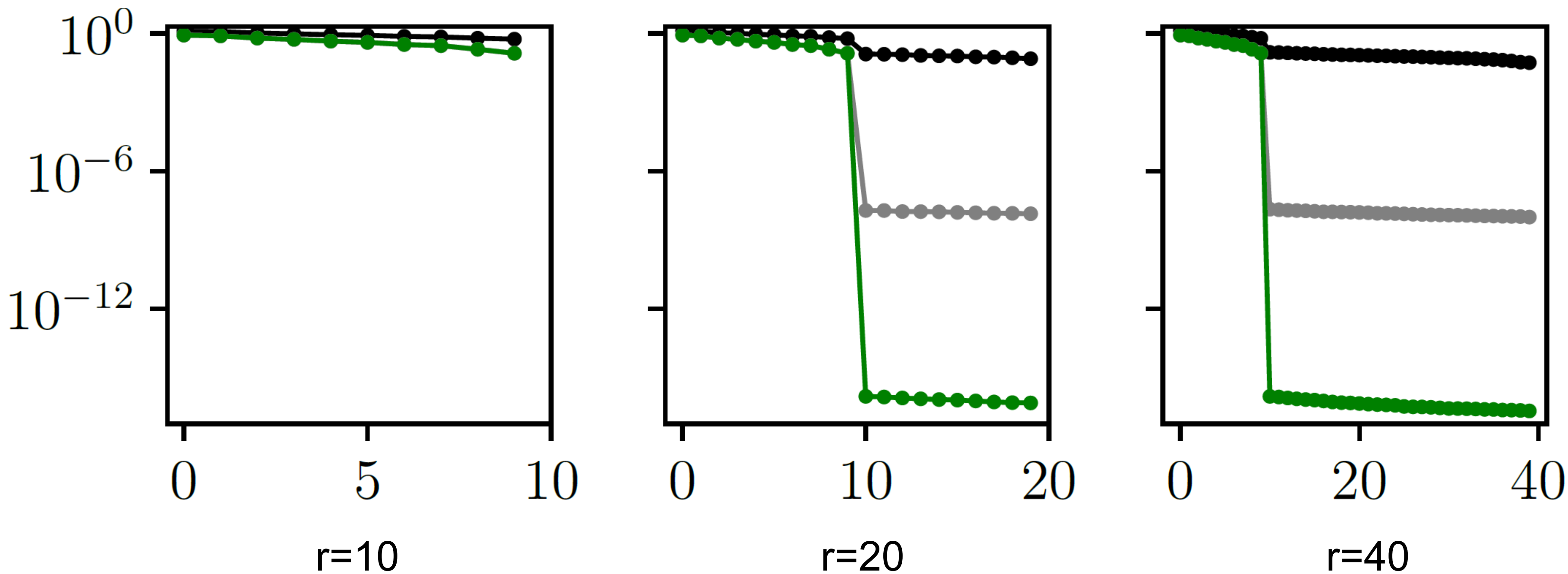
Synthetic Experiments for Fixed Size

- Comparing deterministic and stochastic dropout for factorizing a 100 x 100 matrix with fixed size r = 160.
- Run 10,000 iterations of GD with diminishing step size.



Synthetic Experiments for Variable Size

- Comparing dropout with fixed rate (black), adaptive rate (gray) and closed form solution (green) for factorizing 100 x 100 matrix of rank 10 + noise.



Conclusions

- Dropout for matrix factorization is an SGD method
- Dropout for matrix factorization induces explicit regularization
- Dropout for matrix factorization with a fixed dropout rate does not limit the size of the factorization
- Dropout for matrix factorization with a dropout rate that increases with the size of the factorization induces low-rank factorizations

JHU vision lab

On the Implicit Bias of Dropout

Poorya Mianjy[1,2] Raman Arora[1,2] and René Vidal[1,3]

[1]Mathematical Institute for Data Science, Johns Hopkins University, USA

[2]Department of Computer Science, Johns Hopkins University, USA

[3]Department of Biomedical Engineering, Johns Hopkins University, USA

JOHNS HOPKINS MATHEMATICAL INSTITUTE for DATA SCIENCE

THE DEPARTMENT OF BIOMEDICAL ENGINEERING
The Whitaker Institute at Johns Hopkins

Center for IMAGING SCIENCE

What Solutions does Dropout Favor?

- Recall dropout is an instance of SGD on the objective

$$\mathbb{E}_{\boldsymbol{z}} \left\| Y - \tfrac{1}{\theta} U \operatorname{diag}(\boldsymbol{z}) V^\top \right\|_F^2 = \|Y - UV^\top\|_F^2 + \frac{1-\theta}{\theta} \sum_{i=1}^{r} \|U_i\|_2^2 \|V_i\|_2^2$$

- Results so far guarantee global optimality when optimizing over (U,V,r) provided that r “large enough”

- Results so far tell us what the optimal product is, but do not tell us what the optimal factors look like

- **Question 1:** Can we find the global minimum for any fixed r ?

- **Question 2:** What optimal solutions does dropout favor?

Any Factorization Can Be Equalized

- The network with weights (U,V) is said to be **equalized** if the product of the norms of incoming and outgoing weights are equal for all hidden nodes, i.e.

$$\|U_i\|_2\|V_i\|_2 = \|U_j\|_2\|V_j\|_2 \quad \forall i,j = 1,\dots,r$$

- **Theorem**: For any pair *(U,V)* there is another pair *(U',V')* such that $UV^T = U'V'^T$ and *(U',V')* can be equalized by a rotation *R*, i.e., there is a rotation *R* such that *(U'R, V'R)* are equalized.

- **Algorithm to compute *(U',V',R)***: based on Gram matrices, eigenvalue decompositions and matrix diagonalization

Global Minima are Equalized

- **Theorem:** global optima of dropout problem are equalized

$$\min_{U,V} \underbrace{\|Y - UV^\top\|_F^2}_{\ell(U,V)} + \lambda \underbrace{\sum_{i=1}^{r} \|U_i\|_2^2 \|V_i\|_2^2}_{\Theta(U,V)}$$

- Loss is rotationally invariant: $\ell(U,V) = \ell(UR, VR)\ \forall R$

- Regularizer minimized when network is equalized by rotation

$$\boldsymbol{n}_{u,v} = (\|U_1\|\|V_1\|, \|U_2\|\|V_2\|, \ldots, \|U_r\|\|V_r\|)$$

$$\Theta(U,V) = \tfrac{1}{r}\|\mathbf{1}_r\|^2\|\boldsymbol{n}_{u,v}\|^2 \geq \tfrac{1}{r}\Big(\sum_{i=1}^{r}\|U_i\|_2\|V_i\|_2\Big)^2$$

Global Optima are Low Rank

$$\min_{U,V} \|Y - UV^\top\|_F^2 + \lambda \sum_{i=1}^{r} \|U_i\|_2 \|V_i\|_2$$

- **Theorem**: (U*,V*) is a global minimum iff it is equalized and

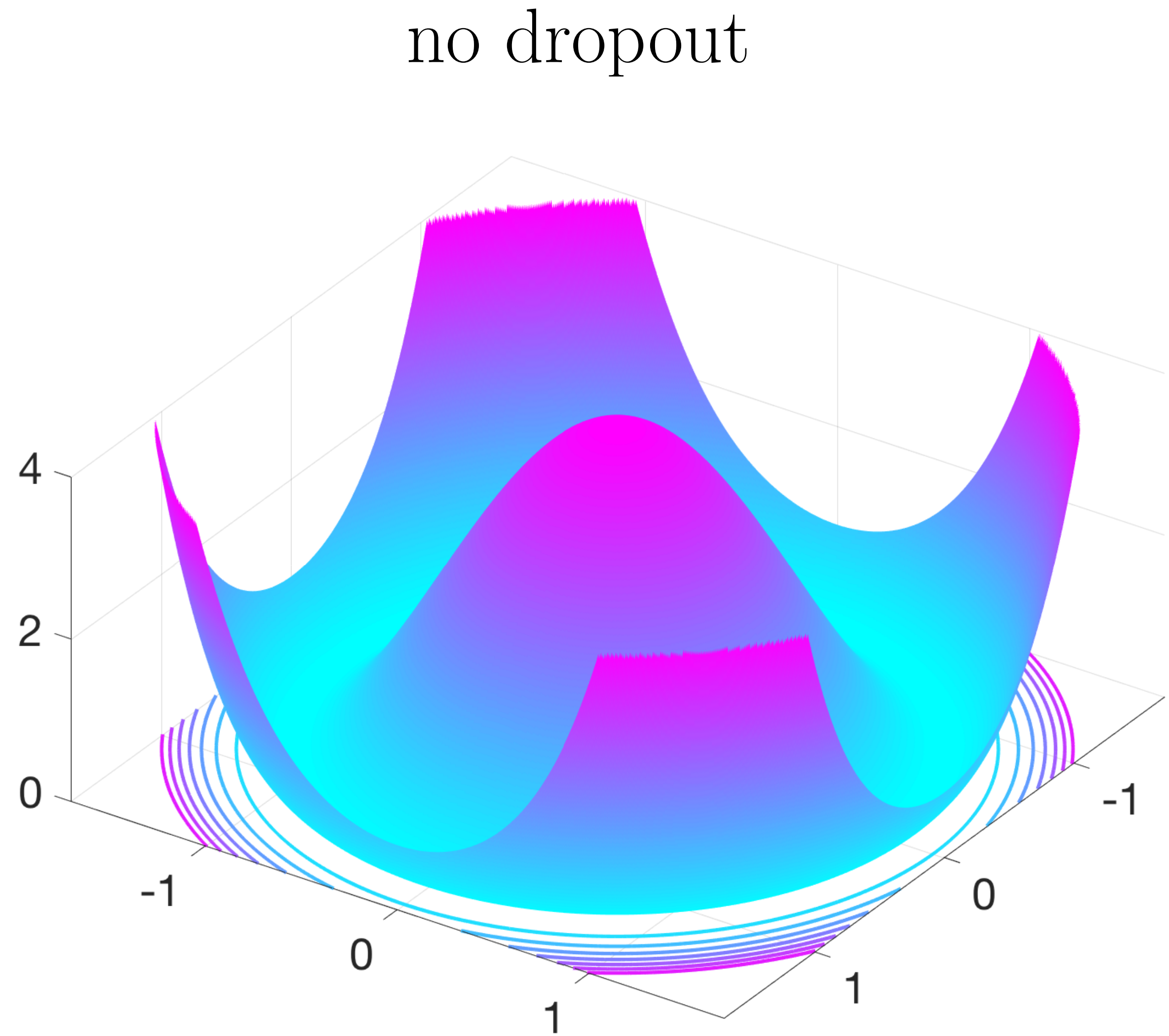
$$U^* {V^*}^\top = \mathcal{S}_\tau(Y)$$

where tau and optimal r depends on singular values of Y

- **Algorithm:** A global optimum (U*,V*) can be found as follows
  - Find any factorization (U,V) of $\mathcal{S}_\tau(Y)$
  - Equalize the factors to obtain (U*, V*) = (UR, VR)

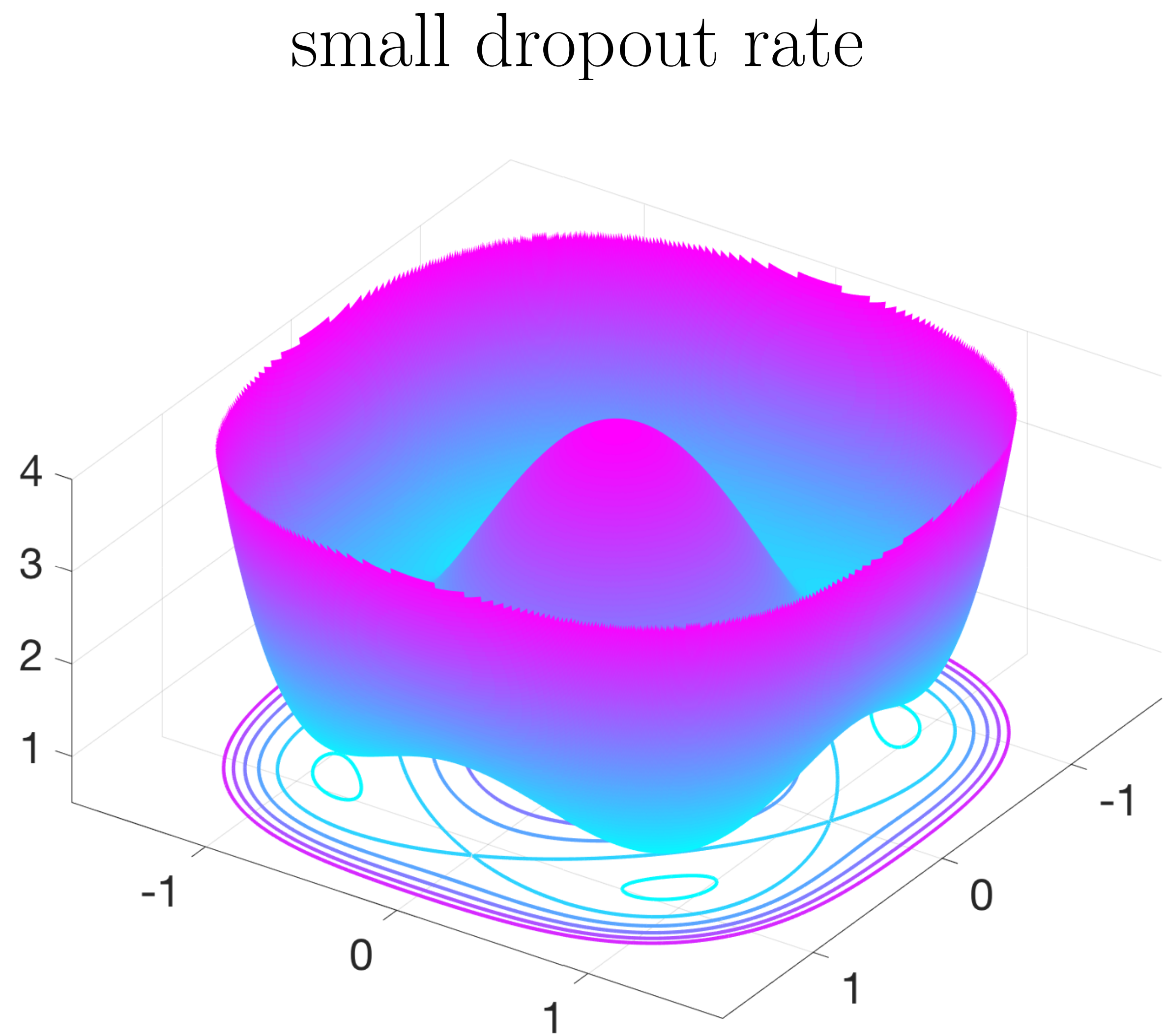
Effect of Dropout Rate on the Landscape

- Linear auto-encoder
- 1 input
- 2 hidden neurons
- 1 output



Effect of Dropout Rate on the Landscape

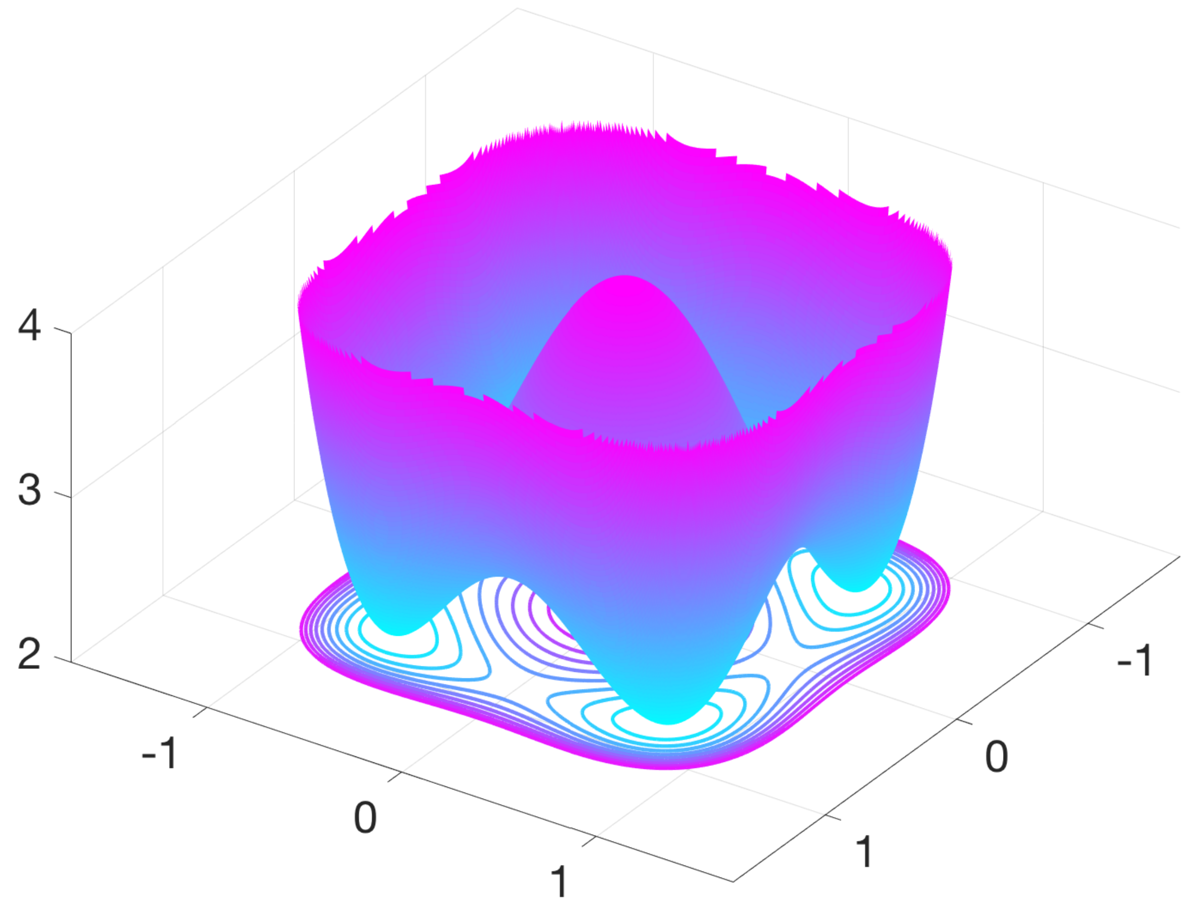
- Linear auto-encoder
- 1 input
- 2 hidden neurons
- 1 output



Effect of Dropout Rate on the Landscape

- Linear auto-encoder
- 1 input
- 2 hidden neurons
- 1 output

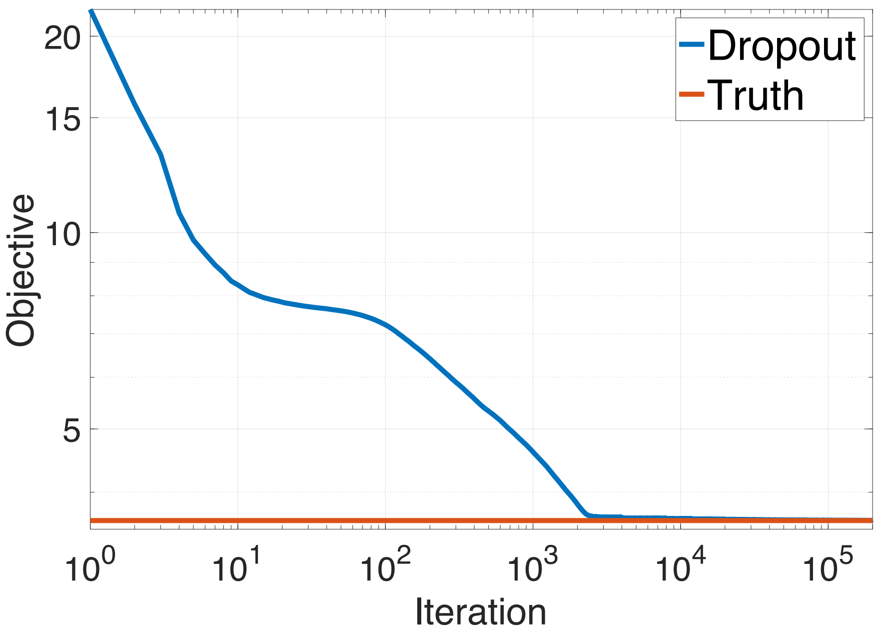
large dropout rate



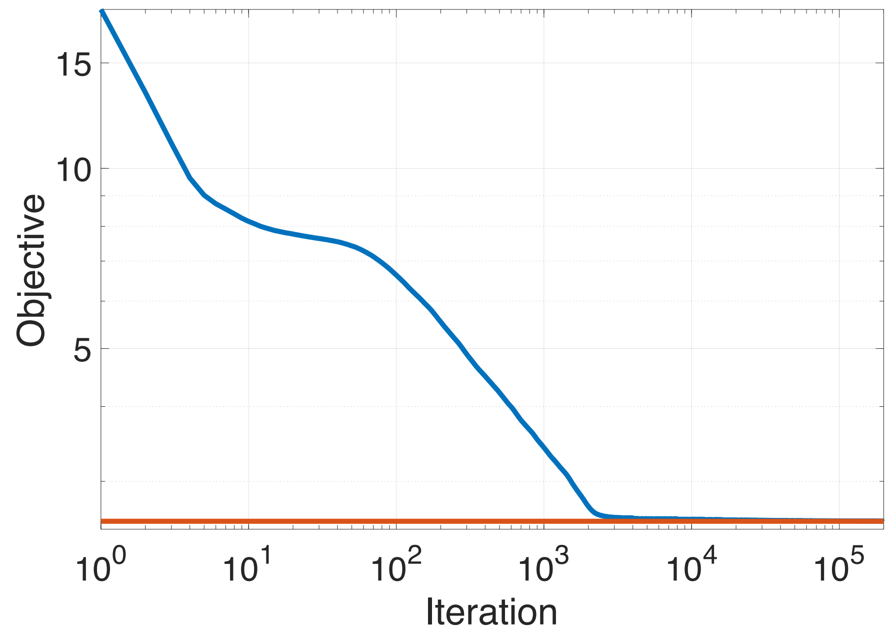
Synthetic Experiments

- Comparing stochastic dropout and closed form solution for factorizing a 120 x 80 matrix with fixed size r = 20.

$\lambda = 1$

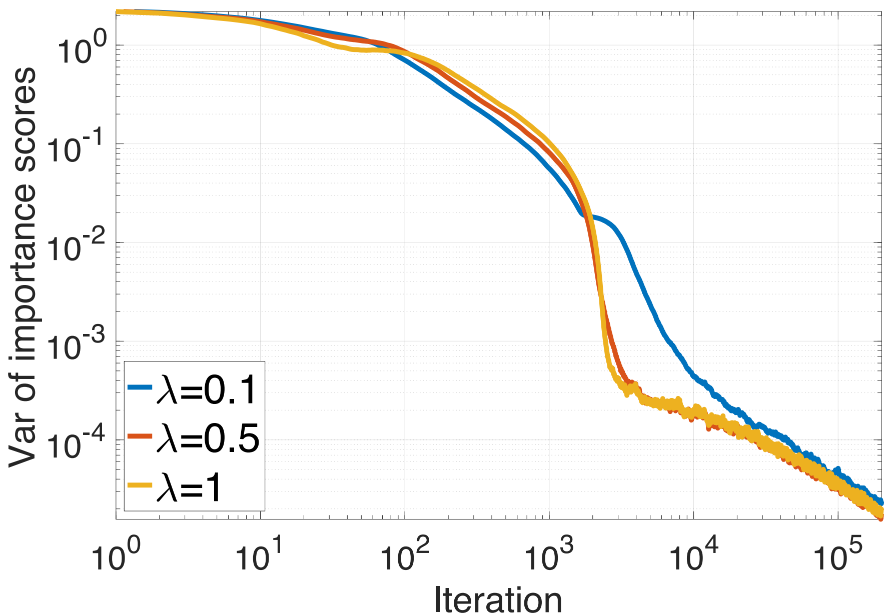


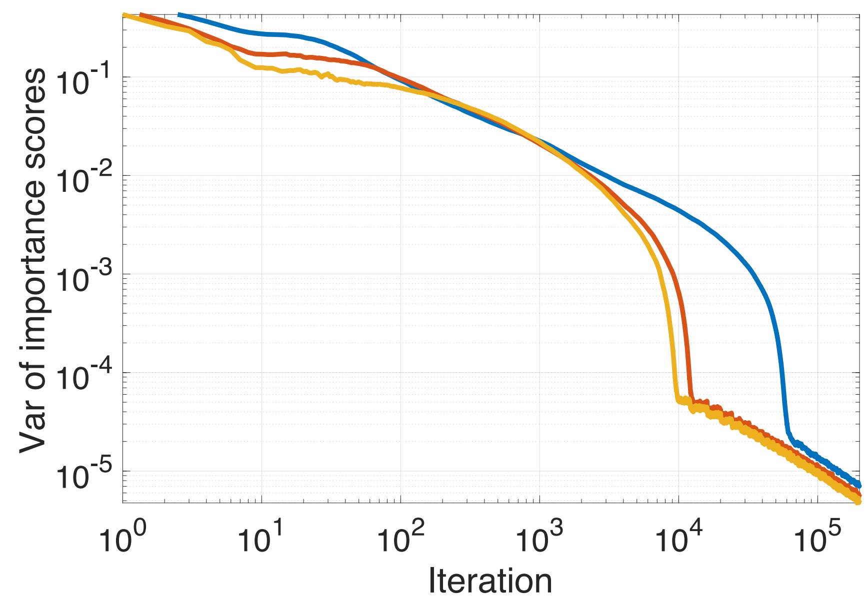
$\lambda = 0.5$



Synthetic Experiments

- Showing that stochastic dropout achieves equalization when factorizing a 120 x 80 matrix with fixed size r = 20 and r = 80.

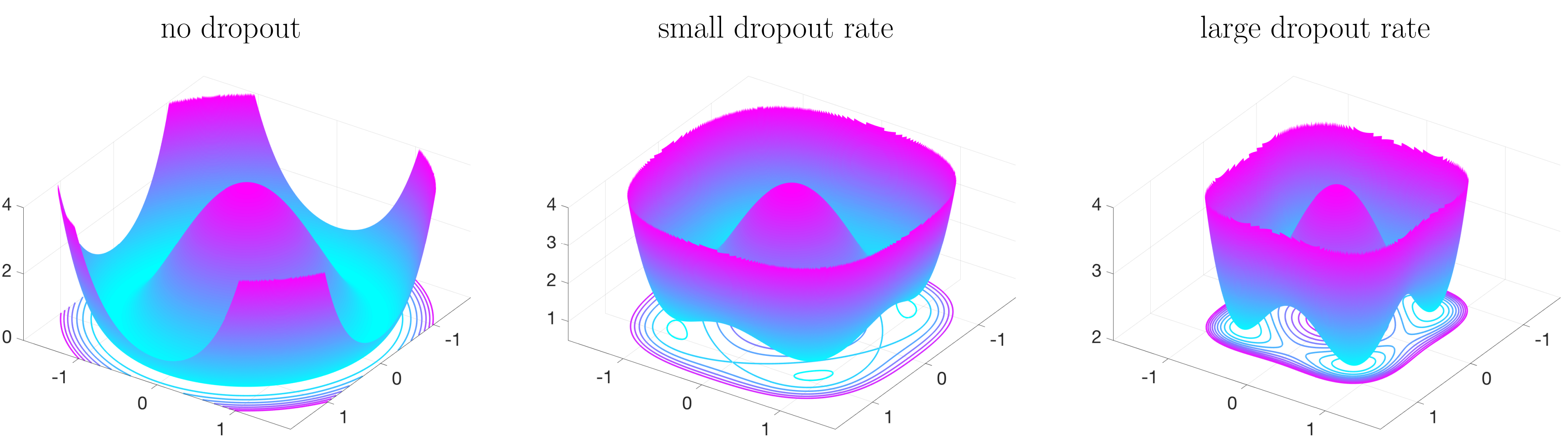




Conclusions

- Dropout with fixed size also induces a low-rank regularizer

- The global optima for any fixed *r* are equalized and low-rank



More Information,

Vision Lab @ JHU

http://www.vision.jhu.edu

Center for Imaging Science @ JHU

http://www.cis.jhu.edu

Mathematical Institute for Data Science @ JHU

http://www.minds.jhu.edu

Thank You!