From Deep BP-Learning to Internal X-Learning

S.Y. Kung

Princeton

Outline

A. From AI1.0 to AI2.0 to XAI3.0

B. X-Learning: Enhance Accuracy via Model Reduction

- ITL: node ranking and structural learning
- DI: Internal Optimization Metrics (IOM)

C. Iterative X-Learning, DNs, & Results

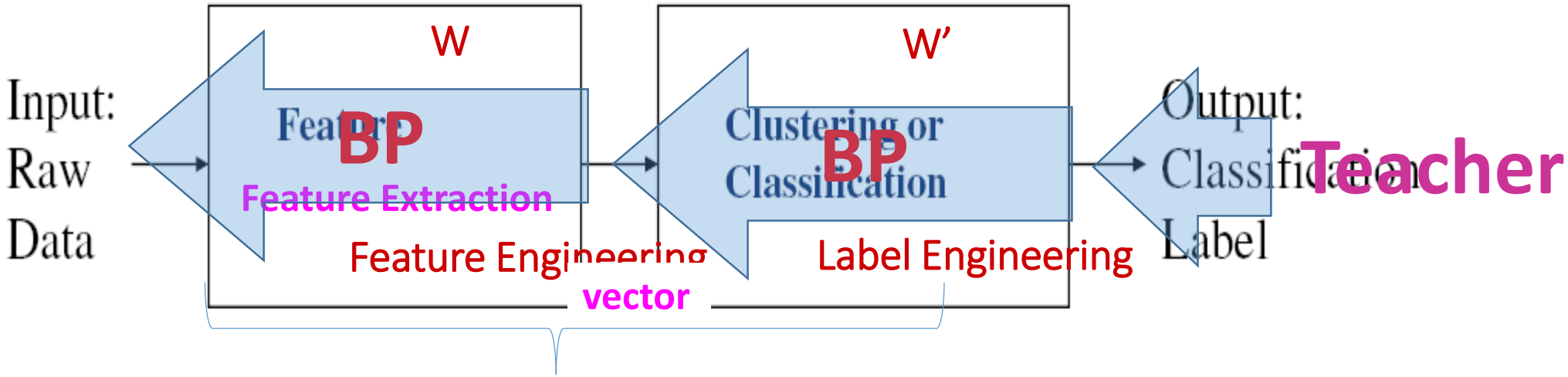
- NP iteration: Joint PS Learning
- Deleterious Nodes (DNs)
- Xnet: MNIST, CIFAR, & ImageNet

D. Extension to Broader Applications

- LAX-learning, EX-learning
- Regression Analysis: RX-Learning
- IMaX-learning (5G-AI)

1950	⇒ 2000	⇒ 2020
NN1.0(MLP)	⇒ NN 2.0(DNN)	⇒ Xnet
BP	Deep BP-Learning	Internal X-Learning
Training by Examples	深度学习	深入学习
M. Minsky, first neural network simulator, Princeton Ph. D. 1951.		
MIT AI Lab (1958, M. Minsky)		
AI 1.0	⇒ AI 2.0	⇒ XAI 3.0
Knowledge Systems Laboratory, (1970 Feigenbaum)	⇒ Data-driven	
Rule-based		

AI2.0: DLN = Feature Engineering + Label Engineering

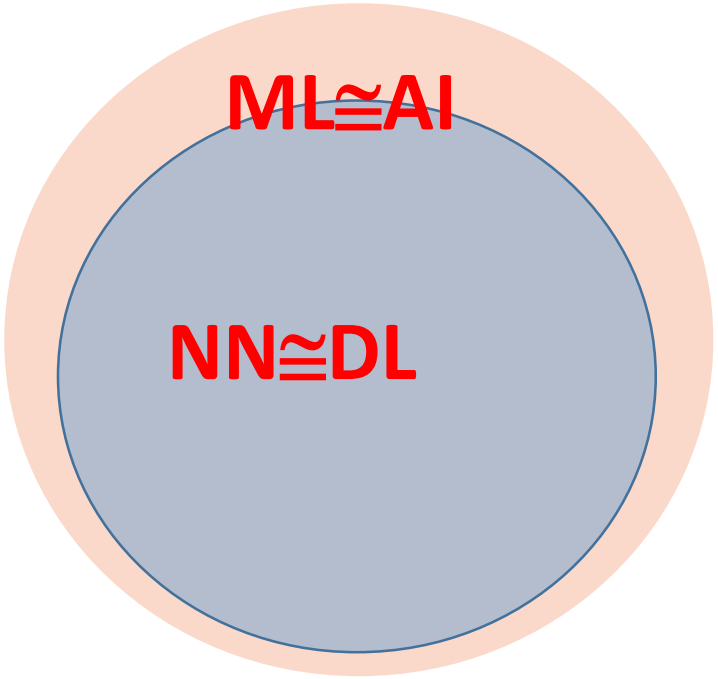


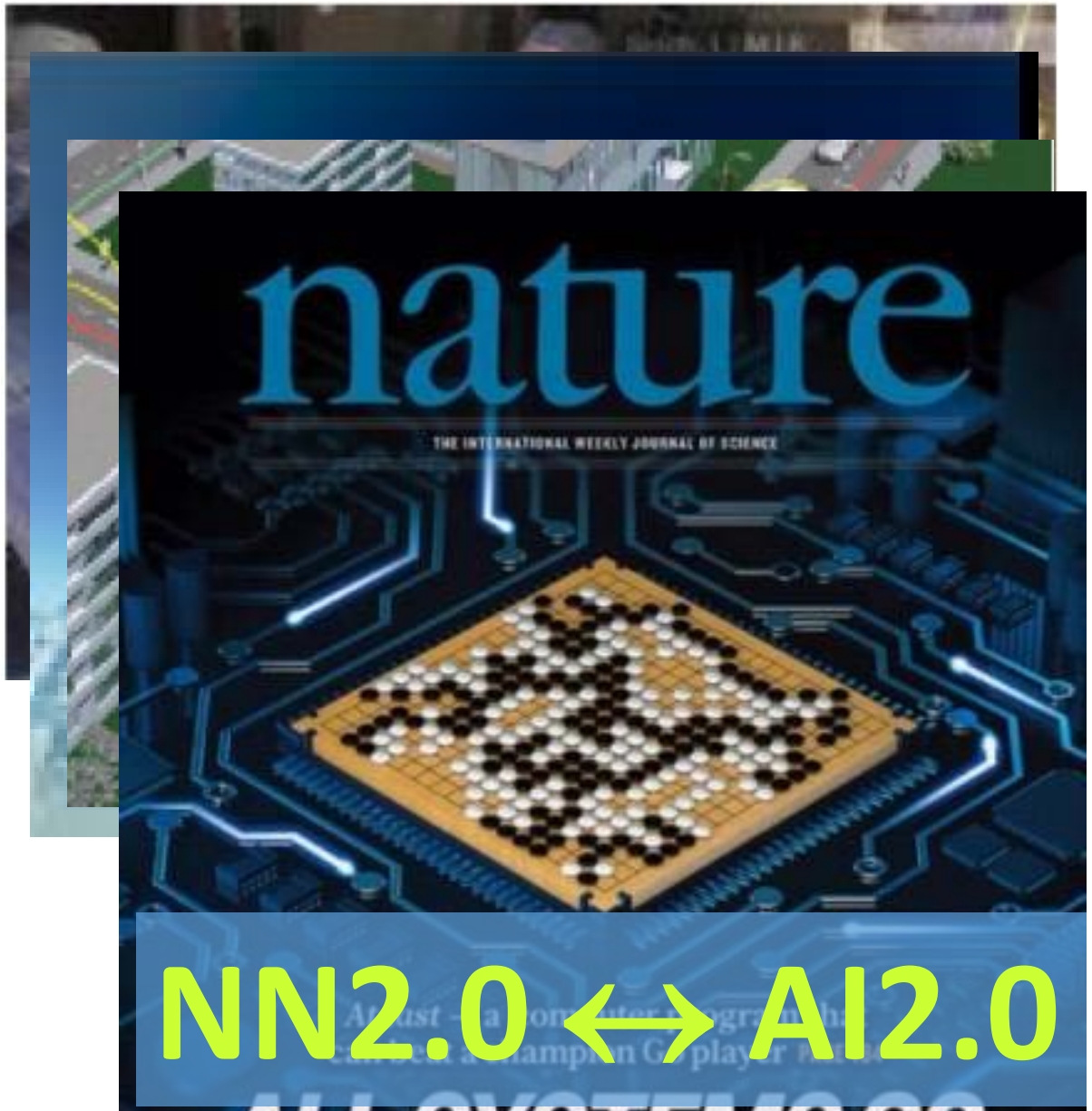
NN2.0 = MLP + ConvNet + Deep BP

Learning, Memorization vs. Generalization…. **doing BP on thousands (if not millions) of parameters**

Deep learning networks represent the **state-of-the-arts** of **ML**.

From Rule-based AI1.0 to Data-Driven AI2.0

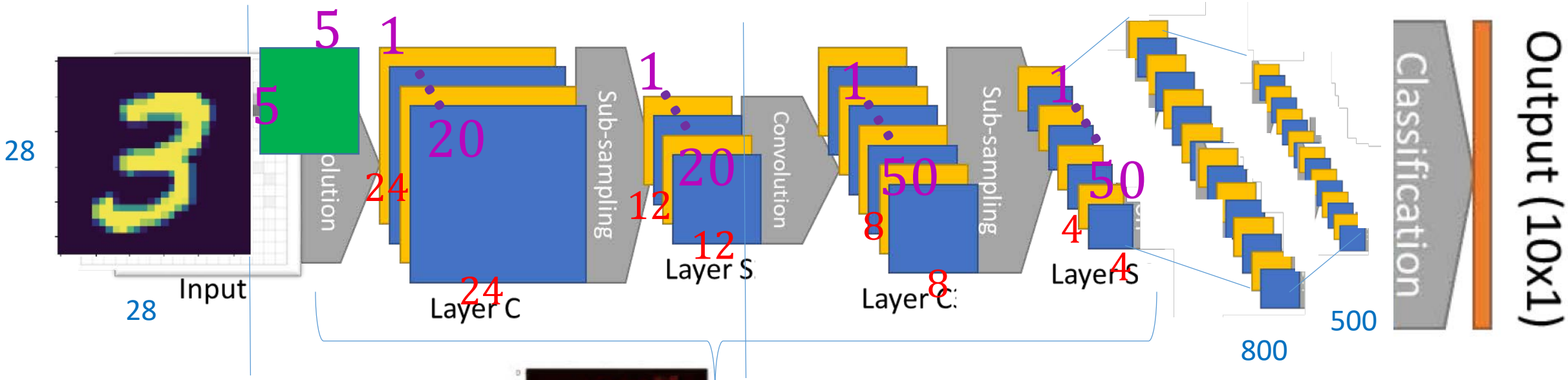




Characterization of Convolutional Neural Nets

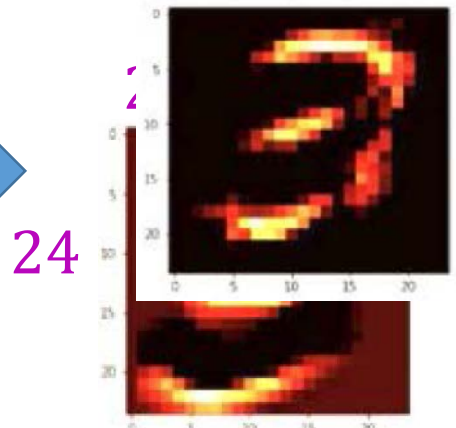
$$\{28^2 \rightarrow [f_1\ n_1\ m_1\ \mu_1] \rightarrow [f_2\ n_2\ m_2\ \mu_2] - 800 - 500 - 10\}$$

LeNet-5[Han15]: $\{28^2 \rightarrow [5\ 20\ 24\ 12] \rightarrow [5\ 50\ 8\ 4] - 800 - 500 - 10\ \}$



BP friendly

X-learning



$24=28-5+1$

$8=12-5+1$

$800=50\times4^2$

From Deep BP-Learning to Internal X-Learning

Back-propagation (**BP**) is only for **parameter learning.**

X-learning adopts **internal &** ***Explainable Learning (Xnet),*** going beyond BP to train both NN's ***structure and parameters***.

DARPA XAI $\Rightarrow$ **X-Learning**

DARPA's XAI ⇒ X-Learning

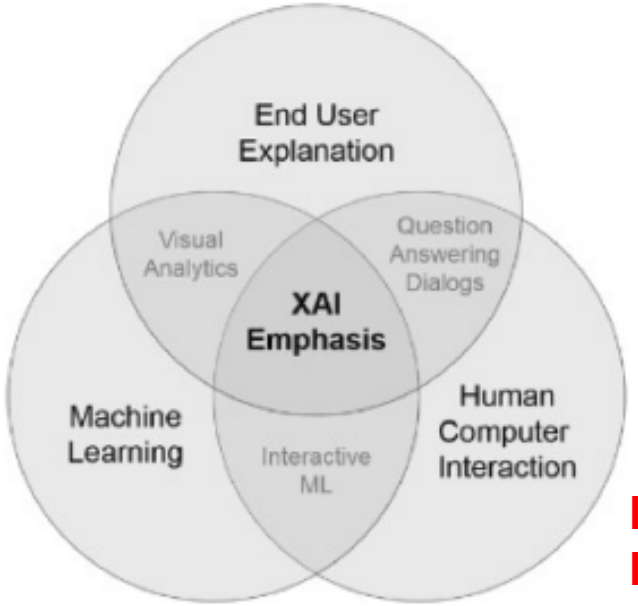
Broad Agency Announcement

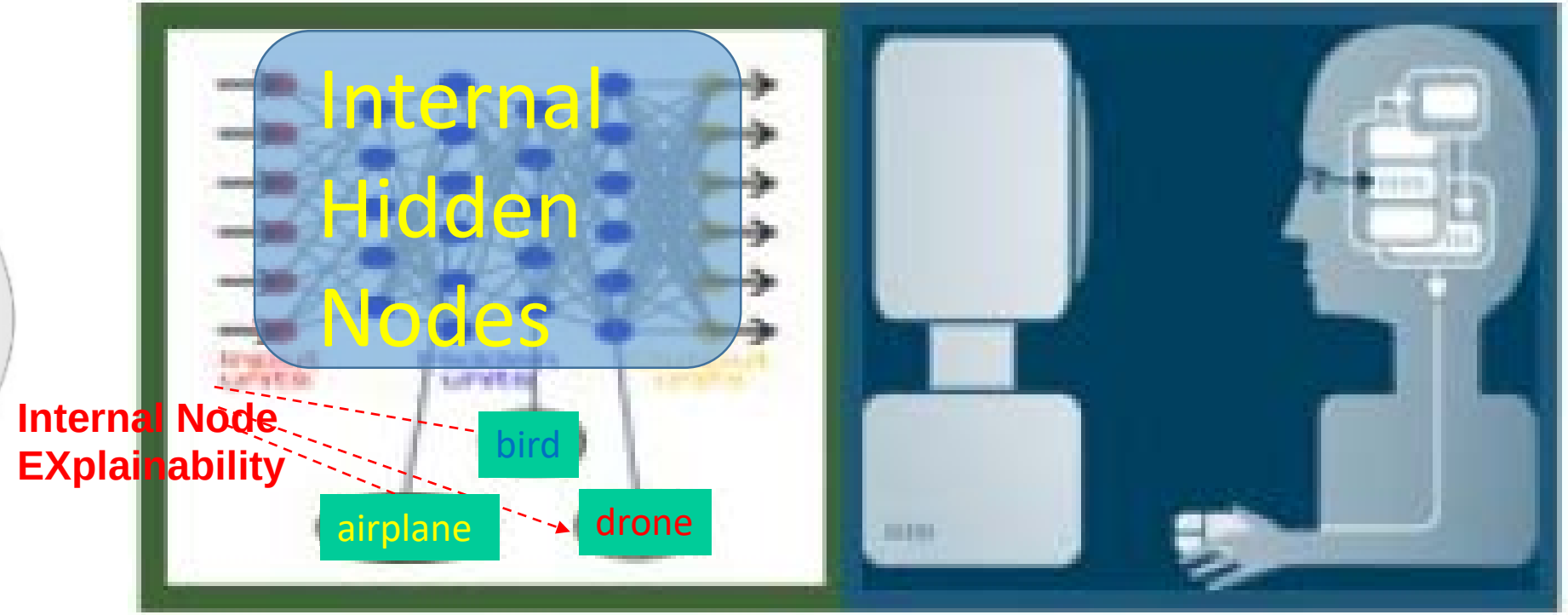
Explainable Artificial Intelligence (XAI)

DARPA-BAA-16-53

August 10, 2016

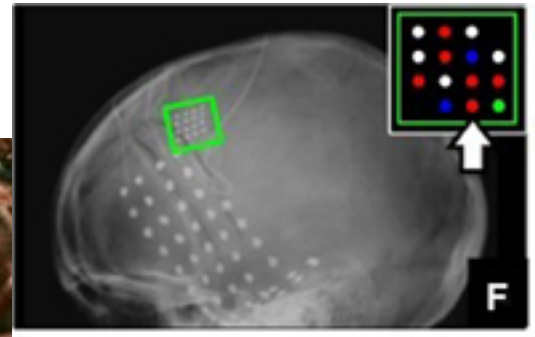
End User Explainability: ``XAI is most interested in explanations of higher-level decisions that would be relevant to the end user and the missions he/she needs to manage."[DARPA2016]

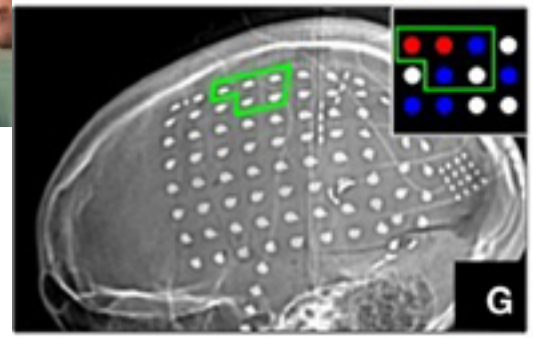


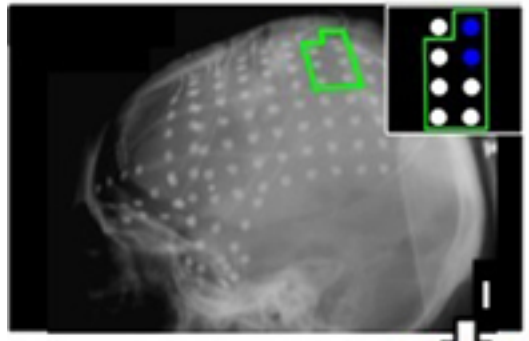


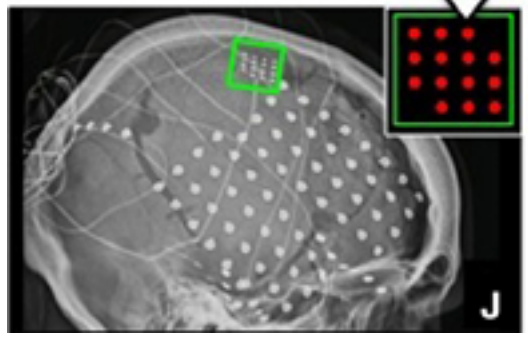
Motivated by XAI, X-learning will play vital roles in XAI.

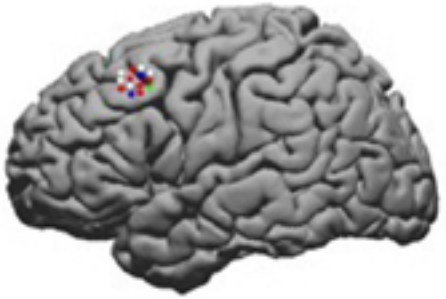
Biology for Internal X-Learning

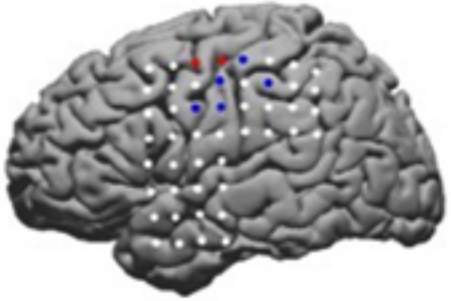


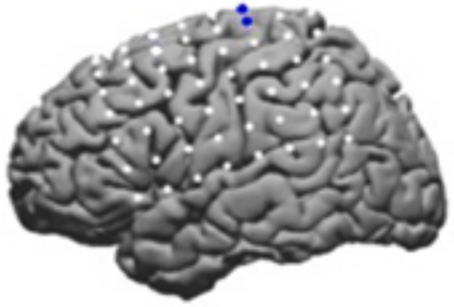


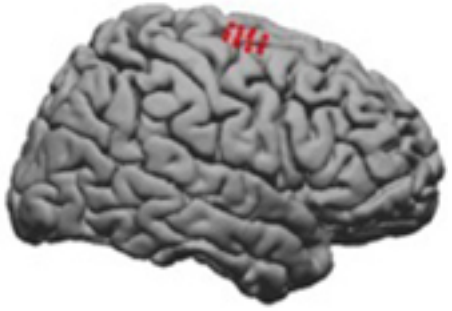












Why Internal X-Learning?

Bird

or Drone

B. X-Learning

X-Learning facilitates Both Parameter/Structure Learning!!

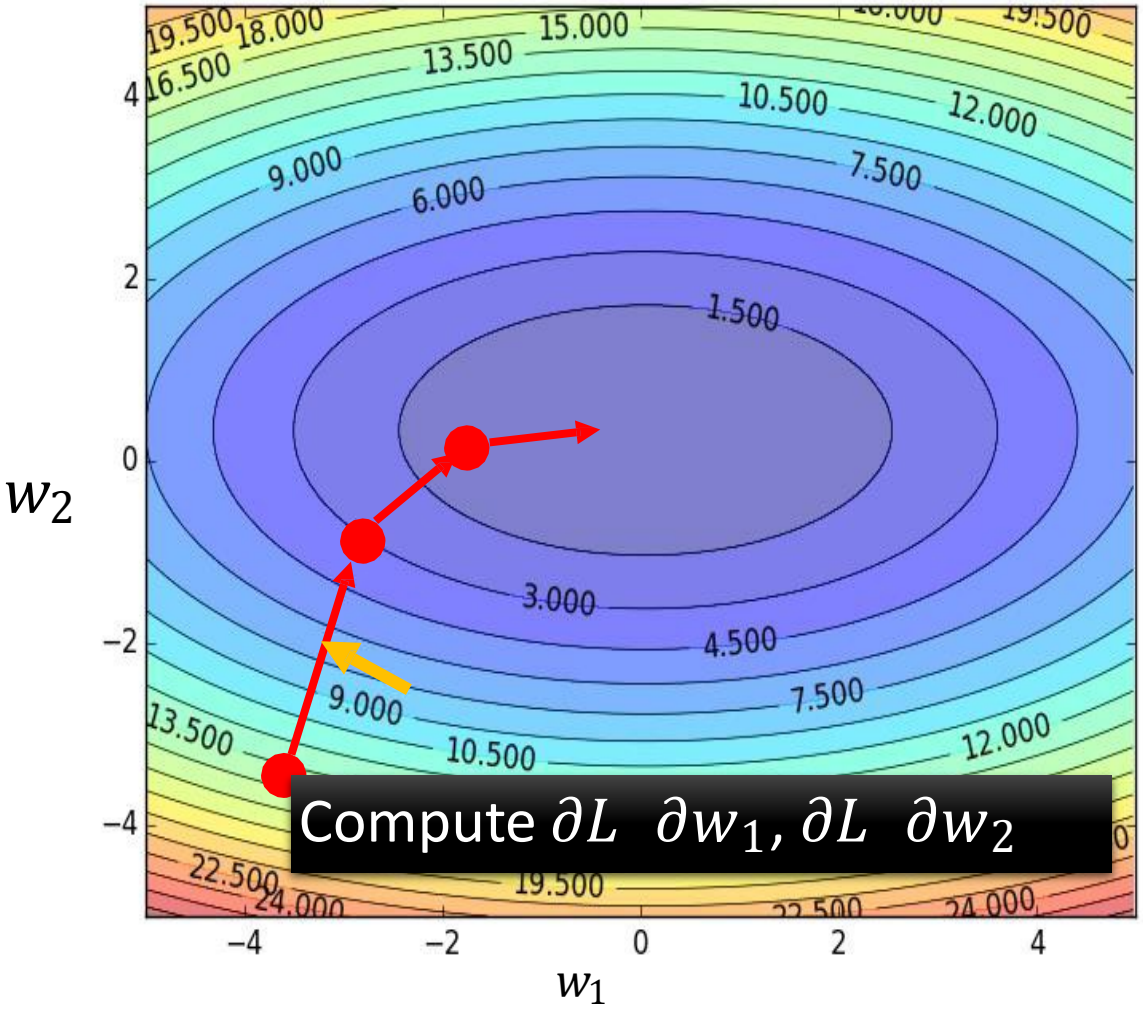
Parameter Learning:

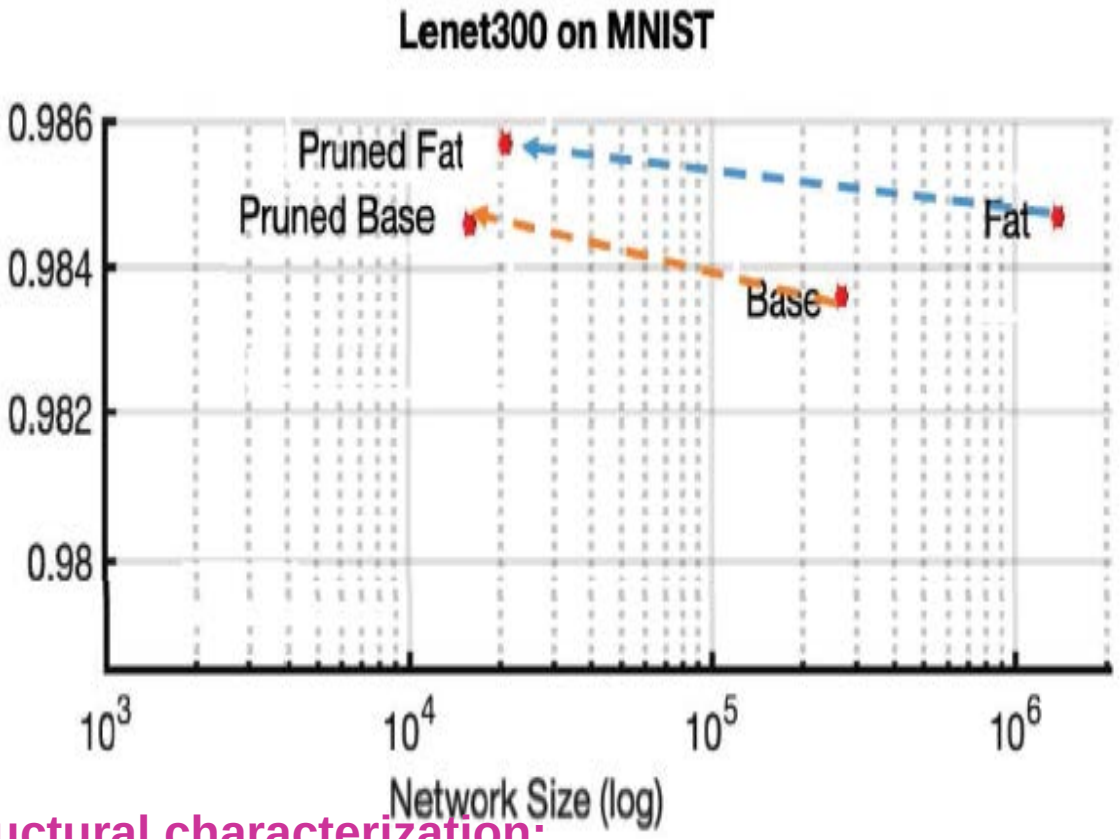
EOM Gradient Descent

+

Structural Learning :

IOM Guided Adaptation

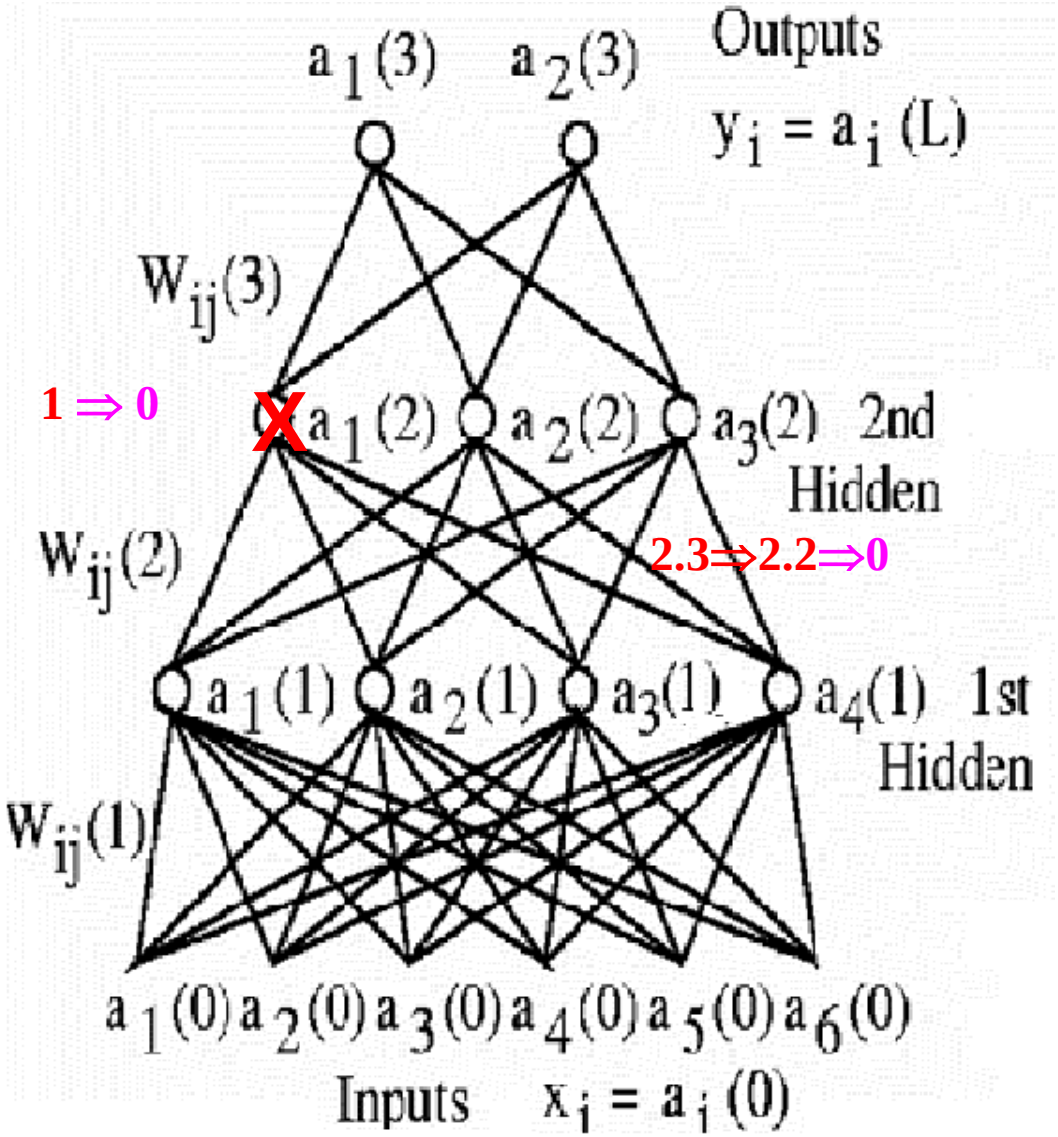




Structural characterization:

- **The number of layers in the model.**
- **The number of nodes in each layer.**

X-learning⇒ Structural Gradient



A quantum jump from finite to zero:

- **individually (connection)**
- **or collectively (node).**

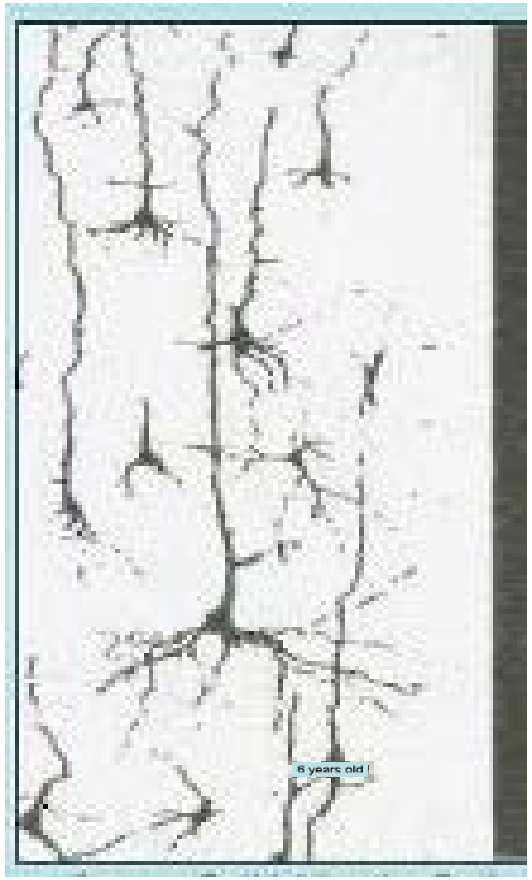
Optimization Algorithm

➡ **LASSO (Link/Node)**

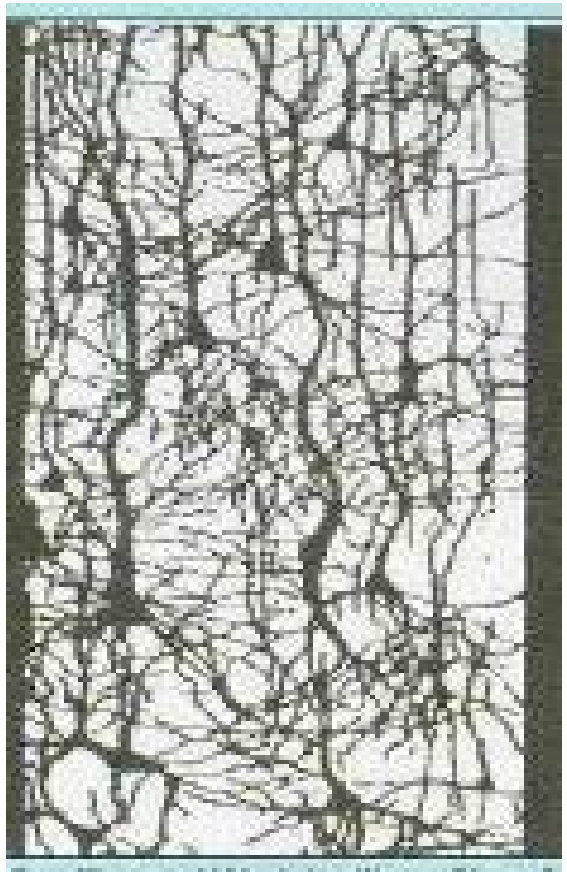
B(i): Why X-Learning?

Biological Motivation

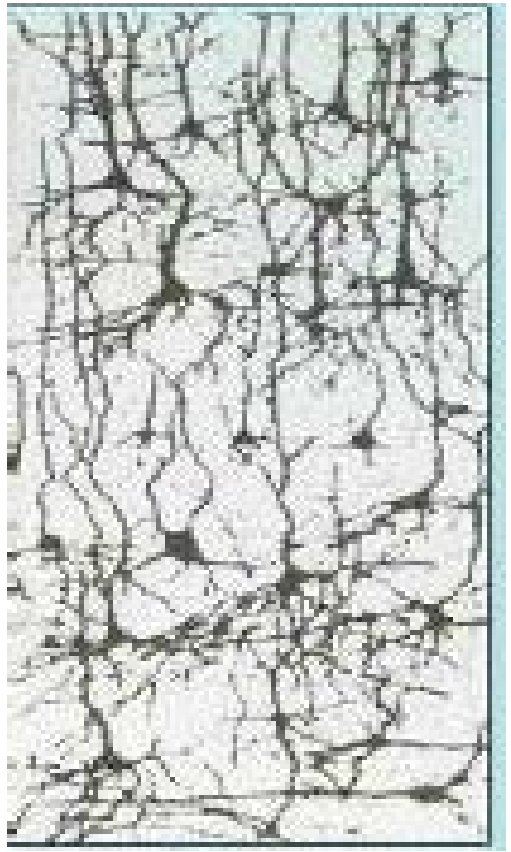
Synaptic Density



At birth

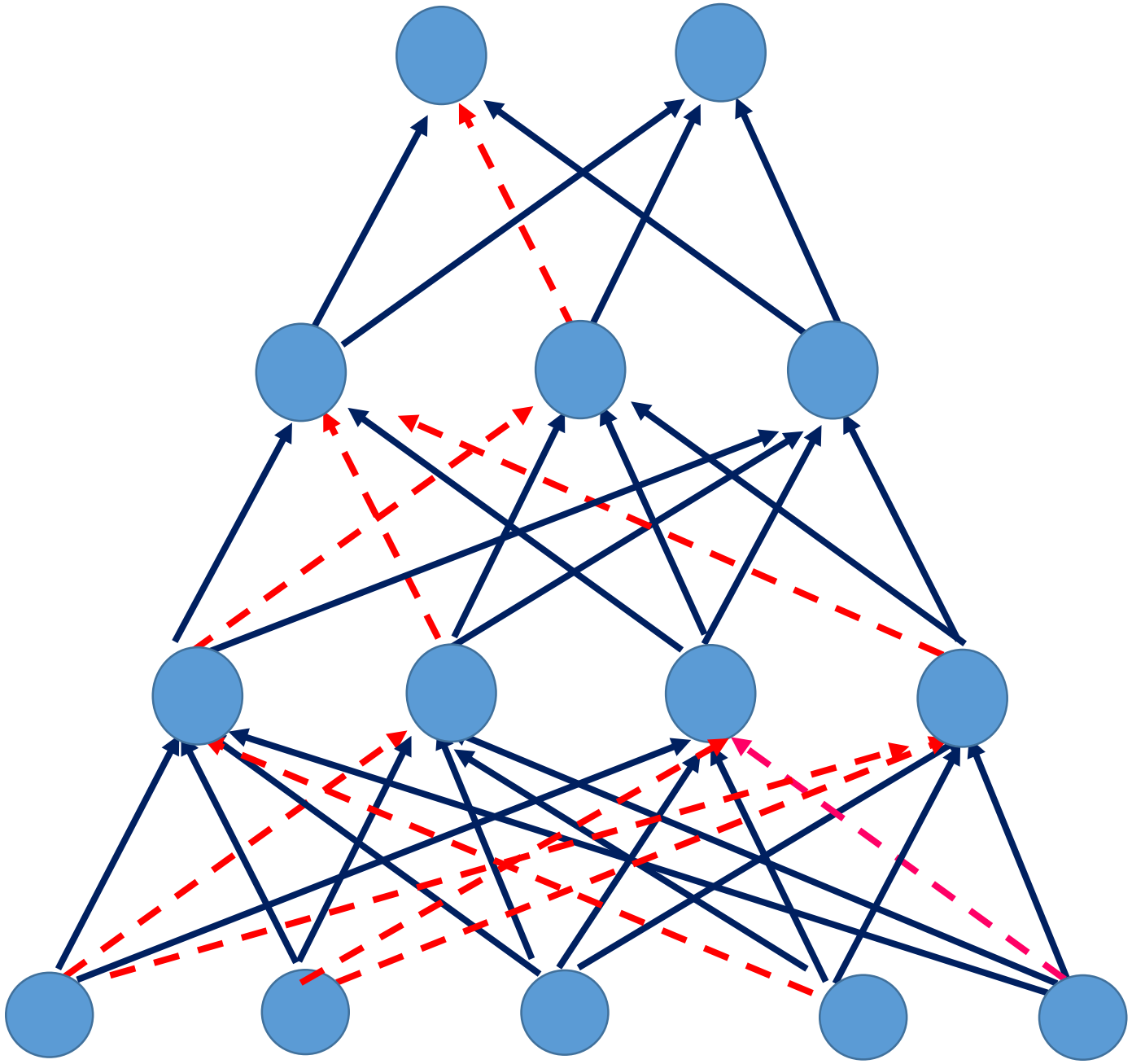


6 year old

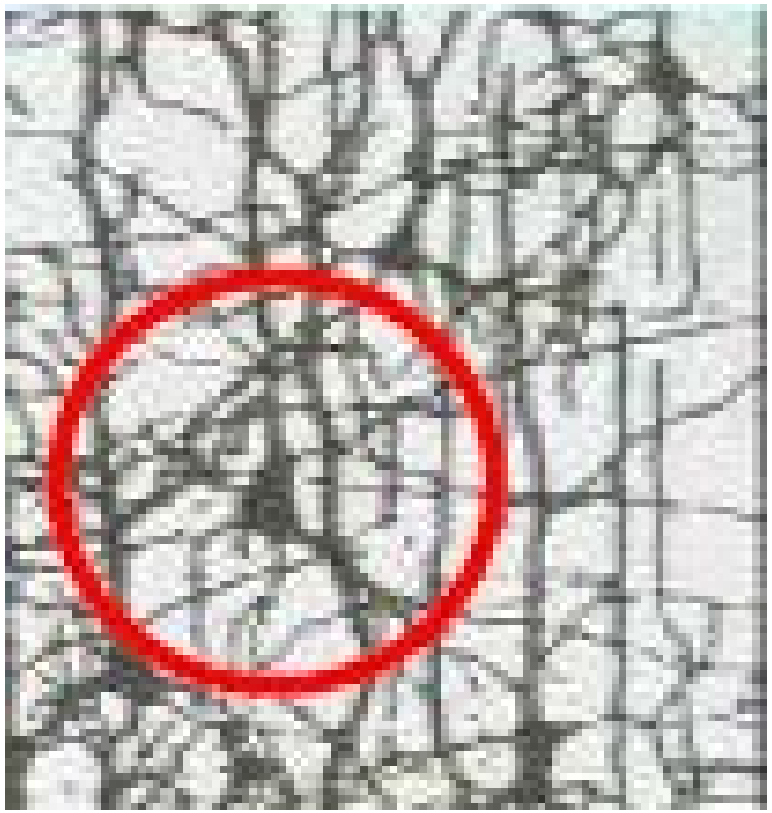


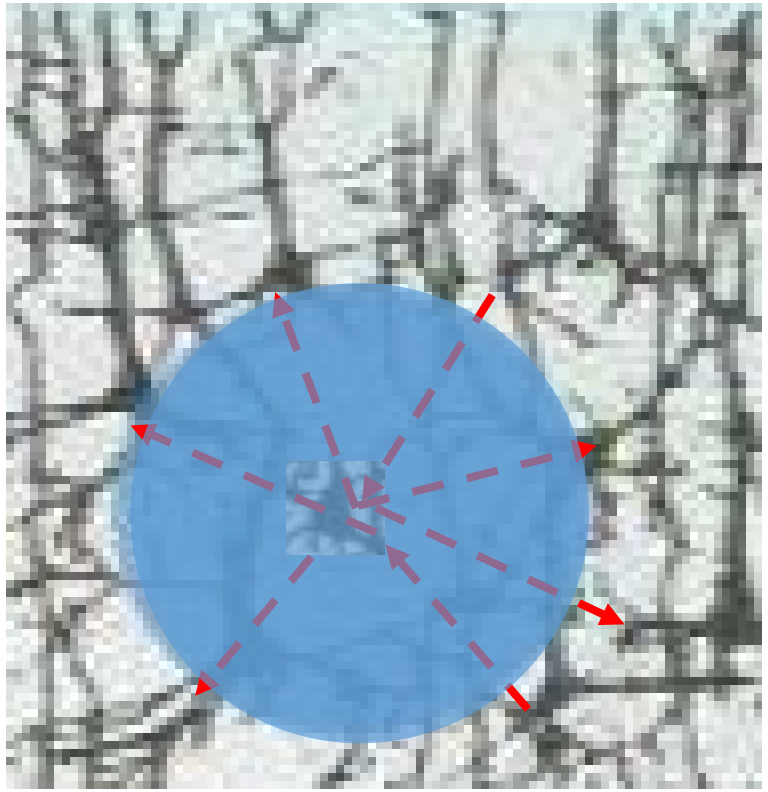
14 year old

Source: Rethinking the Brain, Families and Work Institute, Rima Shore, 1997; Founders Network slide

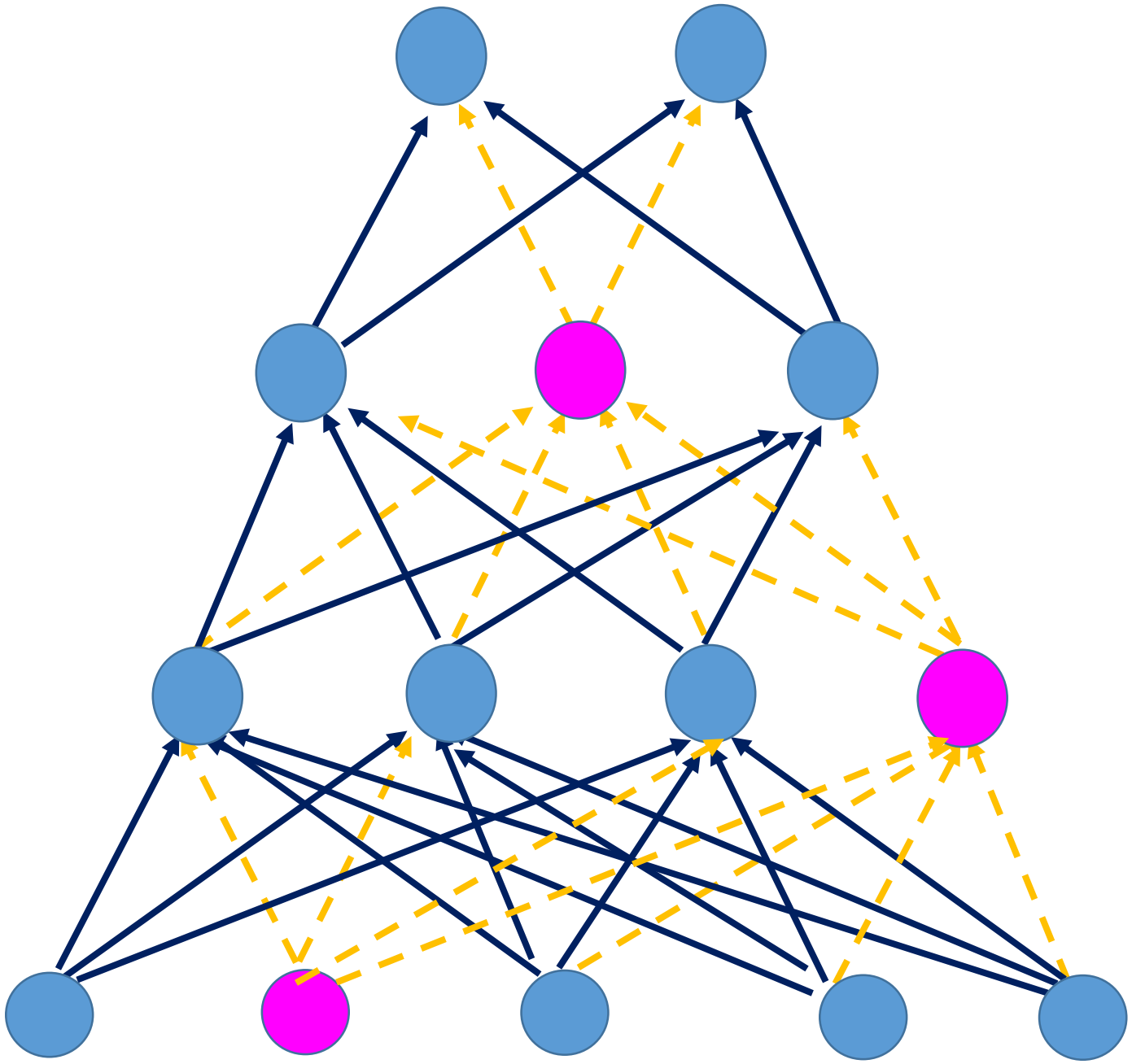


Biological Justification of Neuron Pruning





Human brain prunes more than
5000 brain neurons daily.



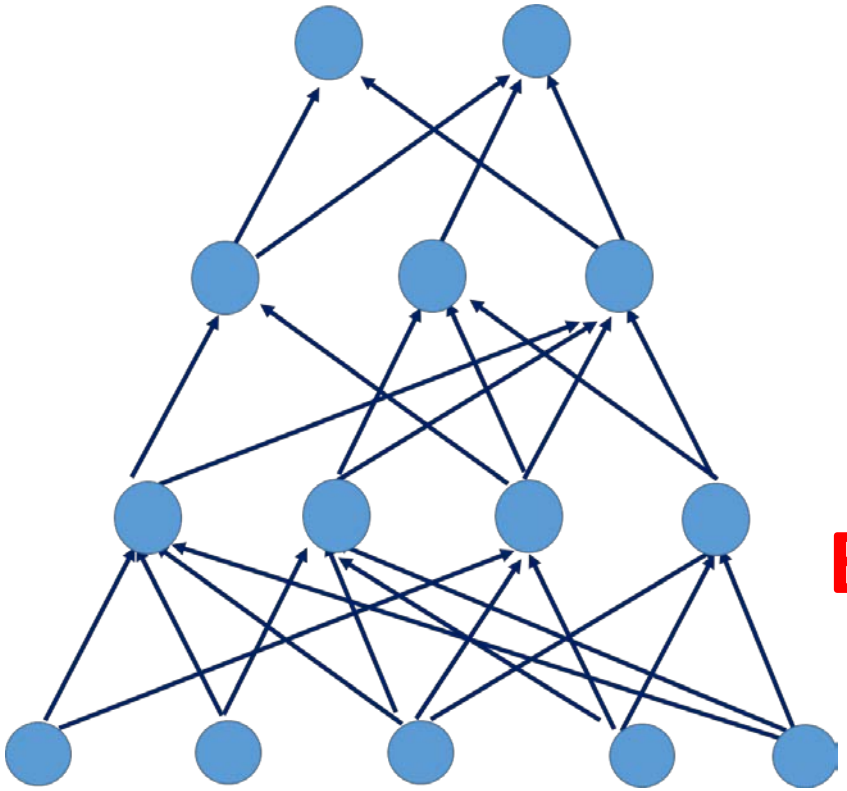
38-10 =28

Sparse
5-4-3-2

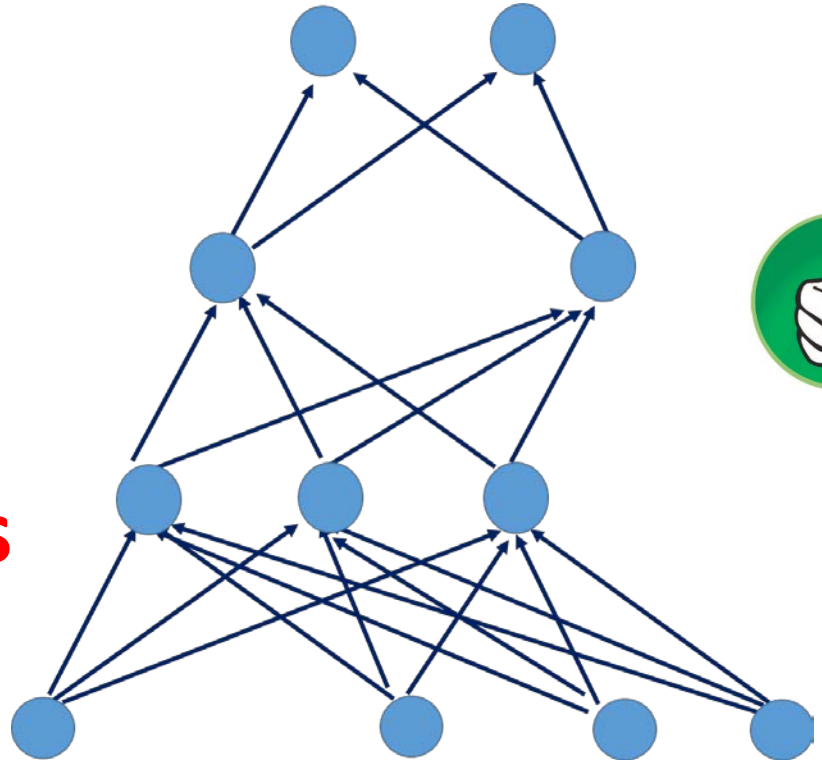
Drop 3
nodes

Dense
4-3-2-2

4×3+3×2+2×2
= 22



Reduction
Effectiveness



Evaluation of Links Vs. Evaluation of Neurons

Accuracy

B(ii): ITL and IOM

For BP learning, all neuron are created equal.

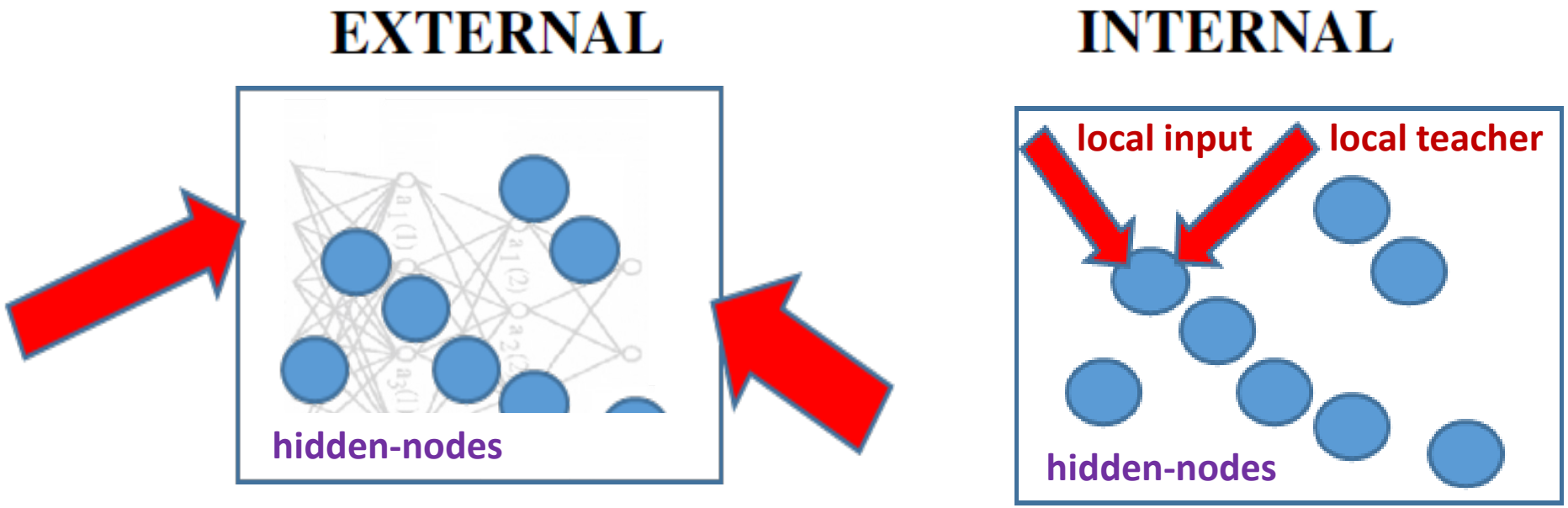
Goal: Ranking of Neurons to facilitate structural gradient.

Internal X-learning for node ranking and structural learning!!

The internal learning paradigm facilitates structural learning.

(1) Internal Teacher Labels (ITL)

For classification problem, the **internal teacher labels** can be metaphorically hidden in "Trojan-horses" and transported (along with the data) from the input layer to all hidden nodes. promising ones:

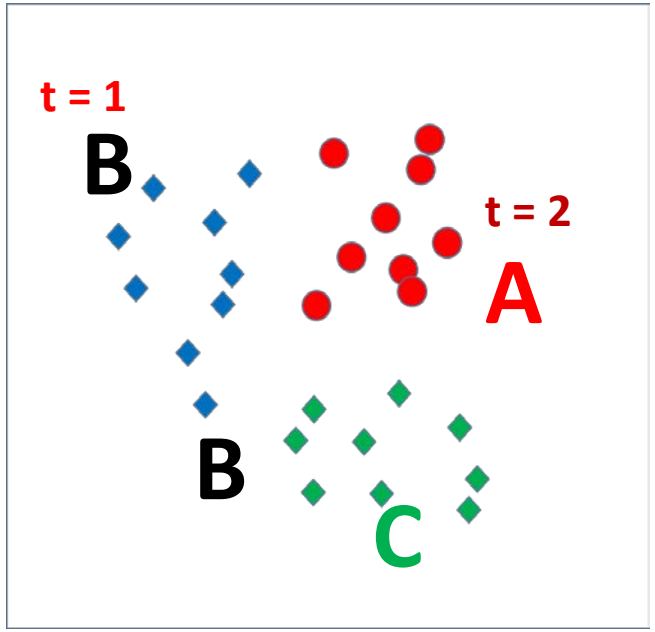


(2) Internal Optimization Metrics (IOM)

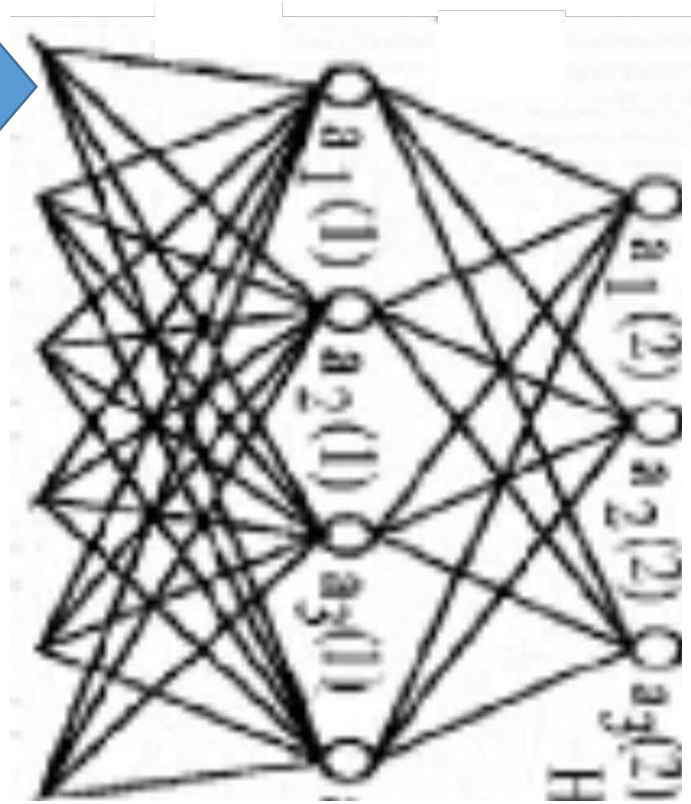
Local Metrics for classification-type

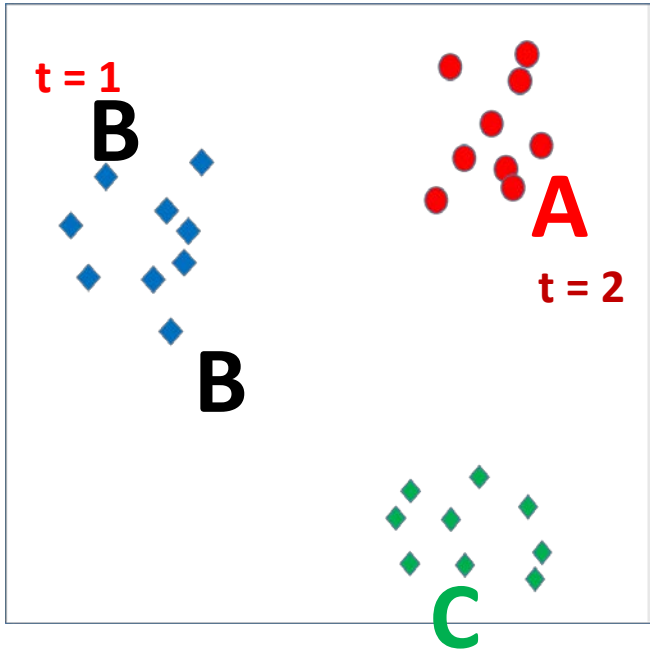
Nonlinear transformations in the neural network is meant to improve the IOM (e.g. DI) from layer to layer.

t = 2 t = 1



Low DI



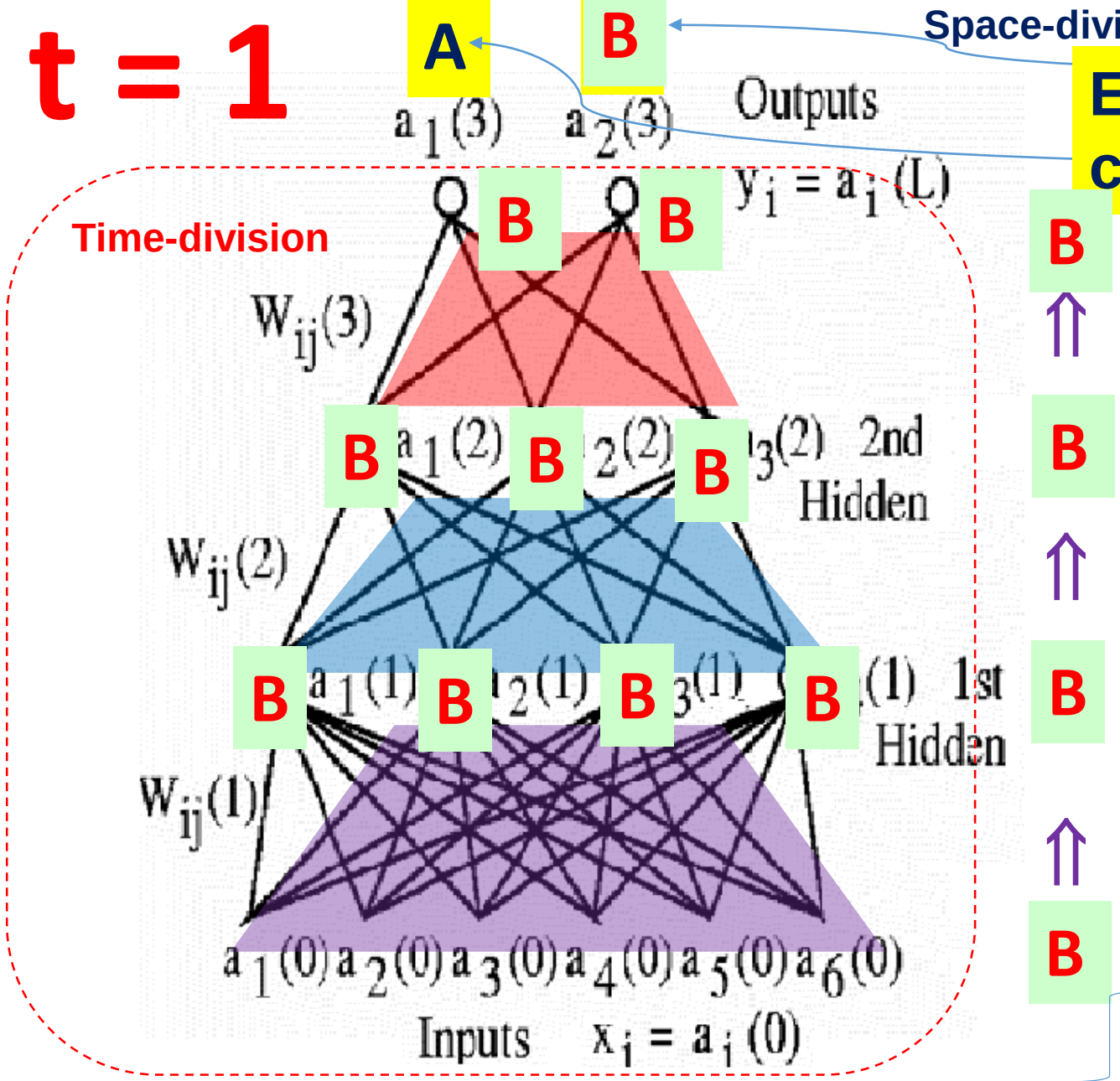


High DI

t = 1

Space-division

ETL is space-divided, the labels correspond to the output nodes.



B ⇑ B ⇑ B ⇑ B

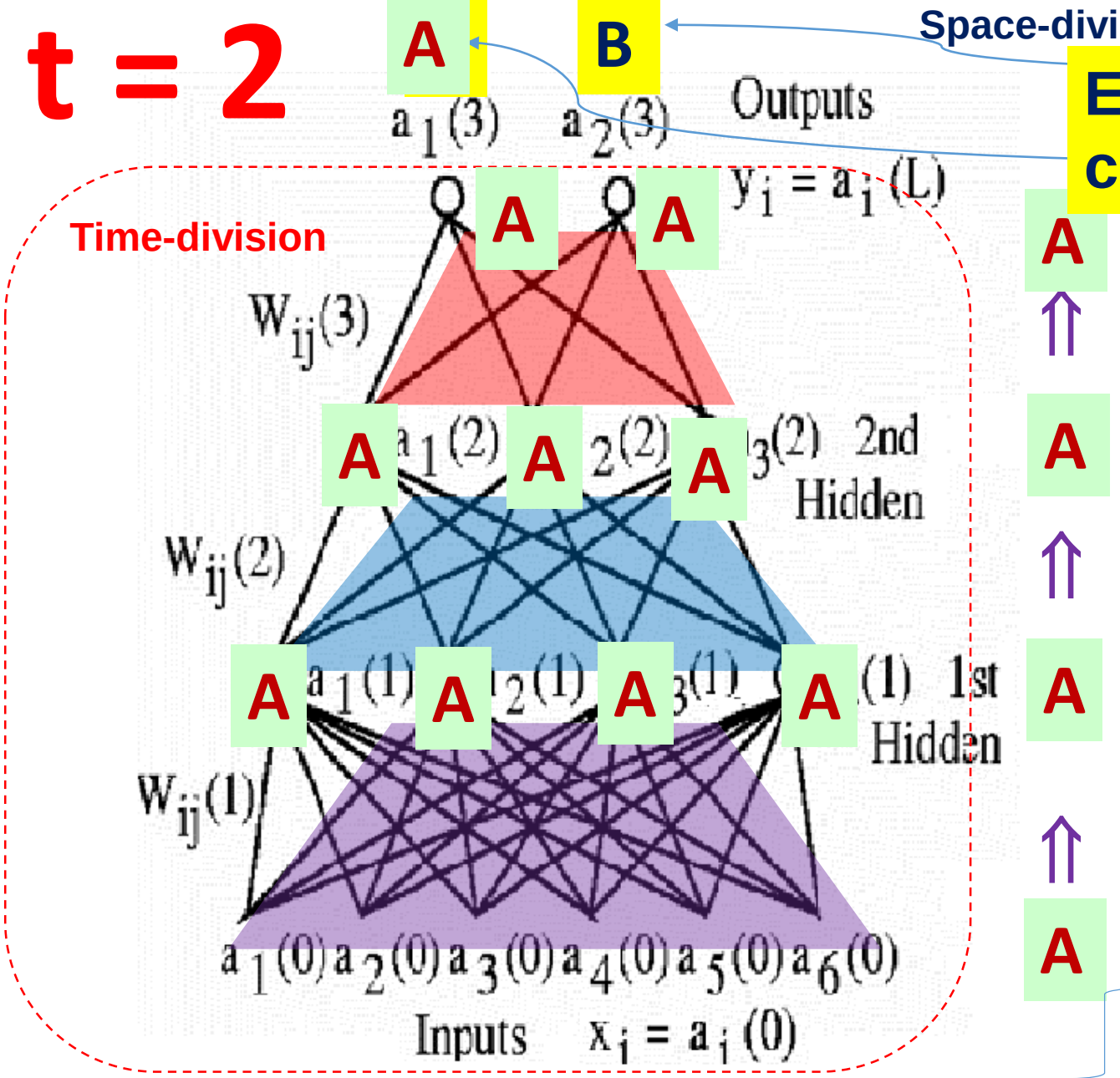
B
A
A
B

In contrast, ITL is time-divided, the labels are assigned to each of incoming input samples at one instance.

t = 2

Space-division

ETL is space-divided, the labels correspond to the output nodes.



A ⇑ A ⇑ A ⇑ A

Data Adaptive BP

B A A B

Batch Method

B(iii): Internal Optimization Metrics (IOM)

Standing on Giant’s Shoulders

Carl Friedrich Gauss – Born 1777
The numerous things named in honor of *Gauss* include: The normal distribution, also known at the *Gaussian* distribution, the most common bell curve in statistics.

Sir Ronald Aylmer Fisher FRS (1890 –1962) was a British statistician and "a genius who almost single-handedly created the foundations for modern statistical science" and "the single most important figure in 20th century statistics".

Claude Elwood Shannon was "the father of information theory". Shannon is noted for having founded information theory with a landmark paper, A Mathematical Theory of Communication 1948.

DI (Discriminant Information)

Three Scatter Matrices

Scatter Matrix $$\bar{\mathbf{S}} \equiv \bar{\mathbf{X}}\bar{\mathbf{X}}^T = \sum^{N} [\mathbf{x}_i - \vec{\boldsymbol{\mu}}][\mathbf{x}_i - \vec{\boldsymbol{\mu}}]^T$$

Between Class Scatter Matrix $$\mathbf{S}_B = \sum_{\ell=1}^{L} N_\ell \left[\vec{\boldsymbol{\mu}}_\ell - \vec{\boldsymbol{\mu}}\right]\left[\vec{\boldsymbol{\mu}}_\ell - \vec{\boldsymbol{\mu}}\right]^T = \Delta \Xi \Delta^T$$

Within Class Scatter Matrix $$\mathbf{S}_W = \sum_{\ell=1}^{L} \sum_{j=1}^{N_\ell} \left[\mathbf{x}_j^{(\ell)} - \vec{\boldsymbol{\mu}}_\ell\right]\left[(\mathbf{x}_j^{(\ell)} - \vec{\boldsymbol{\mu}}_\ell\right]^T$$

C.R. Rao vs. S.Y. Kung

$$\Rightarrow \quad \mathrm{DI} = \mathrm{DI}(\mathbf{I}) = \textbf{trace}\left([\bar{\mathbf{S}} + \rho\mathbf{I}]^{-1}\mathbf{S}_B\right)$$

DI (Discriminant Information)

Numerically, the optimal DCA projection matrix can be derived from the principal eigenvectors of the Discriminant Matrix.

$$\mathrm{DI} = \mathrm{DI}(\mathbf{I}) = \textbf{trace}\left([\bar{\mathbf{S}} + \rho\mathbf{I}]^{-1}\mathbf{S}_B\right)$$

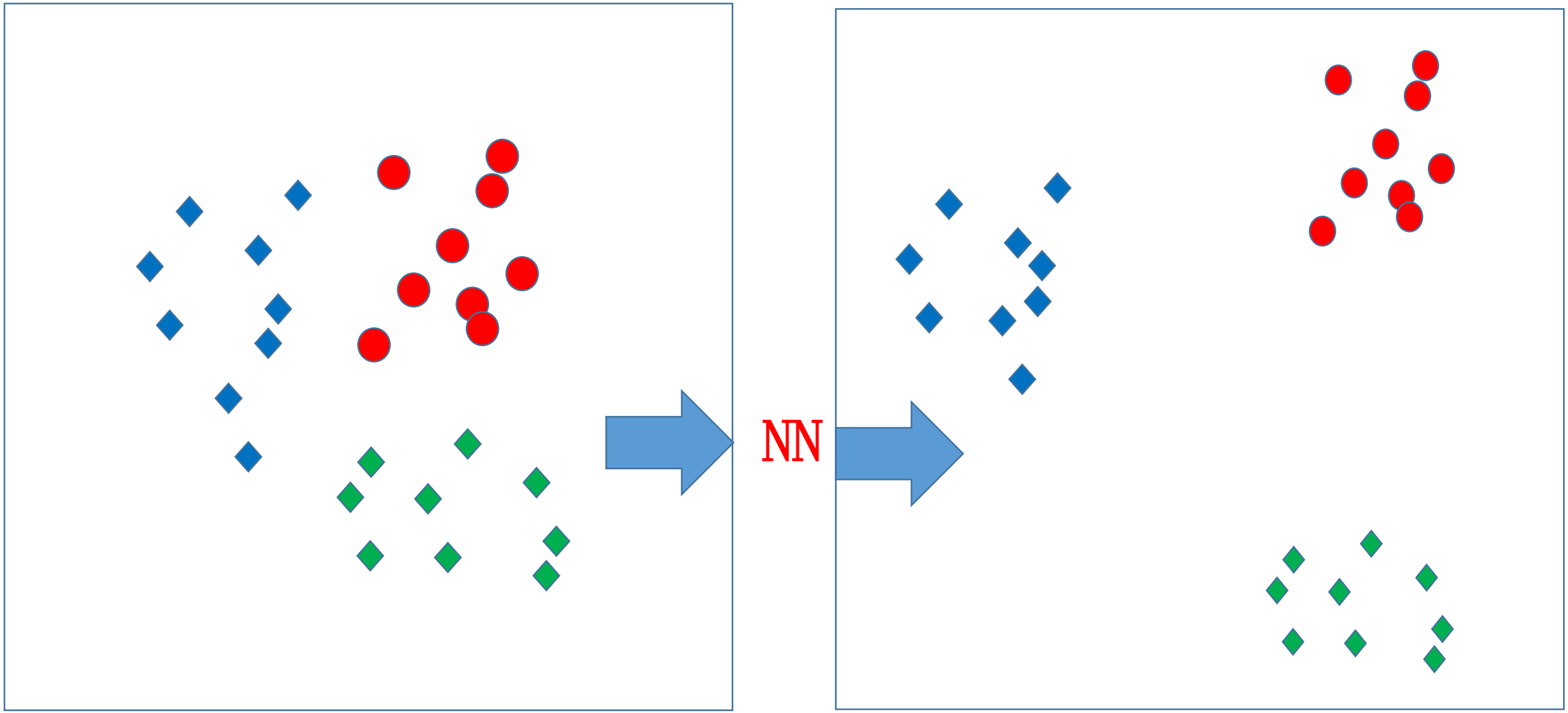
A vital condition is that there exists no inter-component redundancy), a condition termed by C.R.Rao as ``Canonical Orthogonality”. (For DCA, all the eigenvalues are generically distinct, therefore, all the columns of V are canonically orthogonal to each other.)

C.R. Rao vs. S.Y. Kung

- **Rao:** the final score is equal to the product of individual scores, i.e. the volumetric score.
- **Kung** the total score is equal to the sum of individual scores.

Suppose a hidden layer has 2 nodes (or more):

Low-DI Space (2 nodes) → NN → High-DI Space (2 nodes)



C. Iterative X-Learning, DNs, & Results

- DI with SOTP Property
- NP iteration: Joint PS Learning
- Deleterious Nodes (DNs)
- Xnet: MNIST, CIFAR, & ImageNet

B(iii): Internal Optimization Metrics (IOM)

Beauty of Math: A simple change (from **determinant** to **trace**) leads to

`total = sum of individuals`

which then bears offer a vital monotonic increasing property, crucial for using DI to rank the nodes and subspace.

C(i): Subspace/Component DI

$$\mathrm{DI} = \mathrm{DI}(\mathbf{I}) = \mathrm{tr}\left([\bar{\mathbf{S}} + \rho\mathbf{I}]^{-1}\mathbf{S}_B\right)$$

ρ: variance of additive noise

Subspace DI: $\mathrm{DI}(\mathbf{W}) = \mathrm{tr}\left((\mathbf{W}^T\bar{\mathbf{S}}\mathbf{W} + \rho I)^{-1}\mathbf{W}^T\mathbf{S}_B\mathbf{W}\right)$

Three Scatter Matrices

Scatter Matrix $\bar{\mathbf{S}} \equiv \bar{\mathbf{X}}\bar{\mathbf{X}}^T = \sum^{N}[\mathbf{x}_i - \vec{\mu}][\mathbf{x}_i - \vec{\mu}]^T$

Between Class Scatter Matrix $\mathbf{S}_B = \sum_{\ell=1}^{L} N_\ell\,[\vec{\mu}_\ell - \vec{\mu}][\vec{\mu}_\ell - \vec{\mu}]^T = \Delta\Xi\Delta^T$

Within Class Scatter Matrix $\mathbf{S}_W = \sum_{\ell=1}^{L}\sum_{j=1}^{N_\ell}\left[\mathbf{x}_j^{(\ell)} - \vec{\mu}_\ell\right]\left[(\mathbf{x}_j^{(\ell)} - \vec{\mu}_\ell\right]^T$

For node evaluation:

Subspace DI

$$\mathbf{DI}(\mathbf{W}) = tr((\mathbf{W}^T \bar{\mathbf{S}} \mathbf{W} + \rho I)^{-1} \mathbf{W}^T \mathbf{S}_B \mathbf{W})$$

To assess the IOM of the full space of a layer, we set **W**= **I**:

For (supervised) deep compression, we adopt

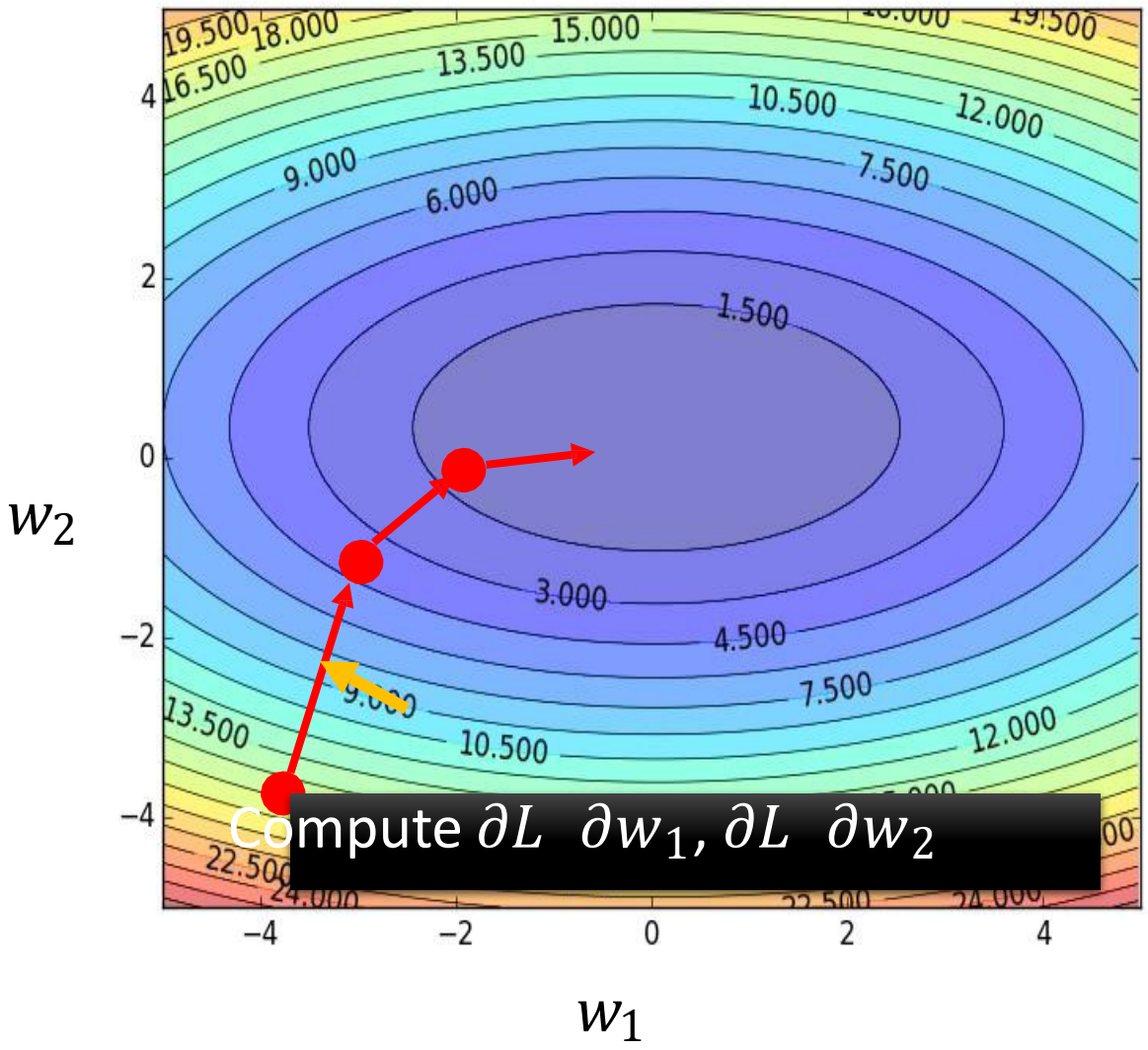
$W_{i\text{-keep}}$ / $W_{i\text{-drop}}$ to keep/drop only the *i-th* node/channel:

$$\mathbf{W}_{i_{keep}} = \begin{bmatrix} 0 & 0 & \cdots & 0 & 0 \\ 0 & \ddots & \cdots & \cdots & 0 \\ \vdots & \cdots & 1 & \cdots & \vdots \\ 0 & \cdots & \cdots & \ddots & 0 \\ 0 & 0 & \cdots & 0 & 0 \end{bmatrix} \qquad \mathbf{W}_{i_{drop}} = \begin{bmatrix} 1 & 0 & \cdots & 0 & 0 \\ 0 & \ddots & \cdots & \cdots & 0 \\ \vdots & \cdots & 0 & \cdots & \vdots \\ 0 & \cdots & \cdots & \ddots & 0 \\ 0 & 0 & \cdots & 0 & 1 \end{bmatrix}$$

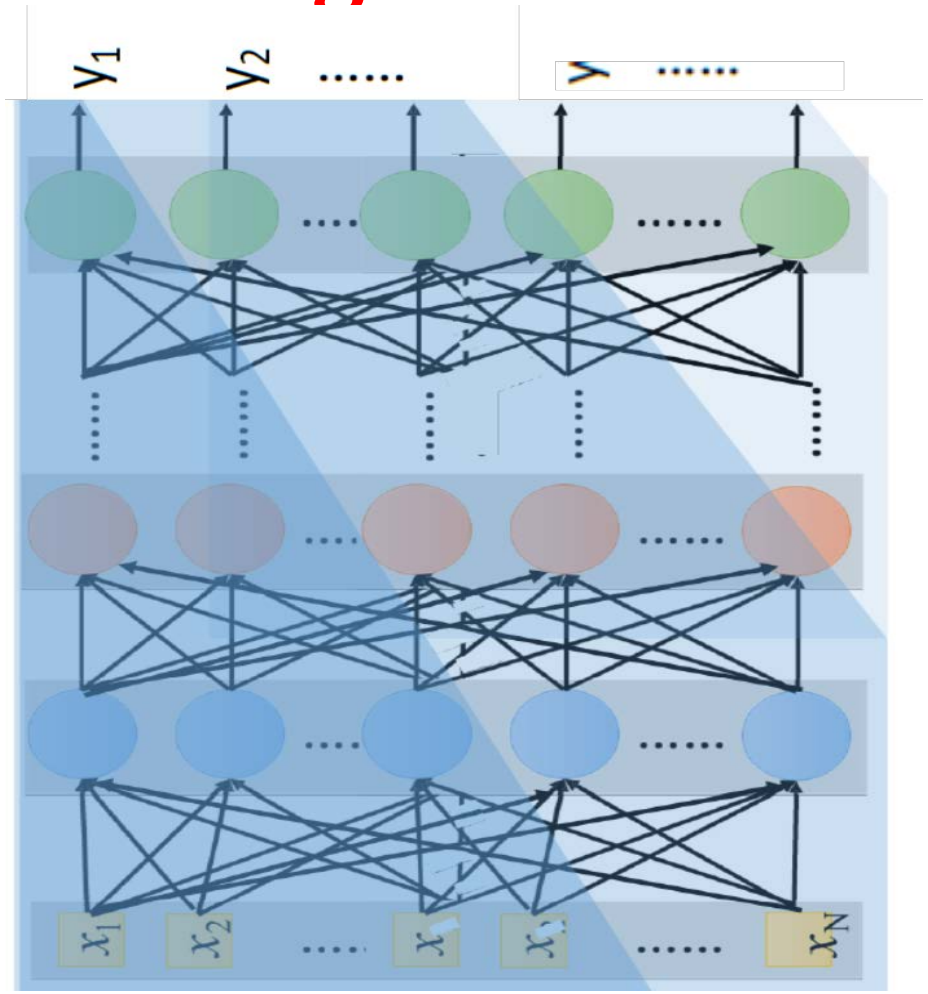
Both Parameter/Structure Gradient-Type Learning

Parameter Learning:

EOM Gradient Descent

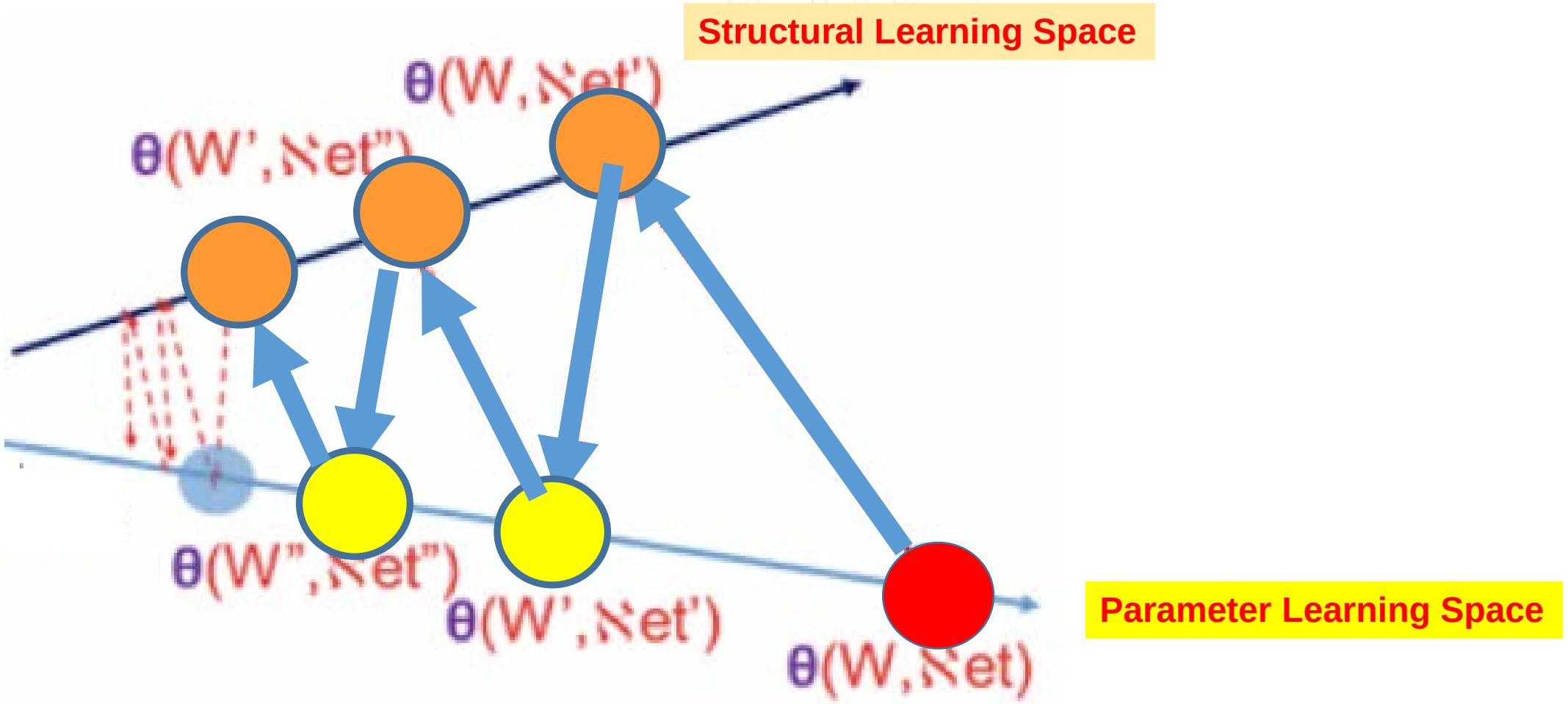


ranking of nodes



NP-Iterative Pruning Method

N: IOM
P: EOM



DI-based Pruning Method Offers Win-Win Deep Compression/Quantization

C(ii): Performance Enhancement via Model Reduction

DN= Deleterious neurons

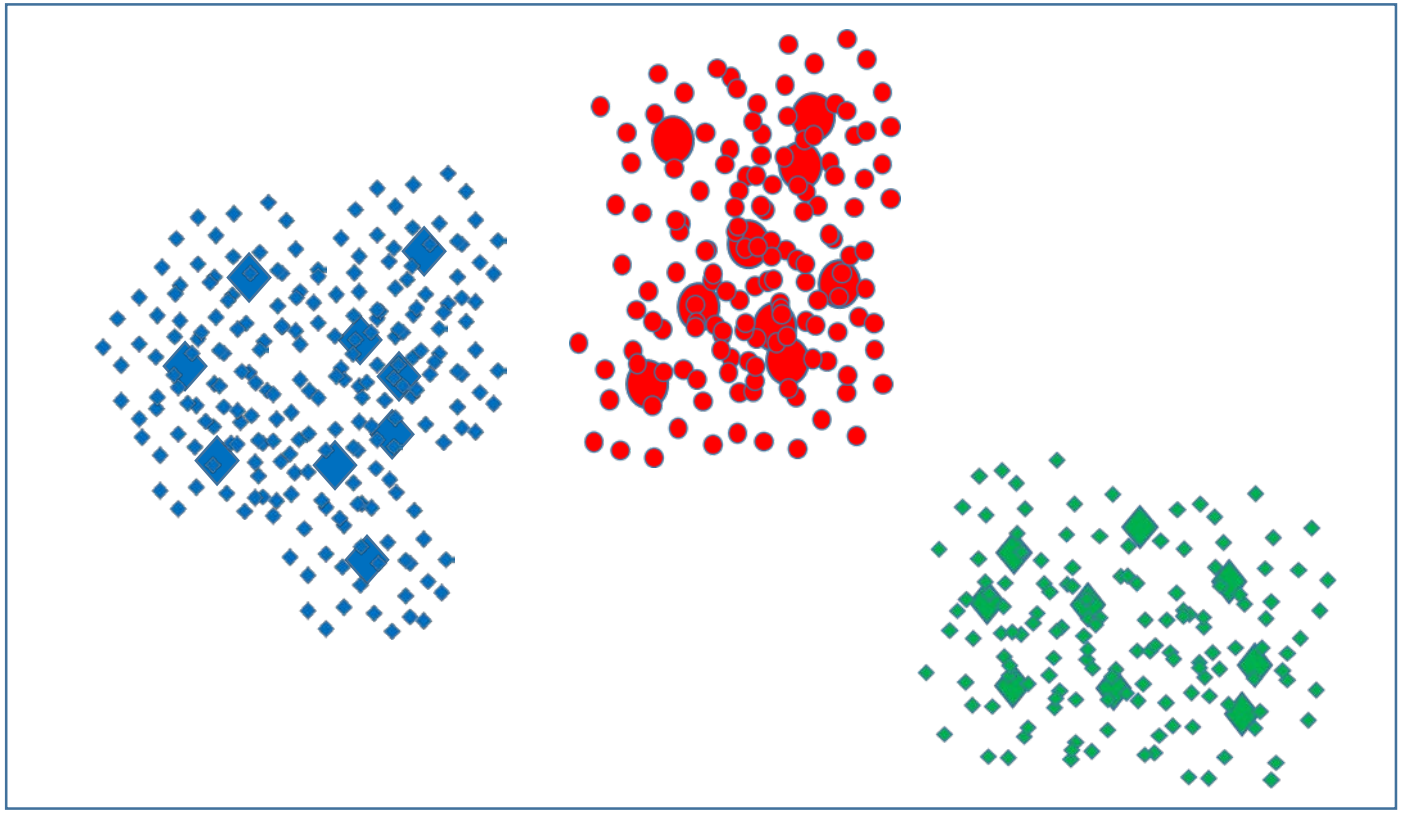
- **Constructive (discriminative) nodes**
- **Redundant (but robust) nodes**
- ***Deleterious nodes***

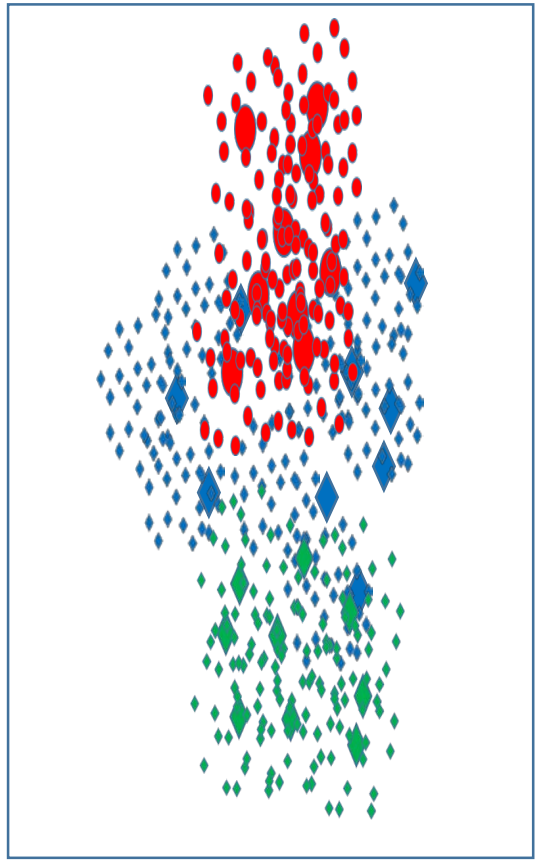
Theme: Finding/Pruning *deleterious neurons* enhances performance.

DN= Deleterious neurons

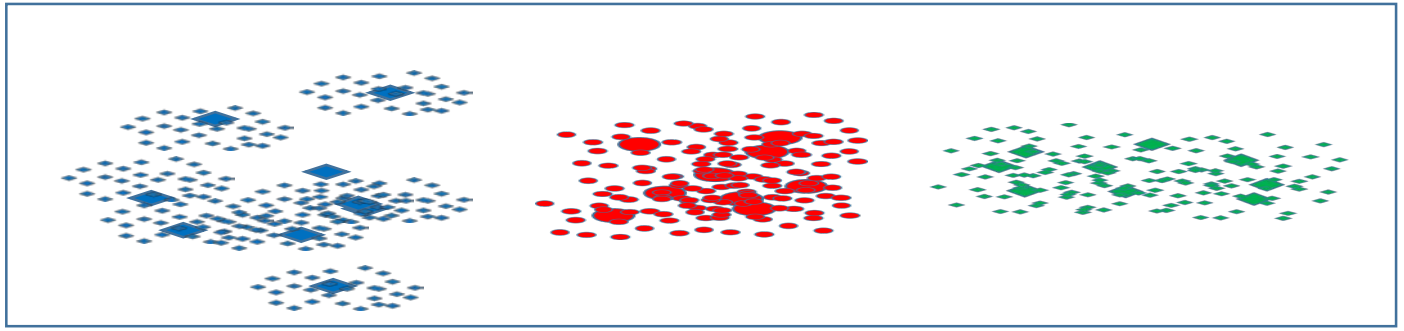
X-learning adopts a joint parameter/structural gradient learning to gradually reduce the network towards an optimal structure, while traversing across the winning structural space.

Differential-DI Subspace (2 nodes: left-vs-right, top-down-neuron)





Low-DI: top-down neuron

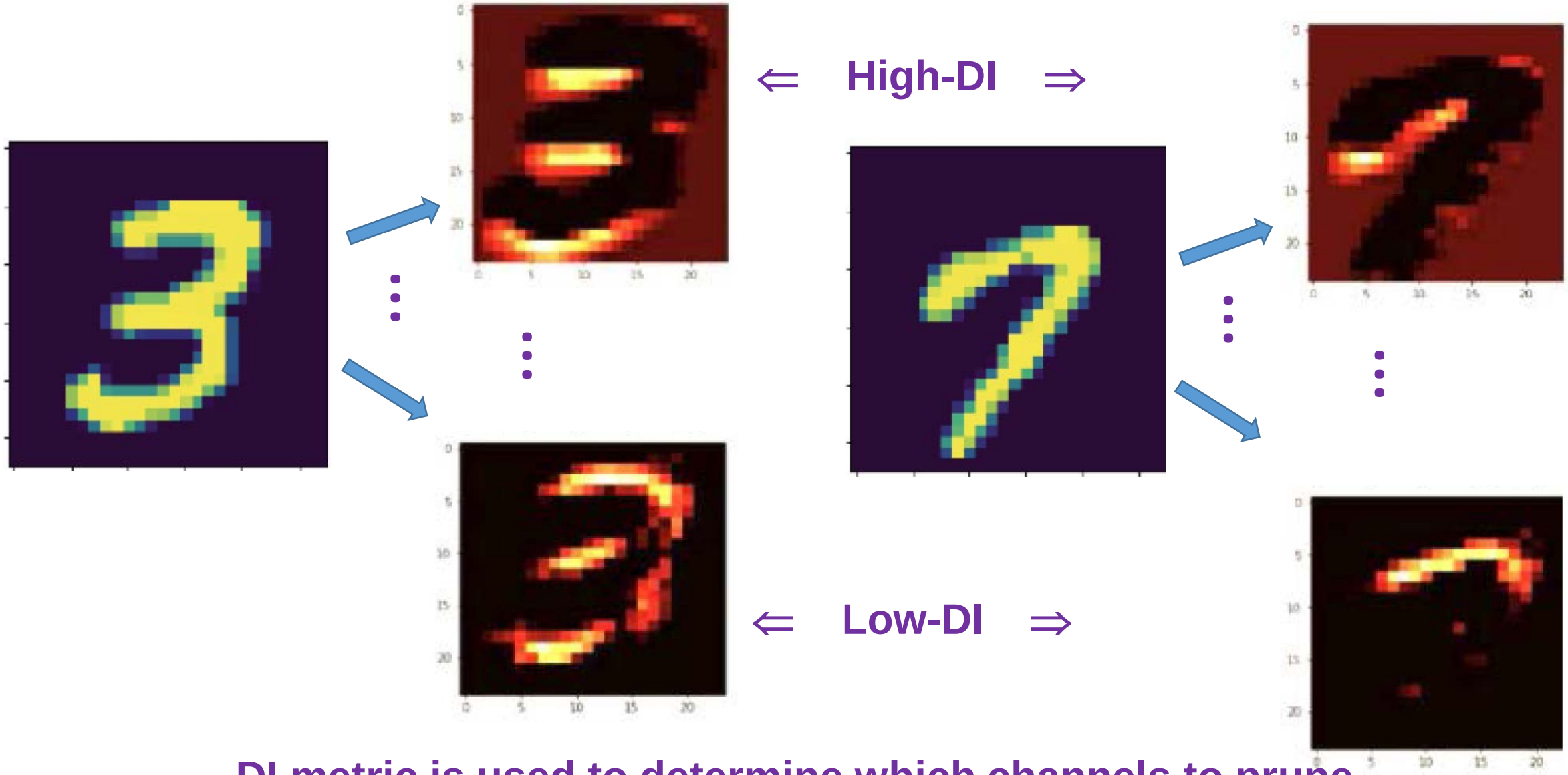


High-DI: left-vs-right neuron

Our punchline:

1 + 1 > 2 → 2 - 1 > 2

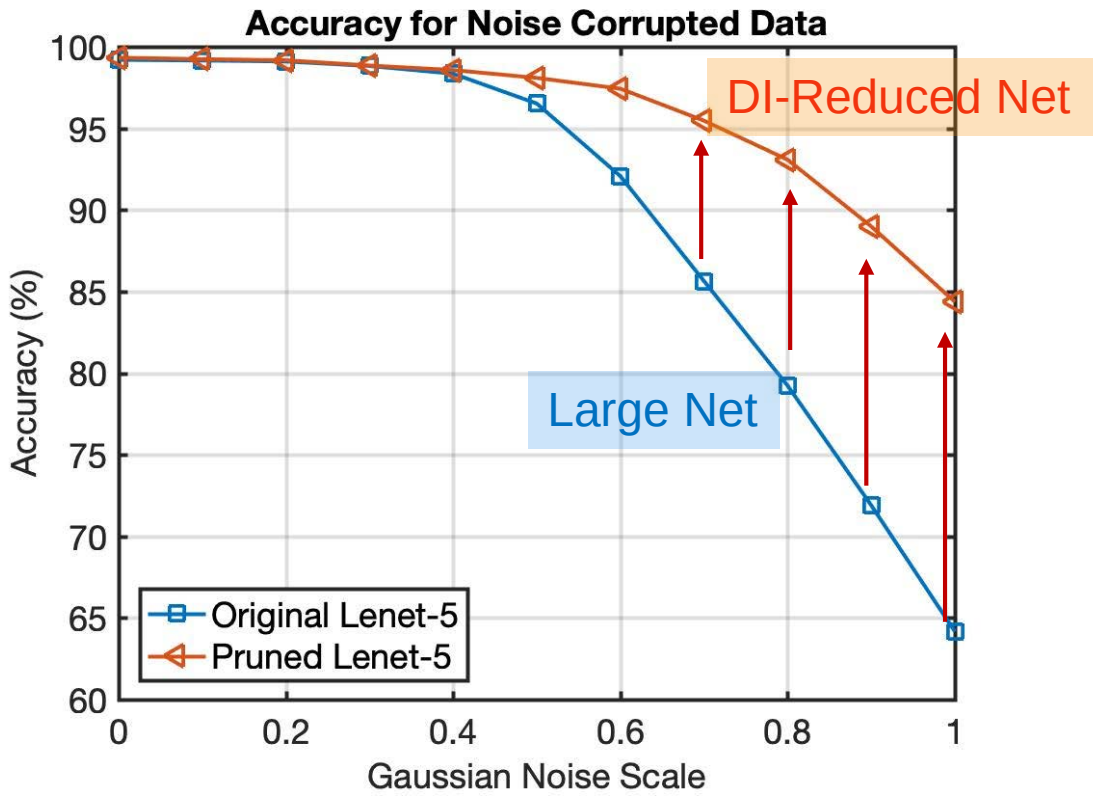
Visualization via Chanel Images [KHL19]



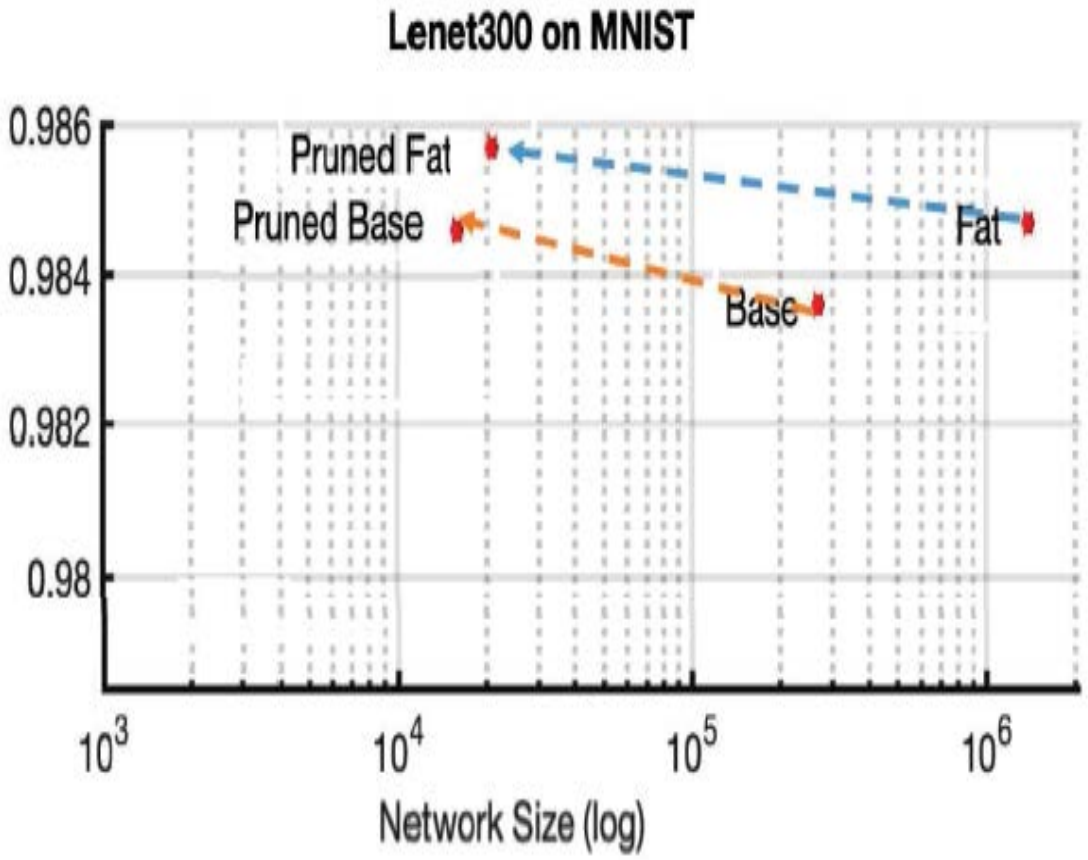
DI metric is used to determine which channels to prune

Deleterious neurons: Pruning enhances robustness

By simply removing redundant and harmful nodes, one can enhance the model robustness. This leads to a joint parameter/structural learning paradigm.



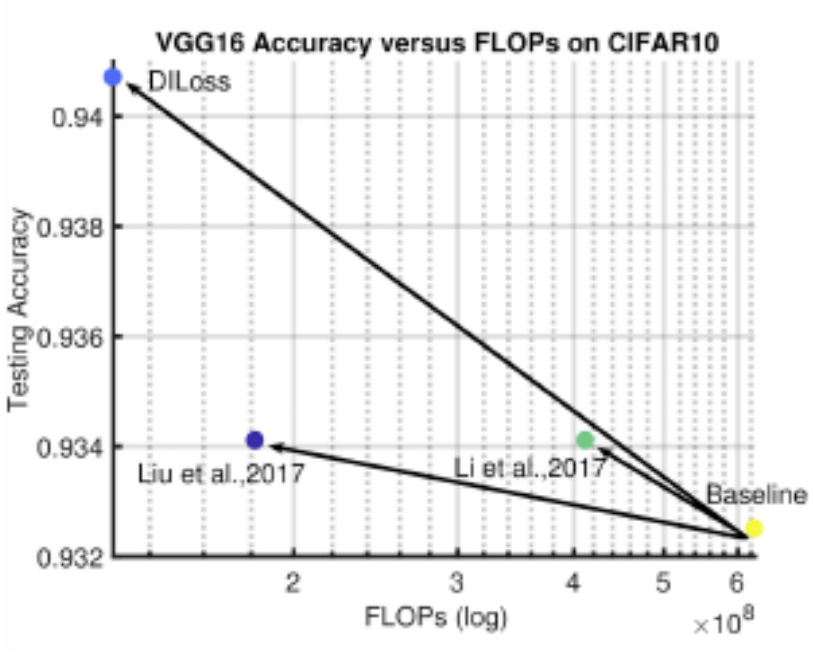
Lenet on MNIST



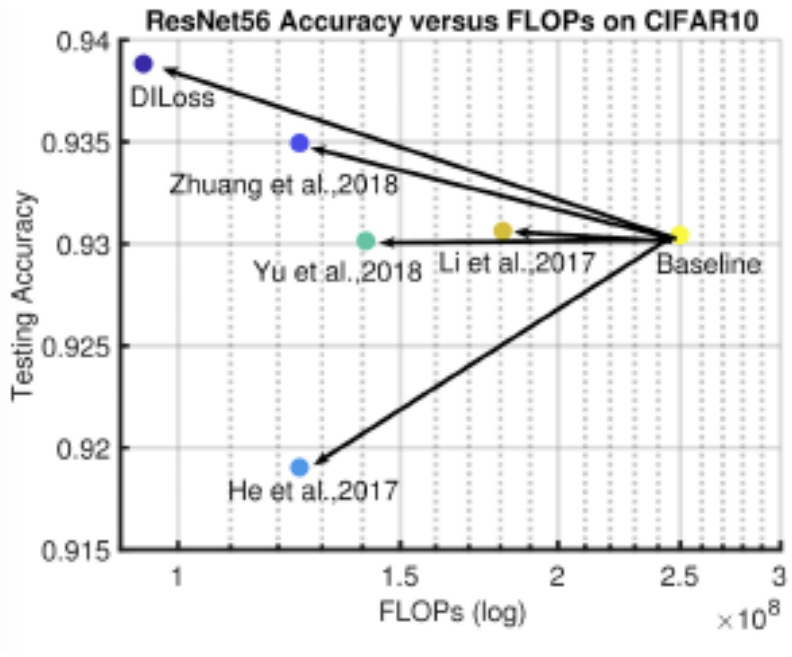
C(iii): Experimental Results

- MNIST, CIFAR Datasets
- ImageNet
- LPIRC

CIFAR-10: Speedup



(c) CIFAR10: VGG16 speedup

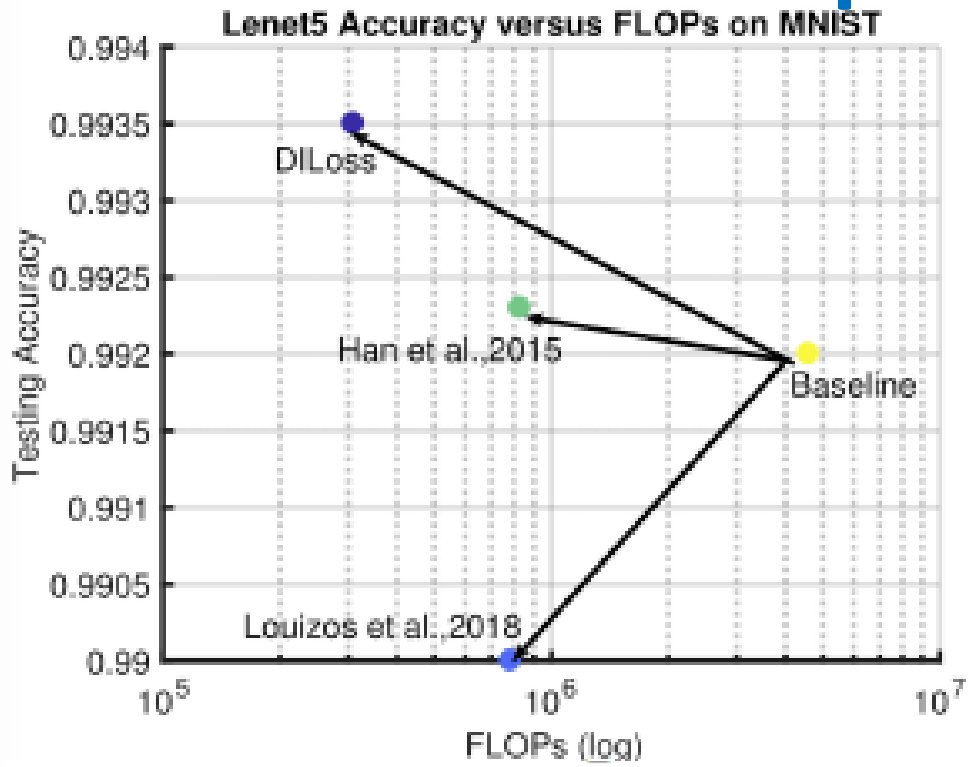


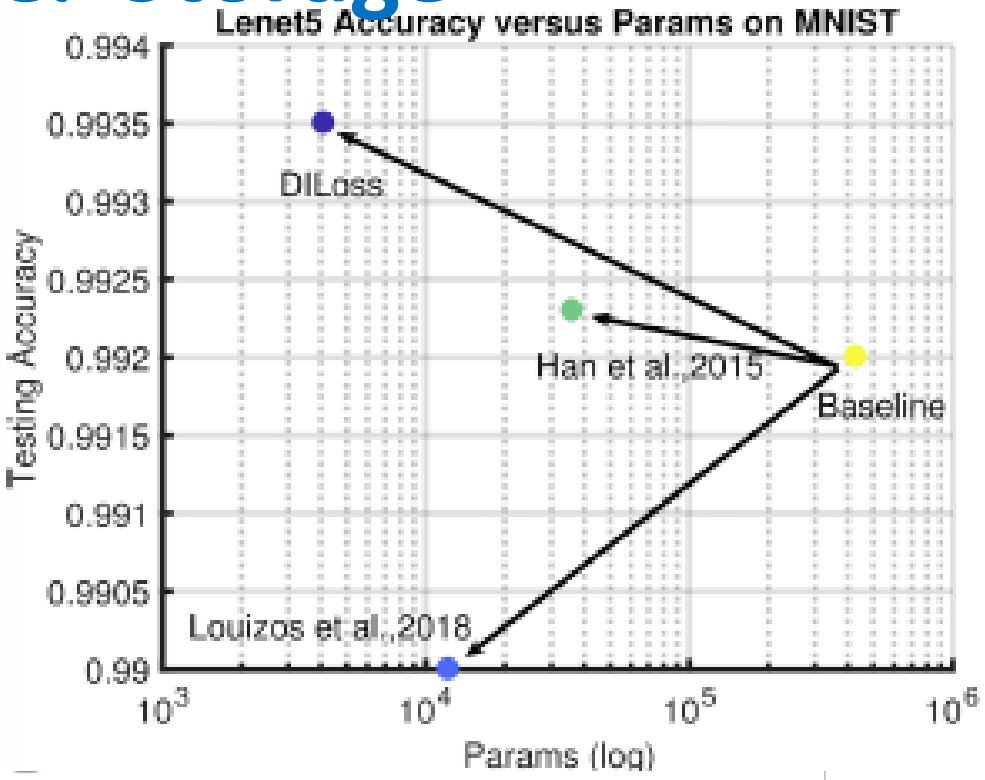
(d) CIFAR10: Resnet56 speedup

Task	Models	Accuracy %	FLOPs	Params
CIFAR-10	*VGG-16 (baseline)*	93.25	6.26×10^8	1.5×10^7
	VGG-16 (Li et al., 2017)	93.41	4.12×10^8(65.81%)	5.4×10^6(36%)
	VGG-16 (FDR)	93.61	1.35×10^8(21.4%)	7.1×10^5(4.7%)
	VGG-16 (DILoss)	**94.07**	$\mathbf{1.28\times10^8}$**(20.45%)**	$\mathbf{5.32\times10^5}$**(3.55%)**
	ResNet-56 (baseline)	93.04	2.5×10^8	8.5×10^5
	ResNet-56 (Li et al., 2017)	93.06	1.81×10^8(72.4%)	7.3×10^5(85.88%)
	ResNet-56 (Yu et al., 2018)	93.01	1.41×10^8(56.4%)	4.94×10^5(58.12%)
	ResNet-56 (He et al., 2017)	91.9	1.25×10^8(50%)	-
	ResNet-56 (Zhuang et al., 2018)	93.49	1.25×10^8(50.25%)	4.3×10^5(50.76%)
	ResNet-56 (DILoss)	**93.84**	$\mathbf{8.38\times10^7}$**(33.52%)**	$\mathbf{3.12\times10^5}$**(35.52%)**
	ResNet-56 (bootstrap:DILoss+Zhuang)	**93.84**	$\mathbf{7.58\times10^7}$**(30.32%)**	$\mathbf{2.81\times10^5}$**(33.05%)**

10×

MINIST: Speedup & Storage



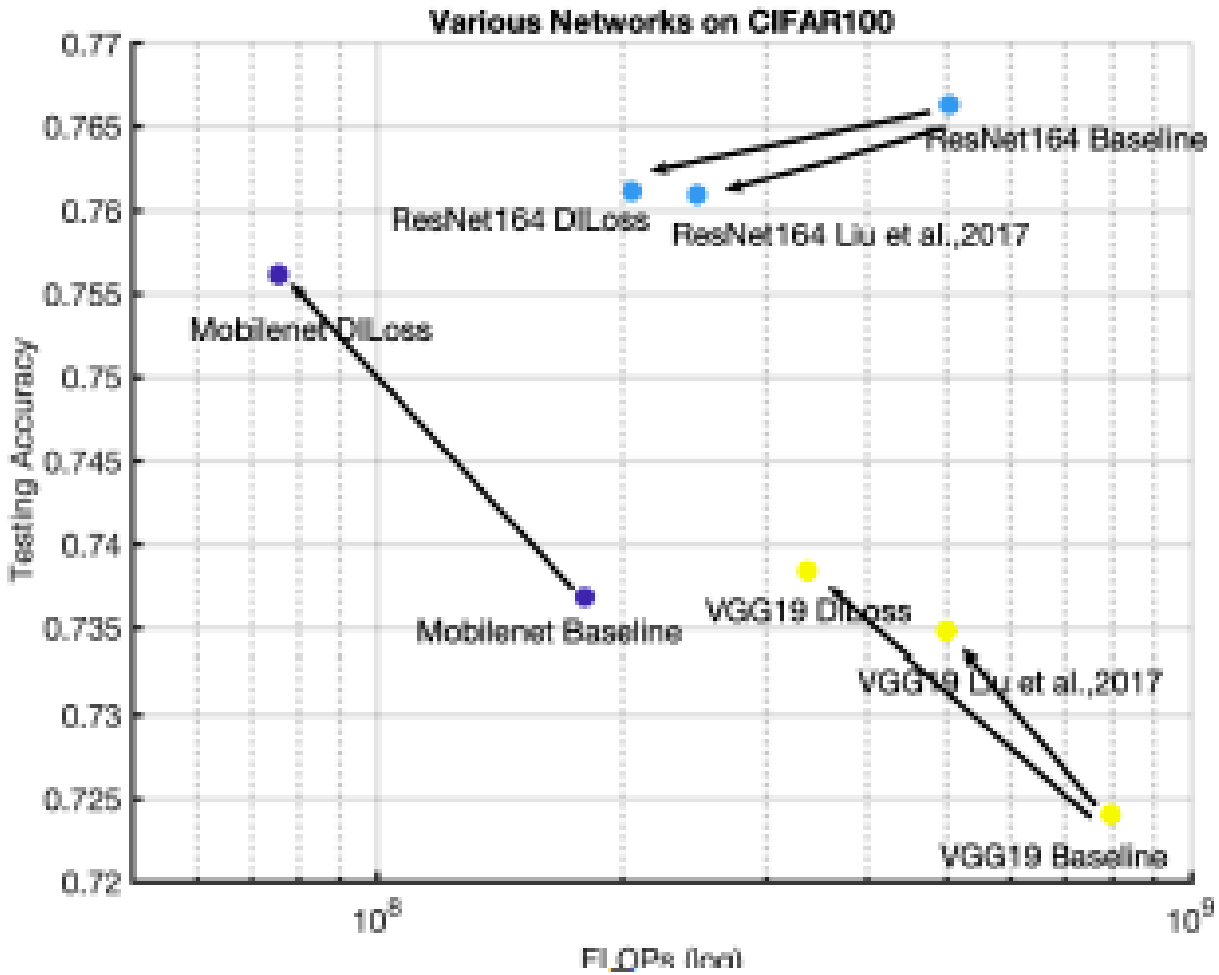


Task	Models	Accuracy %	FLOPs	Params
MNIST	*Lenet-5 (baseline)*	99.2	4.59×10^6	4.3×10^5
	Lenet-5 (Han et al., 2015)	99.23	8.3×10^5(18.1%)	3.6×10^4(8.4%)
	Lenet-5 (Louzois et al., 2018)	99	7.85×10^5(17.1%)	1.22×10^4(2.83%)
	Lenet-5 (FDR)	99.33	2.6×10^5(5.74%)	4.9×10^3(1.1%)
	Lenet-5 (DILoss)	**99.35**	**2.46×10^5(5.36%)**	**3.86×10^3(0.89%)**
	Lenet-300 (baseline)	98.36	-	2.7×10^5
	Lenet-300 (Han et al., 2015)	98.41	-	2.24×10^4(8.3%)
	Lenet-300 (Louzois et al., 2018)	98.2	-	2.7×10^4(10%)
	Lenet-300-100 (FDR)	98.42	-	2.3×10^4(8.5%)
	Lenet-300 (DILoss)	**98.46**	-	**1.63×10^4(6.04%)**

100×

Basic Xnet: Layer-Uniform (Layer-Independent) ITL

CIFAR-100: Speedup

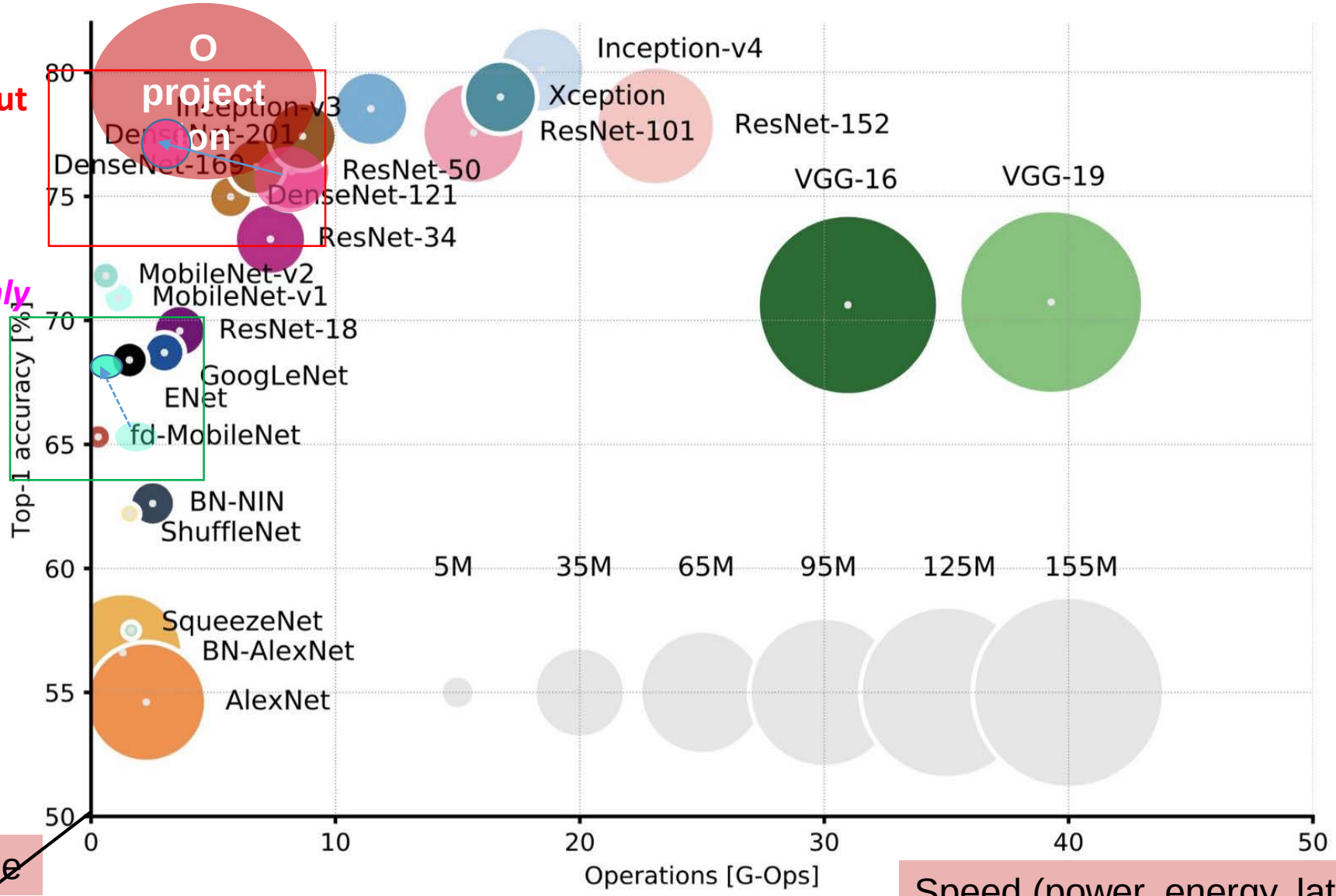


Task	Models	Accuracy %	FLOPs	Params
CIFAR100	*Mobilenet-v2 (baseline)*	73.68	1.8×10^8	2.4×10^6
	Mobilenet-v2 (DILoss)	**75.61**	$\mathbf{7.57\times10^7(42.06\%)}$	$\mathbf{1.07\times10^6(44.58\%)}$

5×* (ITL)

X-learning can spot

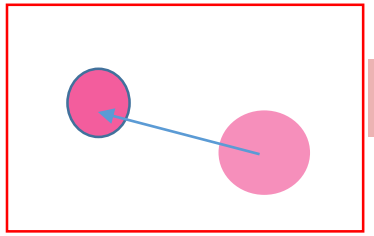
- **Many a Redundant (but robust) nodes**
- ***Very few Highly Deleterious nodes***



Accuracy

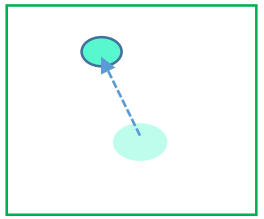
Storage

Speed (power, energy, latency)



Accuracy Winner

Model	Top-1 Acc.	FLOPs	Param.
ResNet-50	75.15%	4.09B	25.6M
Ours-1	76.34%	2.63B	17M
SSS (Huang et al., 2018)	71.82%	2.33B	15.6M
ThiNet (Luo et al., 2017)	72.04%	2.44B	16.94M
GDP (Lin et al., 2018)	72.61%	2.24B	-
SFP (He et al., 2018)	74.61%	2.42B	-
Ours-2	76.1%	2.31B	16.72M
CP (He et al., 2017)	72.26%	2B	-
DCP (Zhuang et al., 2018)	74.95%	2B	-
Ours-3	75.6%	2B	15.6



Speed Winner (power, energy, latency)

Model	Top-1 Acc.	FLOPs	Param.
MobileNetV1	70.2%	569M	4.24M
WM (Sandler et al., 2018)	68.4%	325M	-
NetAdapt (Yang et al., 2018)	69.1%	285M	-
Ours	70.2%	285M	2.28M

Model	Top-1 Acc.	FLOPs	Param.
MobileNetV2	71.8%	300M	3.47M
WM-1 (Sandler et al., 2018)	69.8%	210M	2.61M
Ours-1	70.8%	210M	2.33M
1st place (LPIRC Challenge, 2018)	65.2%	186M	-
WM-2 (Sandler et al., 2018)	65.4%	170M	2.57M
ThiNet (Luo et al., 2017)	65.44%	170M	2.57M
DCP (Zhuang et al., 2018)	65.91%	170M	2.57M
Ours-2	68.2%	170M	1.9M

D. Extension

- **LAXnet: Adaptive (Layer-dependent) ITL**
  - **From coarse ITL to fine ITL**
- **EXnet: End-User Explainable Learning ITL**

DI(A) for A or $\bar{A}$ (=not A)

DI(A) and DI(A') for a special-interest group such as:

- $\bar{A}$ but A'
- A but $\bar{A}$'

- **Regression/Generation**
  - **Super-Resolution Image Enhancement**

DARPA's XAI ⇒ X-Learning

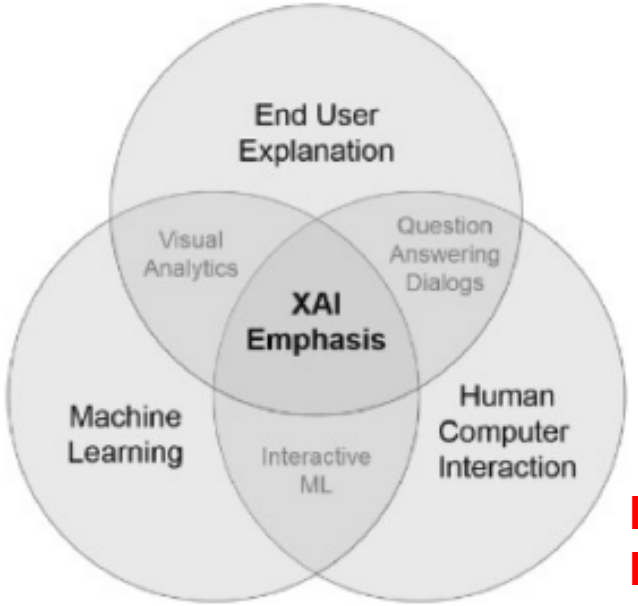
Broad Agency Announcement

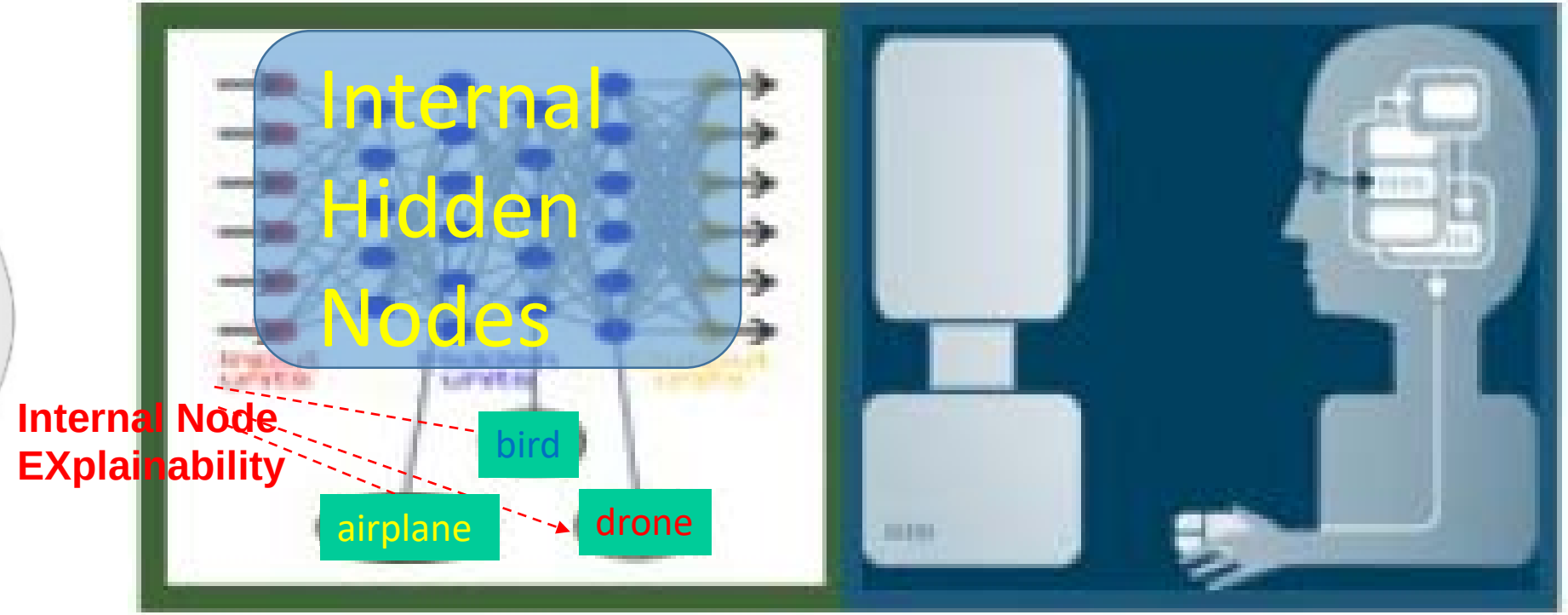
Explainable Artificial Intelligence (XAI)

DARPA-BAA-16-53

August 10, 2016

End User Explainability: ``XAI is most interested in explanations of higher-level decisions that would be relevant to the end user and the missions he/she needs to manage."[DARPA2016]





Motivated by XAI, X-learning will play vital roles in XAI.

Our Approach to XAI: EXnet

Bird

or Drone

EXnet's Approach to XAI

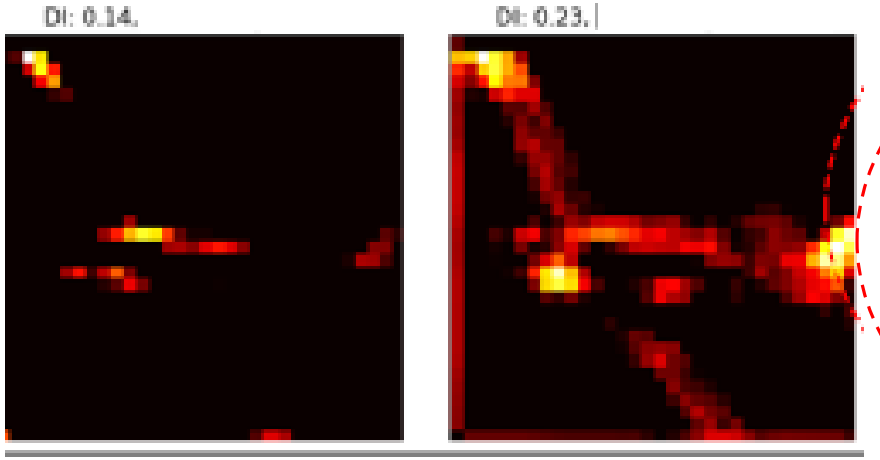
ABC Example

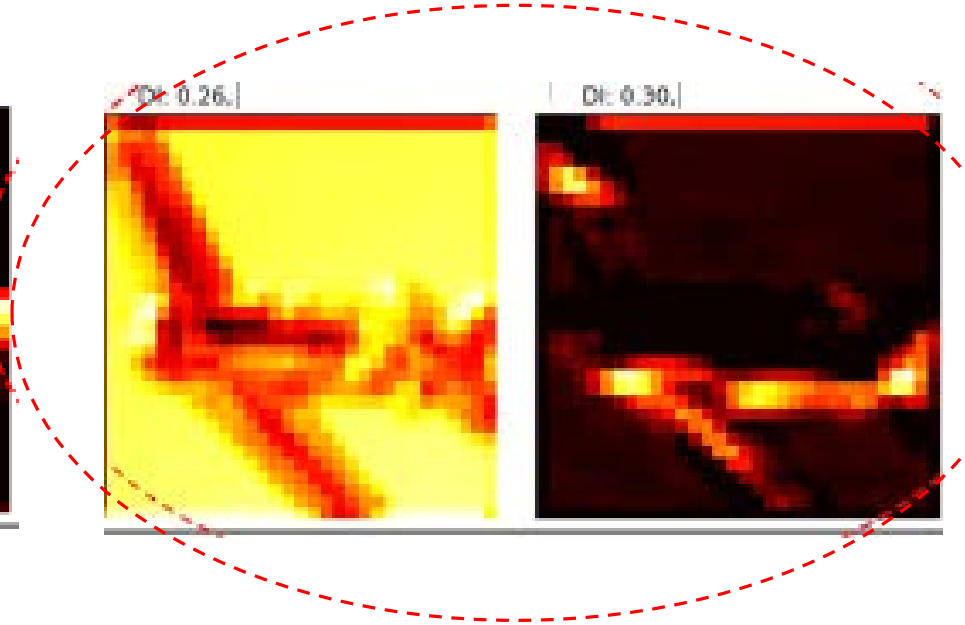
End-user: (subset = top 3 ABC classes):

A= airplane, B= bird, and C= car

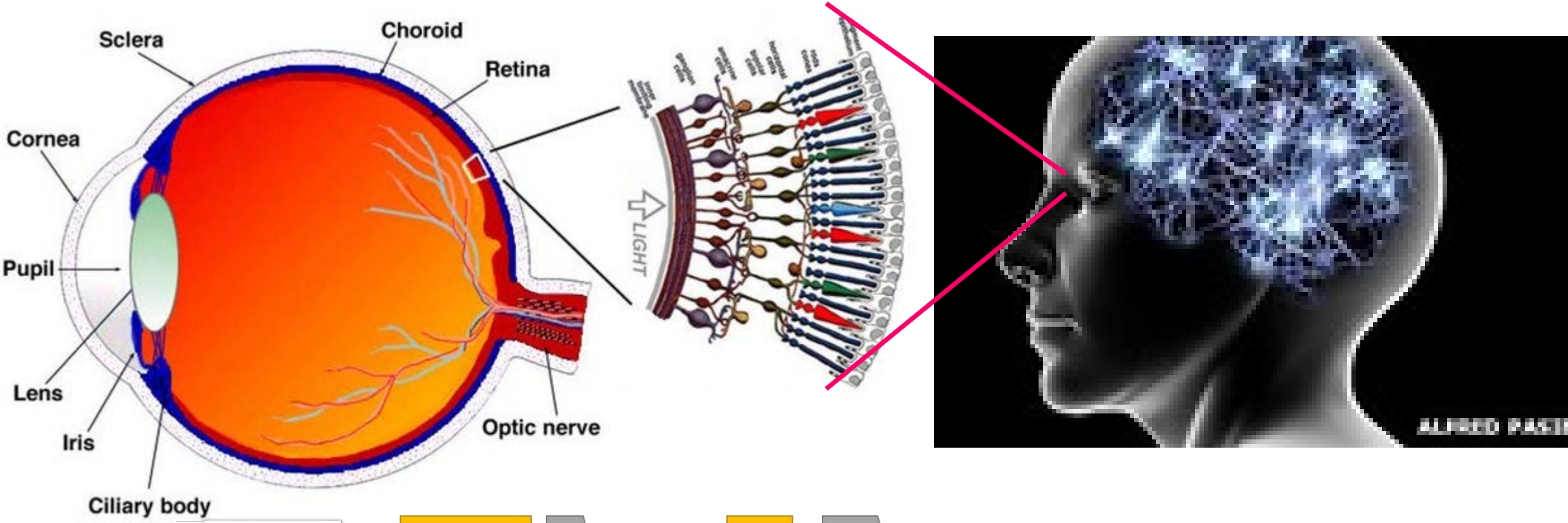
From Low DI to High DI channels (neurons)

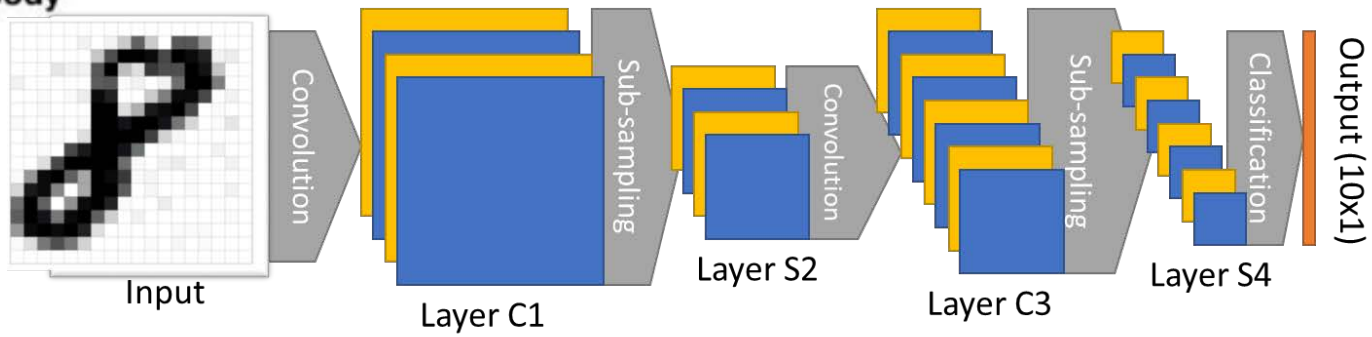
Conv1





LAX-Learning: Different Teachers for Different Layers

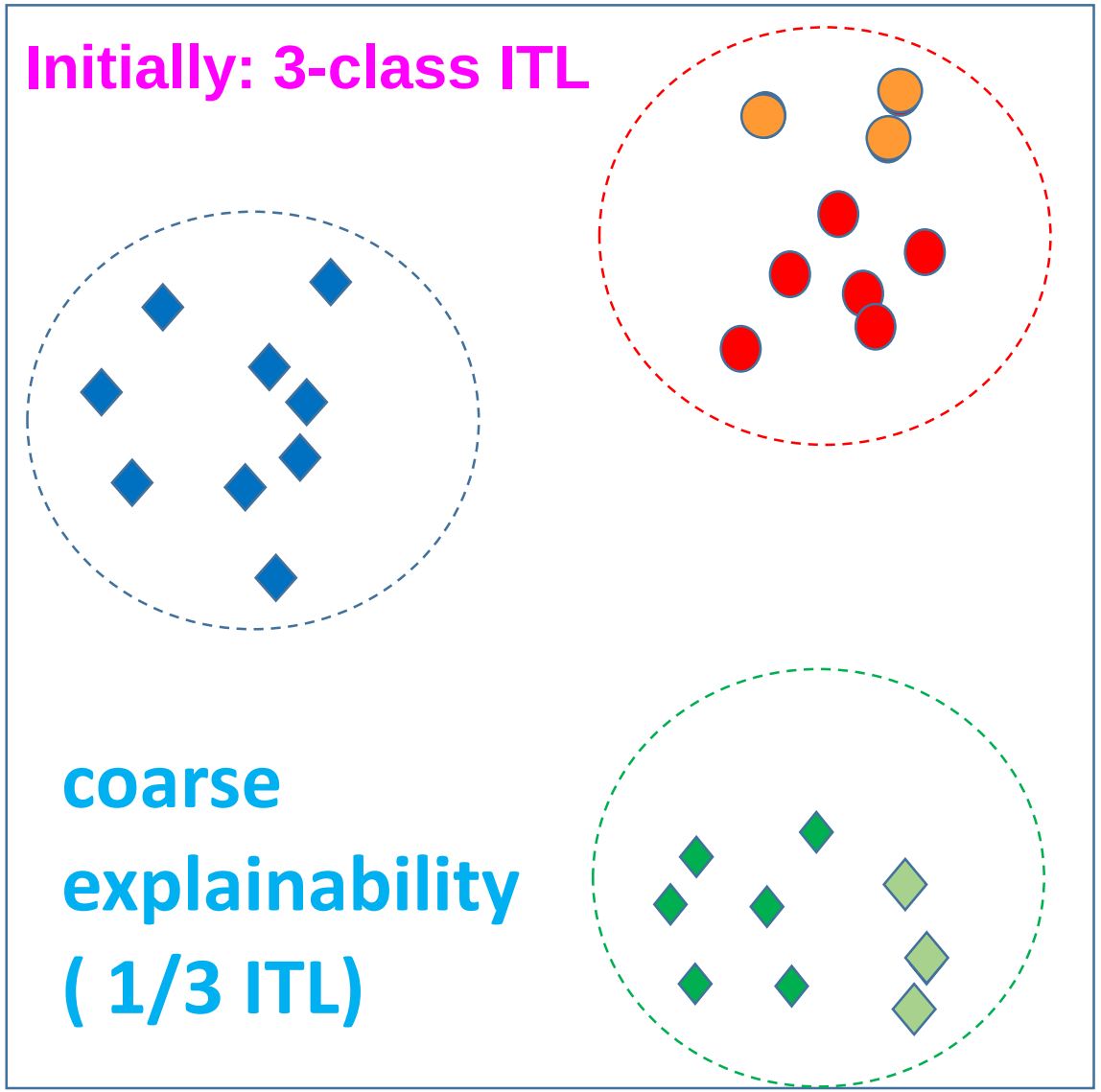


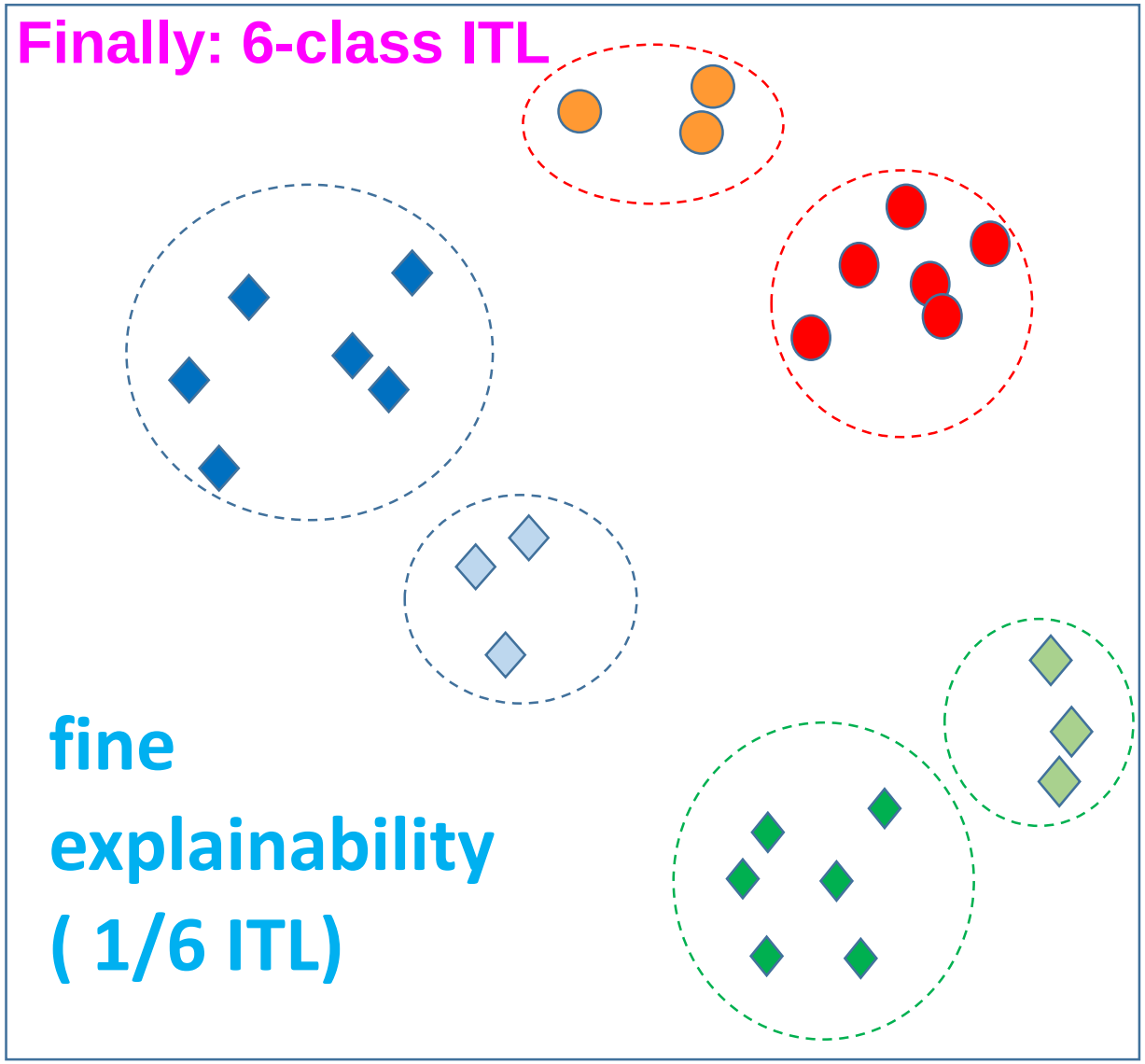


Layer Structure in (Optical Preprocessing) Retina Nervous Systems

Multi-resolution LAX-Learning

Initial tantalizing layers → Final maturing layers

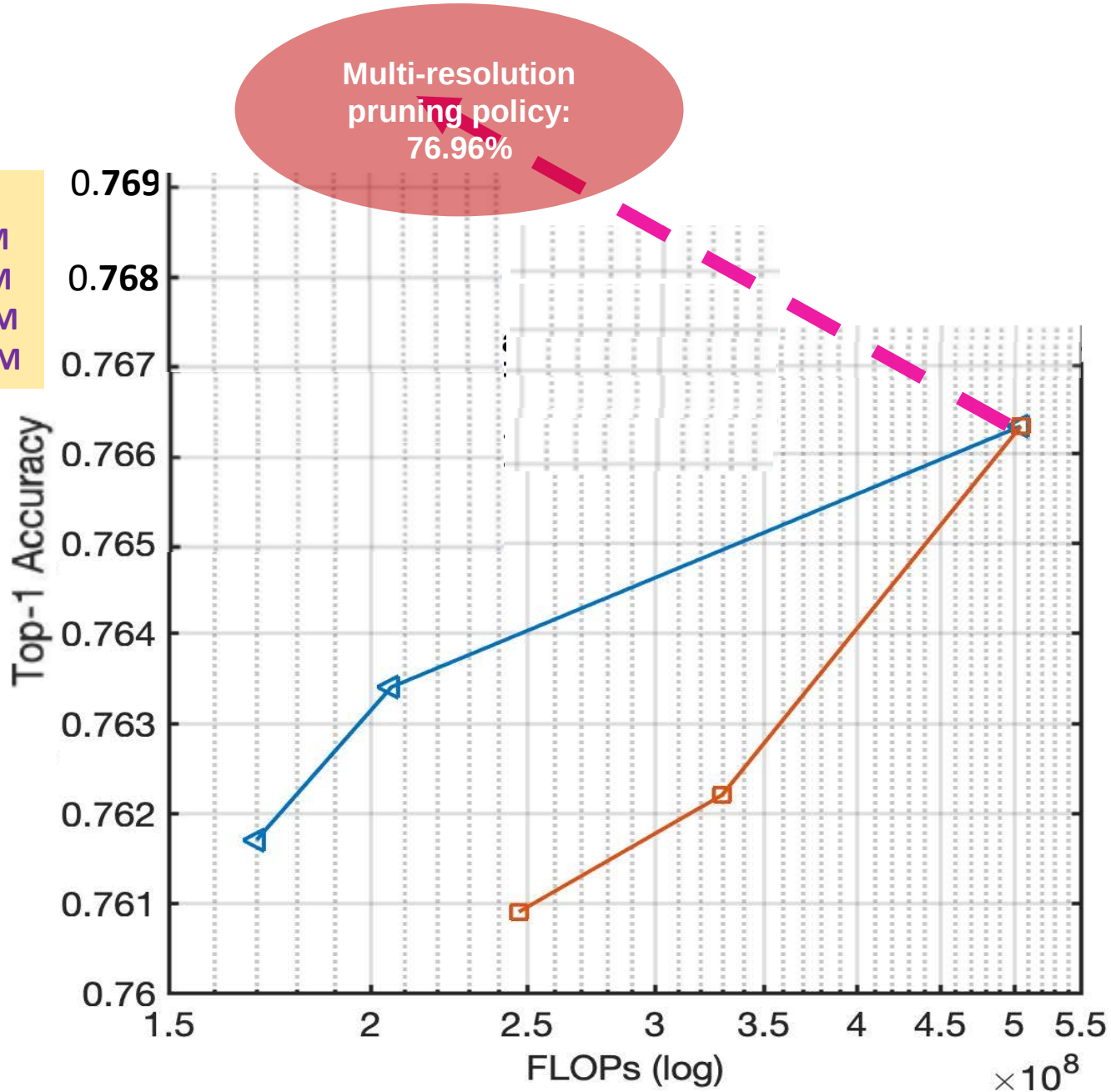




LAXnet: Layer-Adaptive (Layer-dependent) ITL

Multi-resolution X-Learning

Model	Accuracy	FLOPs	Size
ResNet-164 (Baseline)	76.63%	505M	1.7M
ResNet-164 (Liu et al., 2017)	76.09%	247M	1.2M
X-ResNet-164 (Uni-resolution)	76.11%	210M	0.95M
X-ResNet-164 (multi-resolution)	76.96%	210M	0.95M



AXnet improves the accuracy by **0.85%** over the basic Xnet.

Regression Analysis:
RX-Learning

JPEX Examples on

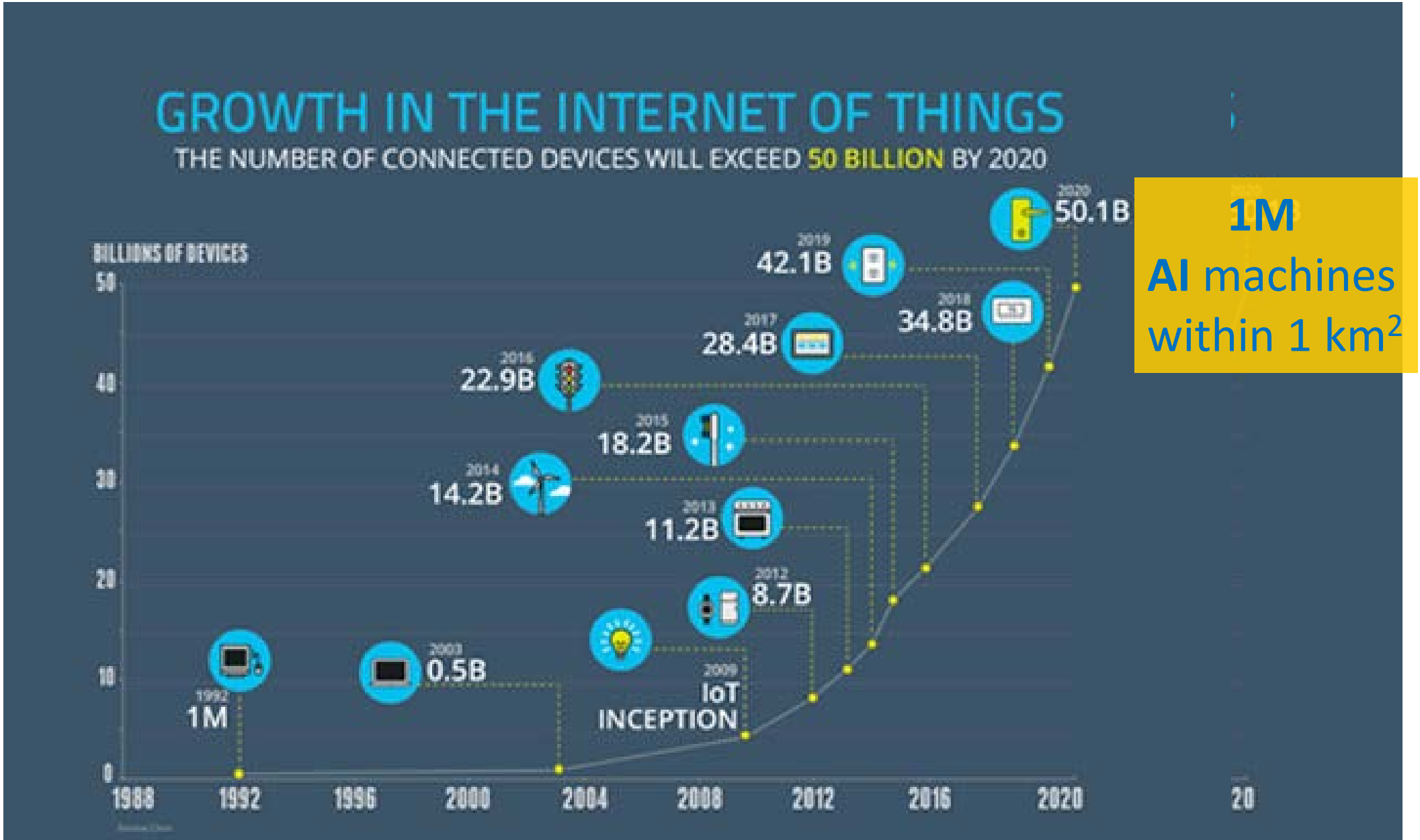
Super-Resolution Enhancement/Restoration


5X Compression
4X Model
3.6X FLOPS/Latency


Mc Graw Hill
OSBORNE
kelstein
Ellen Finkelstein
Mc Graw Hill
OSBORNE

X Learning for 5G Inter-machine Intelligence

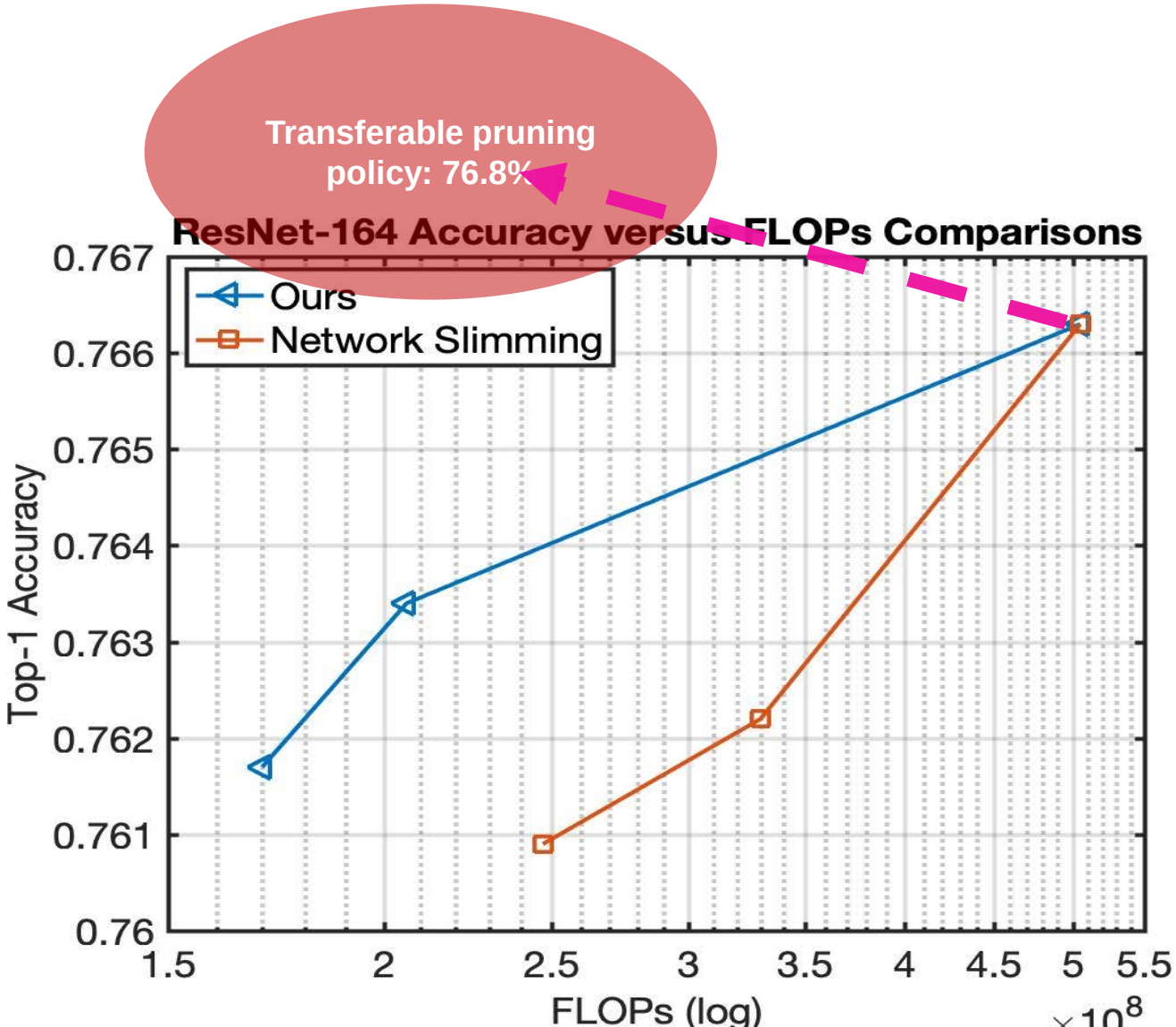
Transferable Learning



IMaX inter-machine learning (5G-AI)

IMaX-Learning: parameter transfer vs. structure transfer

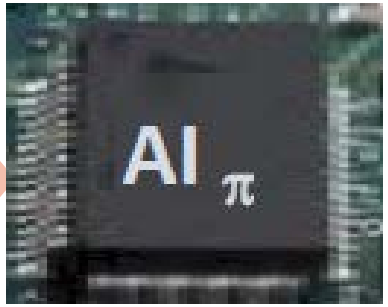
Structural Transfer Learning:
ResNet164 on CIFAR
(CIFAR10-to-CIFAR100)



Automation of X-Learning

Reduction/Trimming of Deep Learning Networks

Automatic Bigdata Deep Compiler
(ABDC)



Thank You!!