HKU Musketeers Foundation
Institute of Data Science
香港大學同心基金數據科學研究院

Neural Networks Through the Lens of Network Science: Structure and Performance

Jiayu Weng

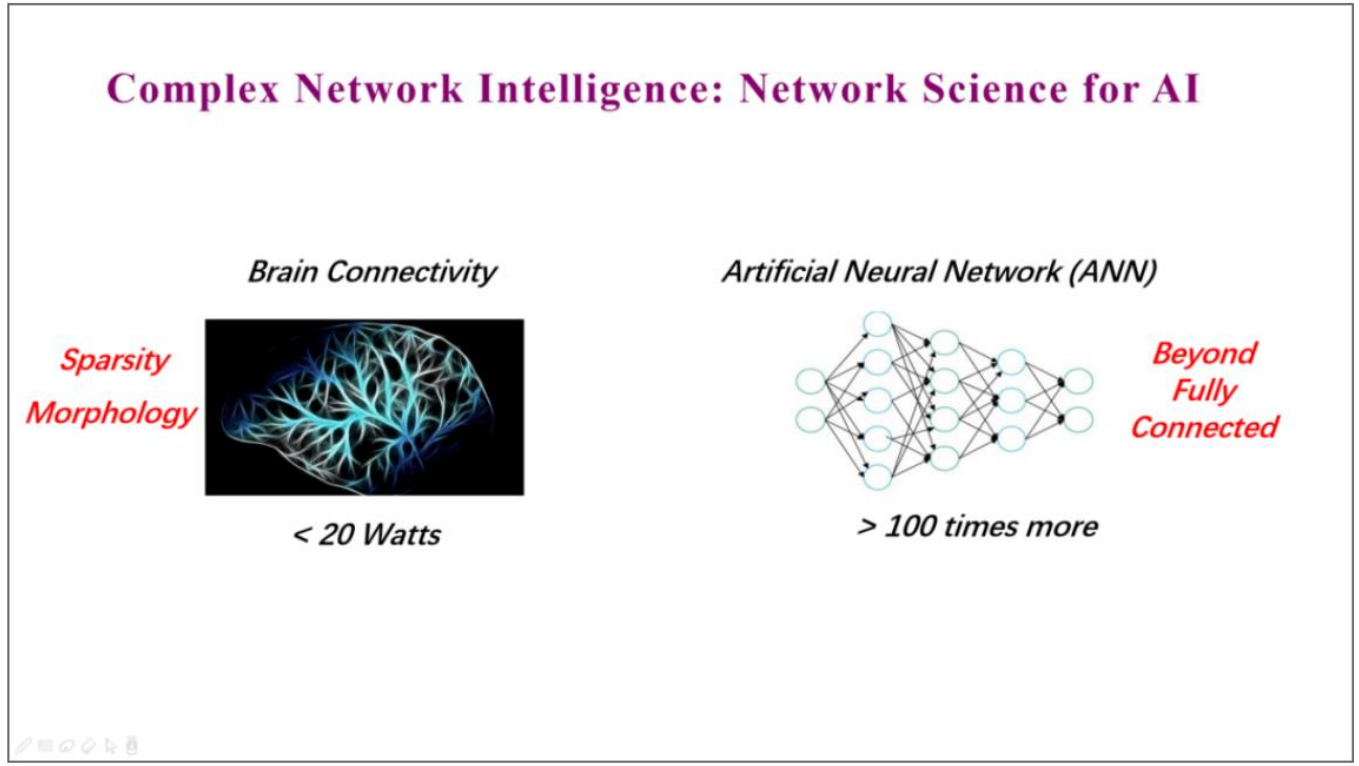
Intro

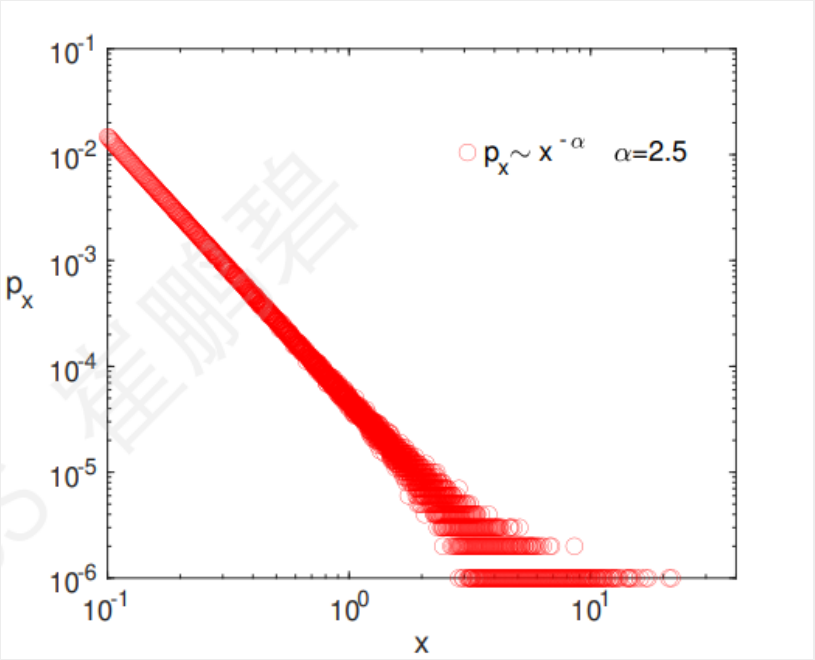
- Large model, the number of parameters of the SOTA models often exceeding 10^6
- high computational cost
  A large GPT model can use as much energy in a day as 30,000 U.S. homes

- How can we better understand those models?
  how to set hyperparameters, a way to predict the performance without much training, can we reduce the number of training parameters?

Intro

the network properties of biological neural networks (e.g. sparsity, scale-freeness)





PowerPoint courtesy of Prof. Carlo Vittorio Cannistraci

Sparse training

Mocanu, Decebal Constantin, et al. "Scalable training of artificial neural networks with adaptive sparse connectivity inspired by network science." Nature communications 9.1 (2018): 2383.

Measure the characteristics of ANNs

Kang, Chris, et al. "Structural network measures reveal the emergence of heavy-tailed degree distributions in lottery ticket multilayer perceptrons." Neural Networks (2025): 107308.

Metric for model selection

Jiang, Chunheng, et al. "Network properties determine neural network performance." *Nature Communications* 15.1 (2024)

·1

nature communications

Article | Open access | Published: 19 June 2018

Scalable training of artificial neural netwo adaptive sparse connectivity inspired by n science

综合性期刊TOP SCI升级版 综合性期刊1区 IF 14.7 Decebal Constantin Mocanu ✉, Elena Mocanu, Peter Stone, Phuong H. Nguyen, Madeleine Gibescu & Antonio Liotta

Nature Communications **9**, Article number: 2383 (2018) | Cite this article

A procedure to replace fully-connected layers with sparse layers

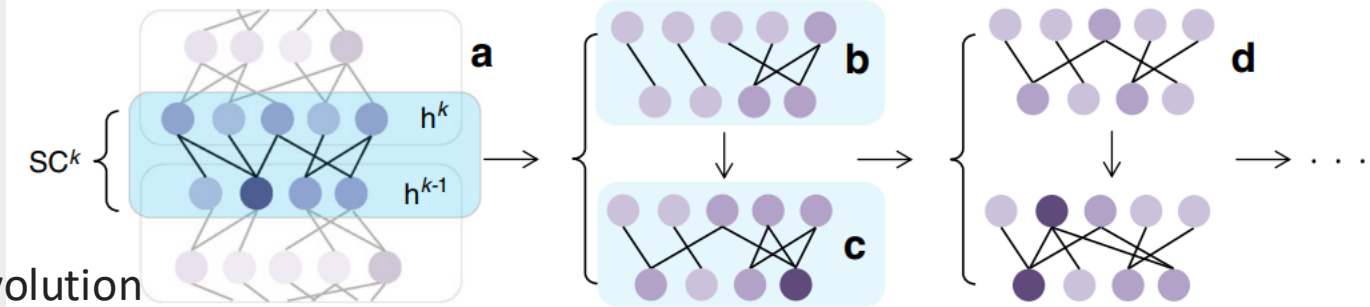
Sparse evolutionary training (SET) method

replace FC with a Sparse Connected (SC) layer having a Erdős-Rényi topology given by ε and Eq.1;

(Layer h_k has n_k neurons)

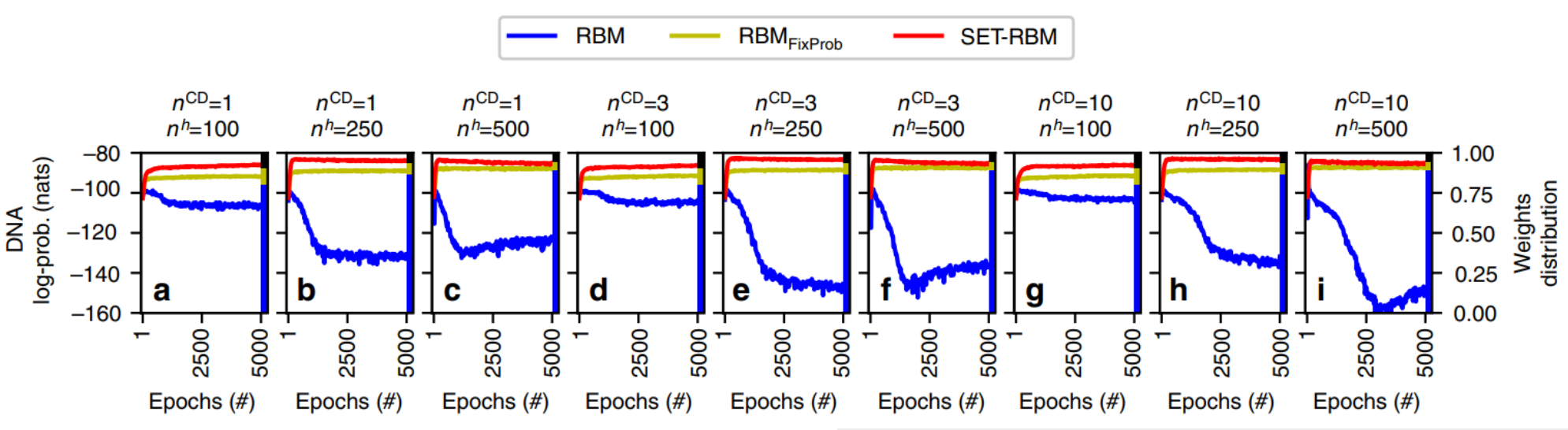
$$p\left(W_{ij}^{k}\right) = \frac{\varepsilon\left(n^{k} + n^{k-1}\right)}{n^{k} n^{k-1}}$$

remove a fraction of the smallest positive weights
and the largest negative weights
add randomly new connections

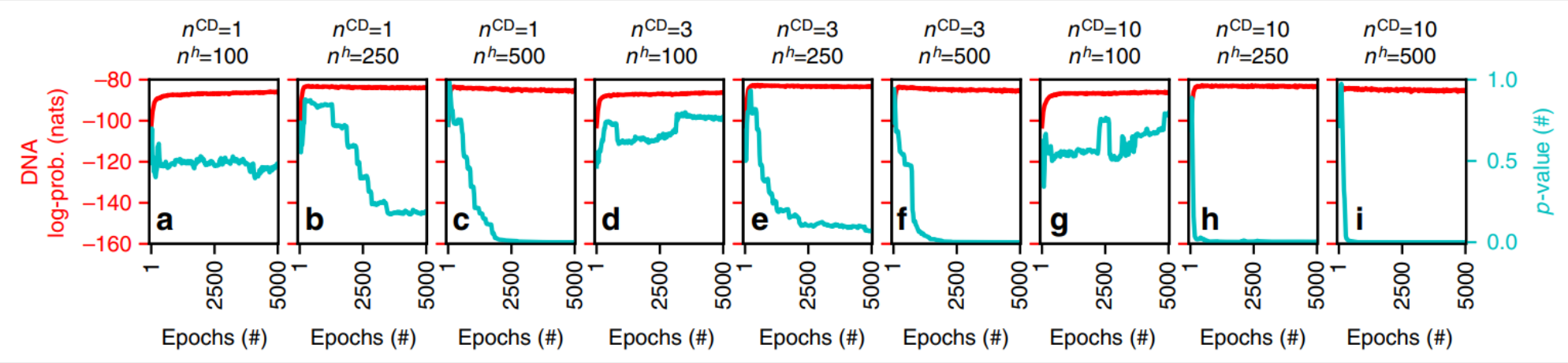


the selection and mutation phase of natural evolution

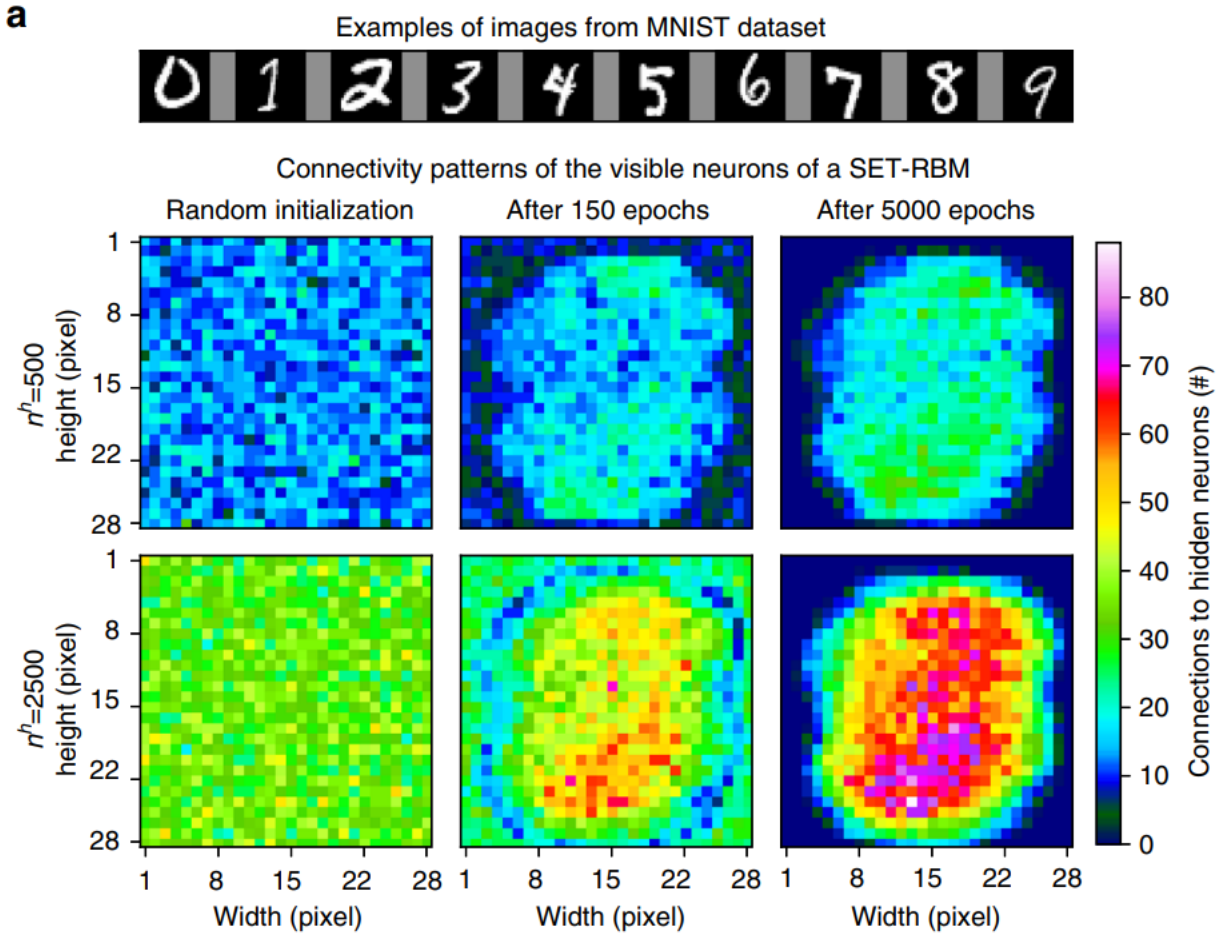
(RBM) Restricted Boltzmann Machine

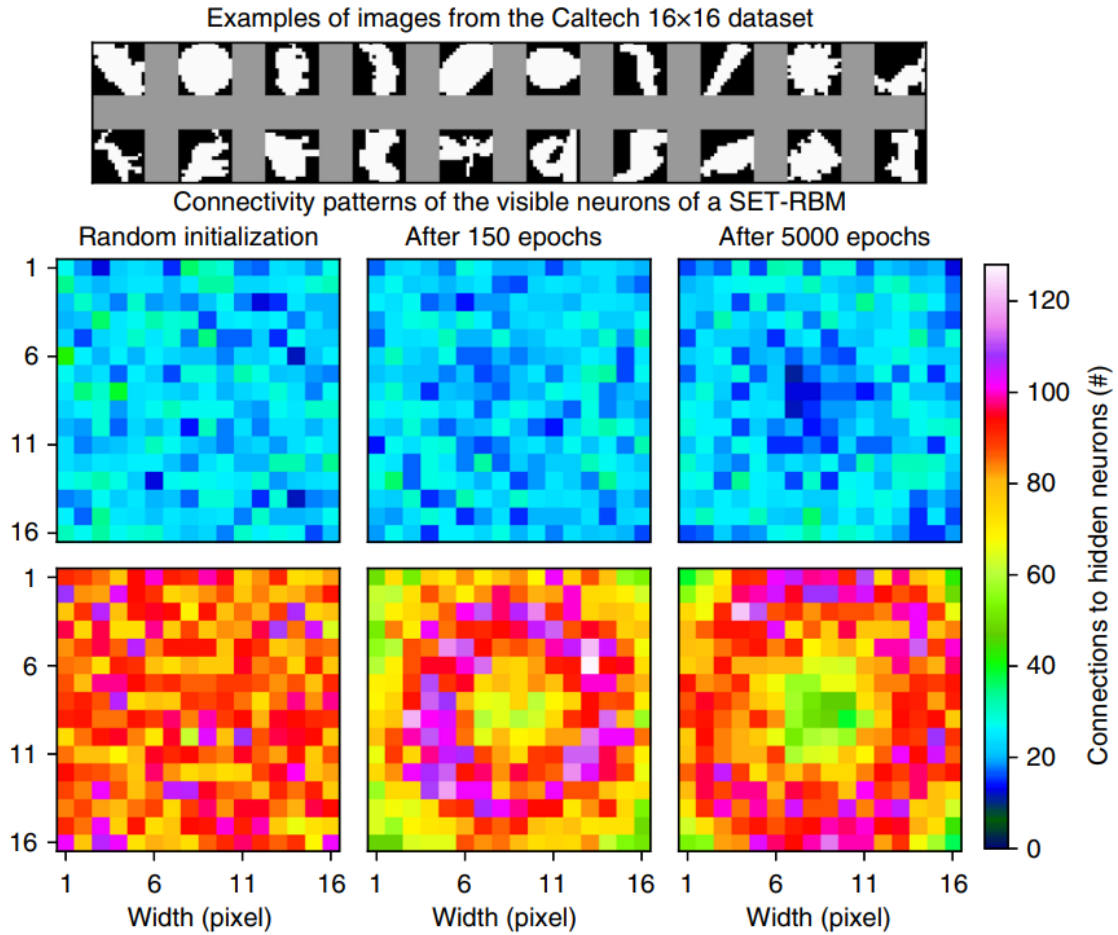


(Higher average log-probability means the model is more confident about the test data.)

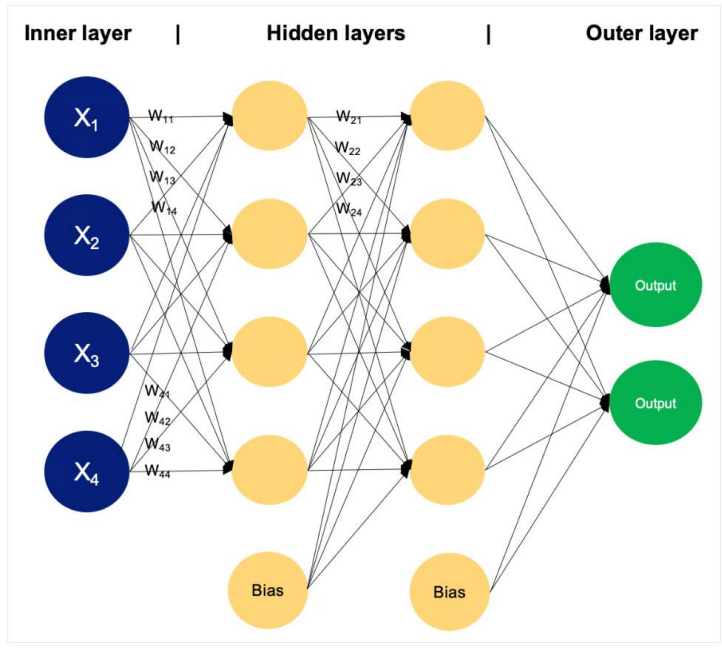


Connectivity patterns for the visible neurons

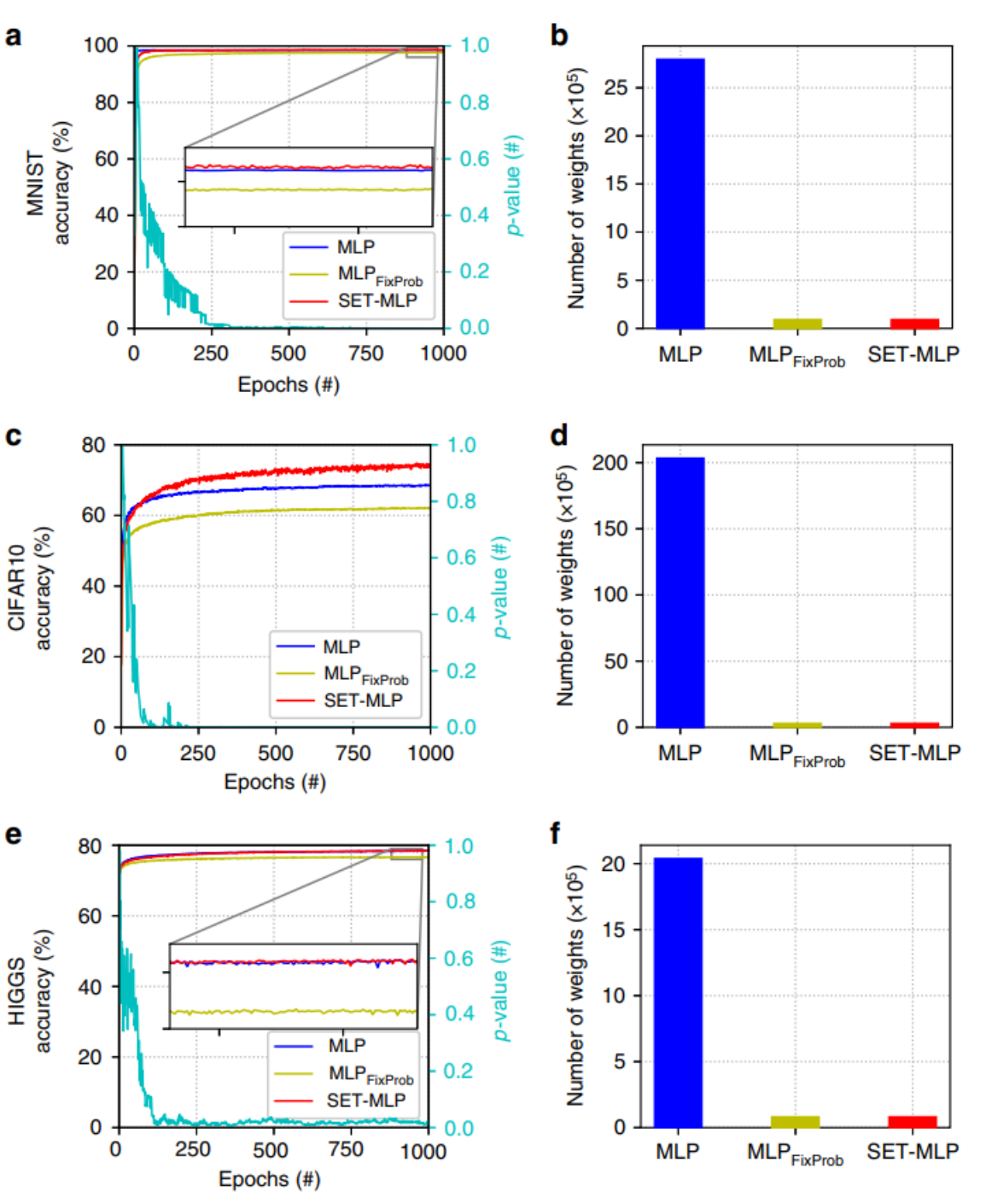




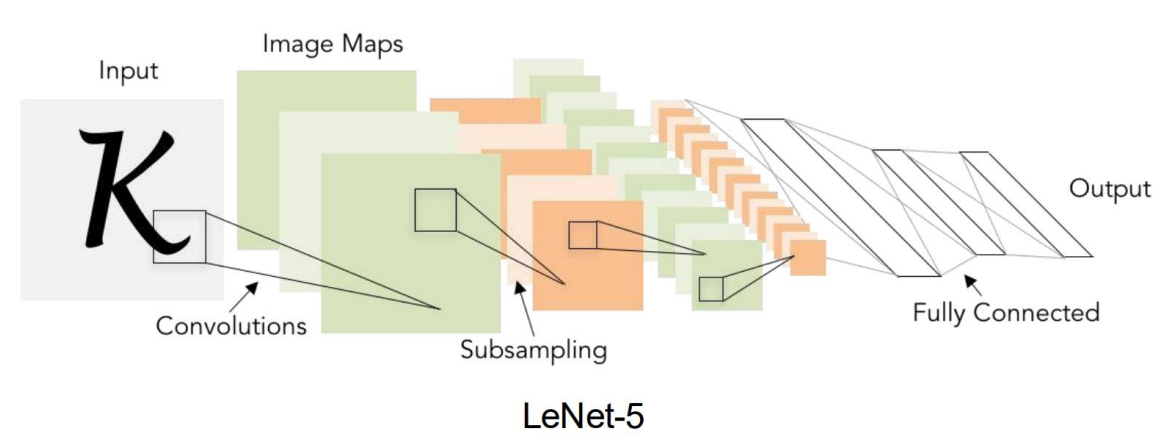
(MLP) Multi-layer Perceptrons

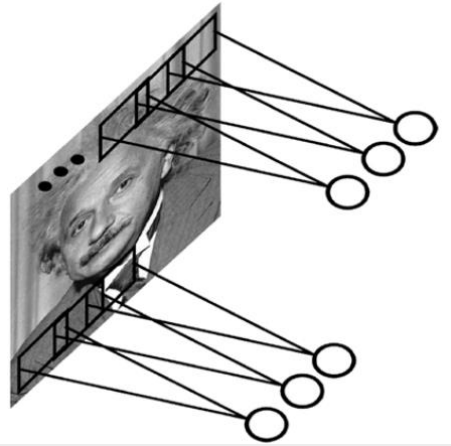


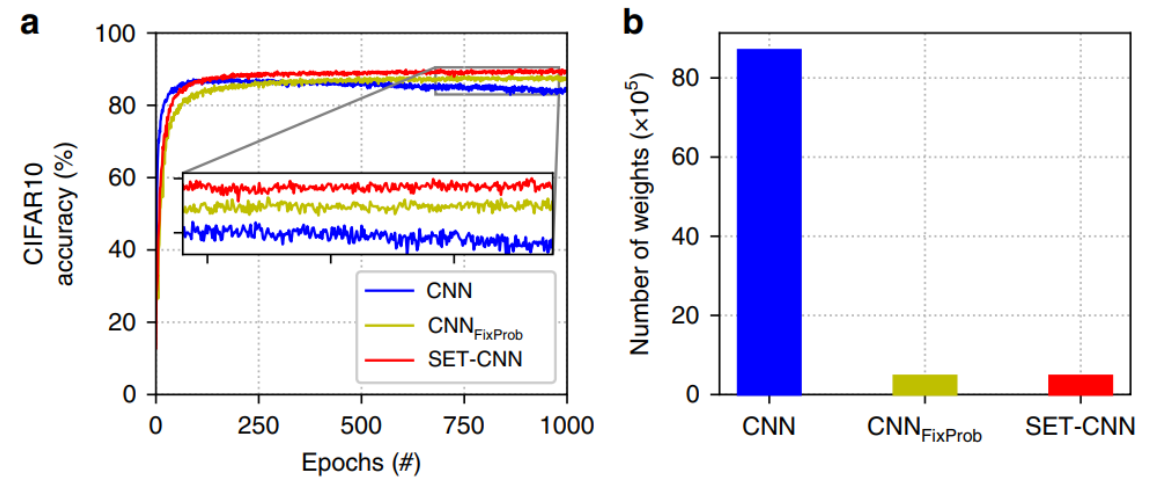
HIGGS dataset: SET-MLP: 78.47% accuracy with 90K params vs. 78.54% with 3× more



(CNN) Convolutional Neural Network







conv(32,(3,3))-dropout(0.3)-conv(32,(3,3))-pooling-conv(64,(3,3))-dropout(0.3)-conv(64,(3,3))-pooling-conv(128,(3,3))-dropout(0.3)-conv(128,(3,3))-pooling)

1 Sparse training

Mocanu, Decebal Constantin, et al. "Scalable training of artificial neural networks with adaptive sparse connectivity inspired by network science." Nature communications 9.1 (2018): 2383.

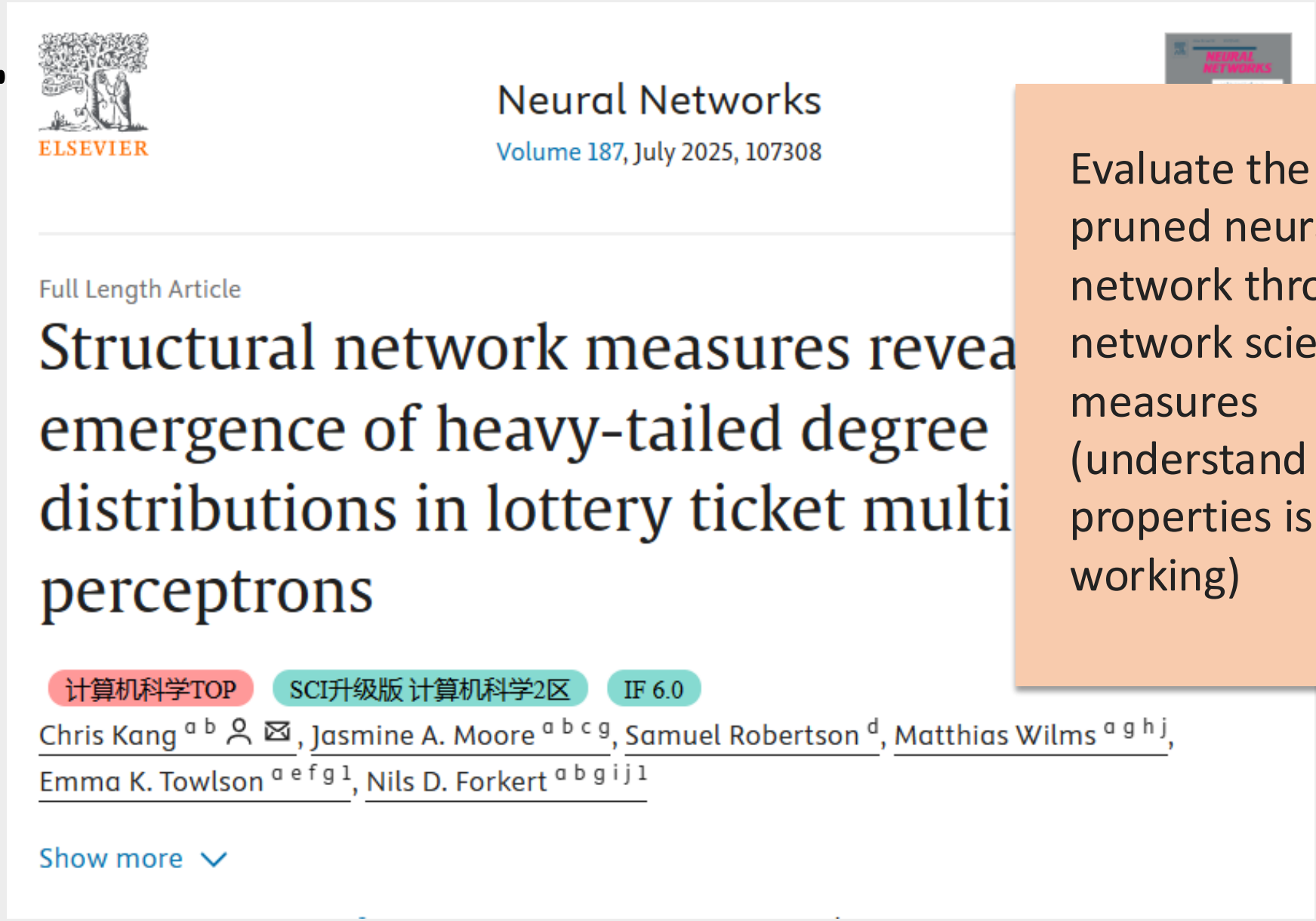
Full connections — they’re not how brains work

SET: Sparse Evolutionary Training algorithm

ER random graph → Scale-free topology

Quadratic reduction in parameters, with no decrease in accuracy

·2



ELSEVIER

Neural Networks

Volume 187, July 2025, 107308

Full Length Article

Structural network measures revea emergence of heavy-tailed degree distributions in lottery ticket multi perceptrons

计算机科学TOP | SCI升级版 计算机科学2区 | IF 6.0

Chris Kang [a b], Jasmine A. Moore [a b c g], Samuel Robertson [d], Matthias Wilms [a g h j], Emma K. Towlson [a e f g 1], Nils D. Forkert [a b g i j 1]

Show more

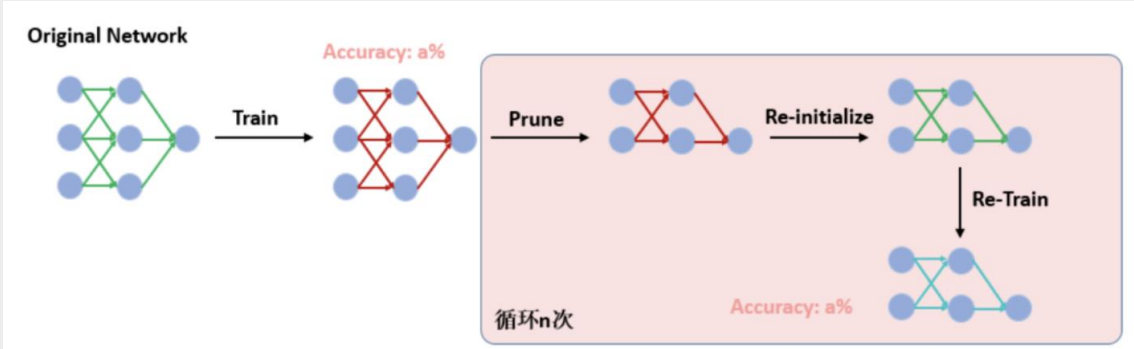
Evaluate the pruned neural network through network science measures (understand what properties is working)

Lottery ticket hypothesis

Frankle, Jonathan, and Michael Carbin. "The lottery ticket hypothesis: Finding sparse, trainable neural networks." *arXiv preprint arXiv:1803.03635* (2018). (ICLR 2019 Best Paper)

Neural network pruning
well-performing sparse subnetworks (winning tickets)

- Small subnetworks = same performance
- Found by pruning + resetting
- Good init weights are key
- Sparse nets = less compute



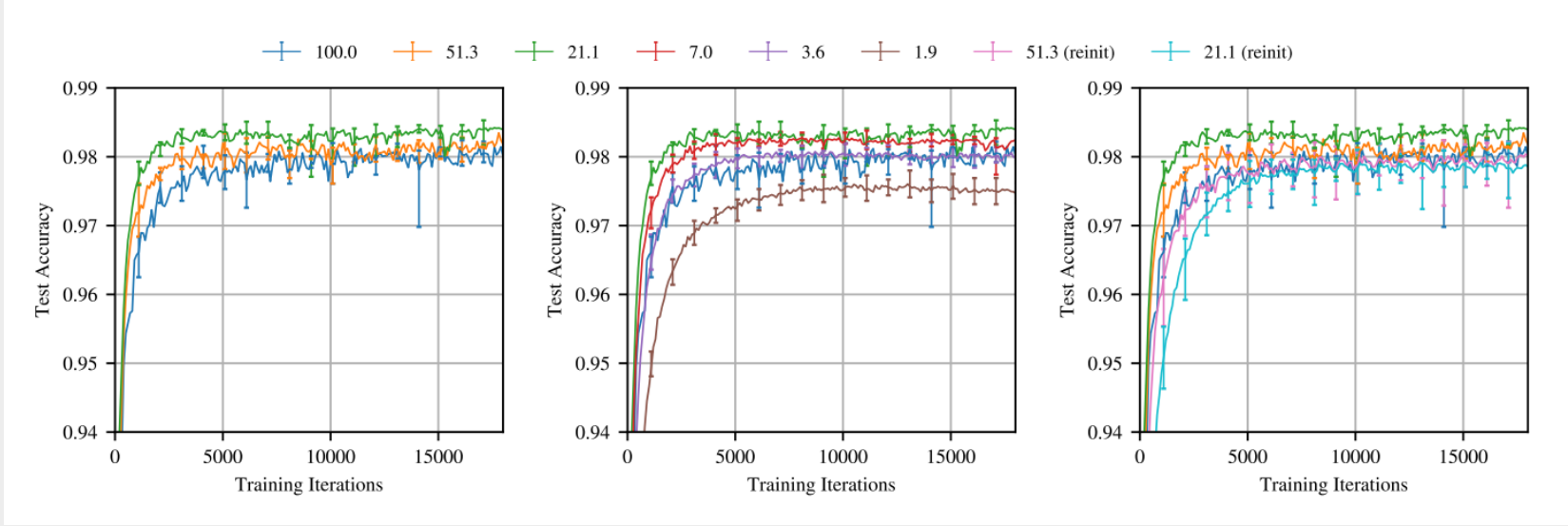
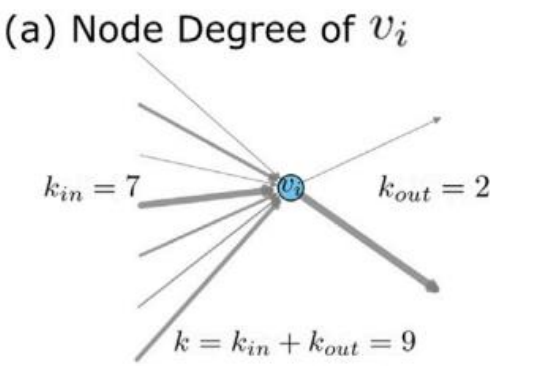
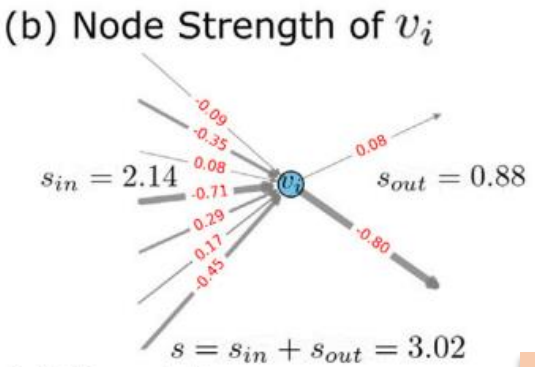
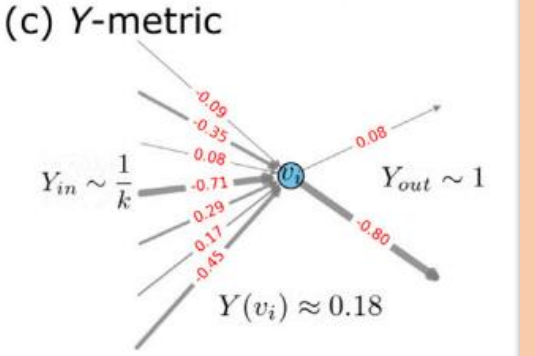
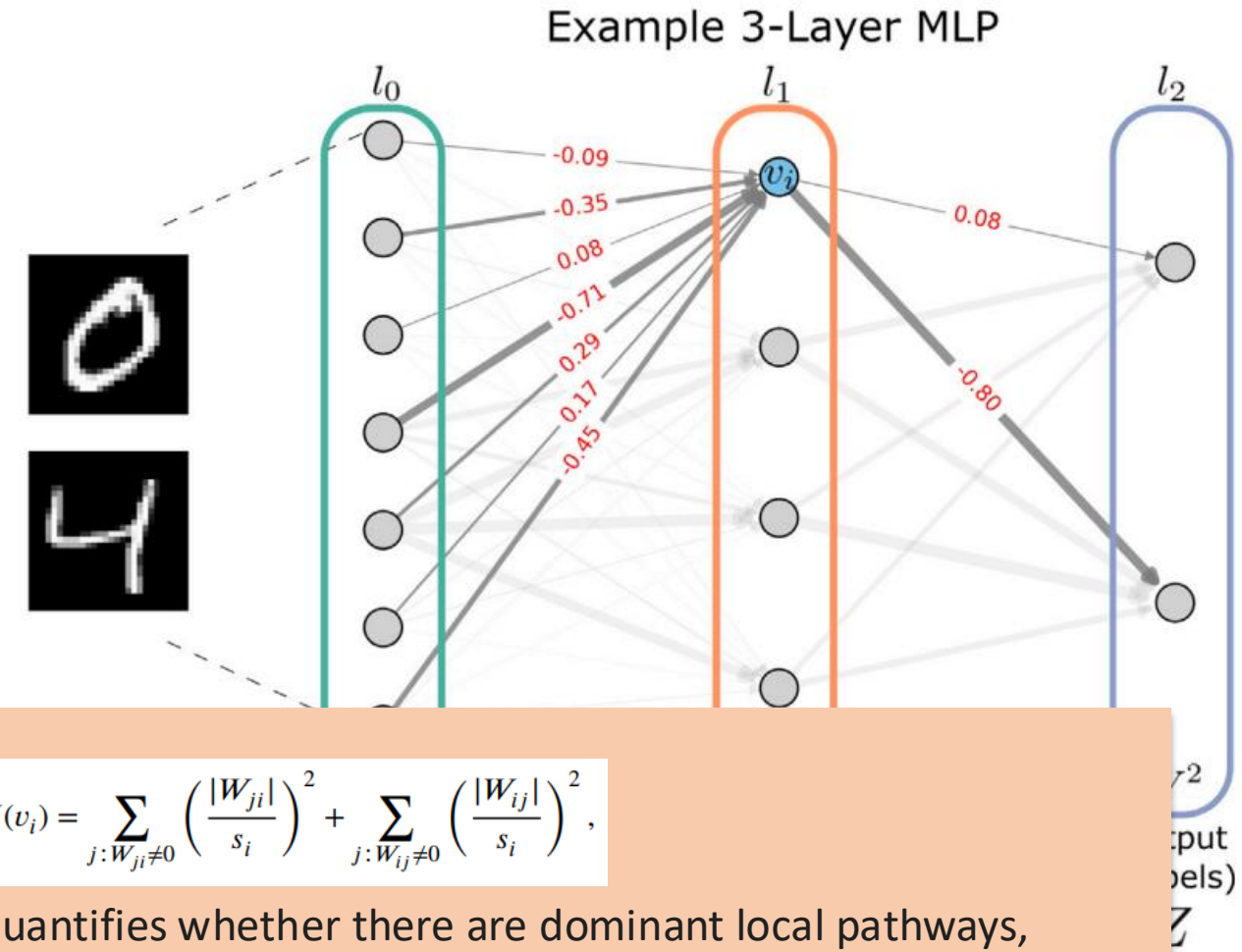


Illustration of MLP and network metrics





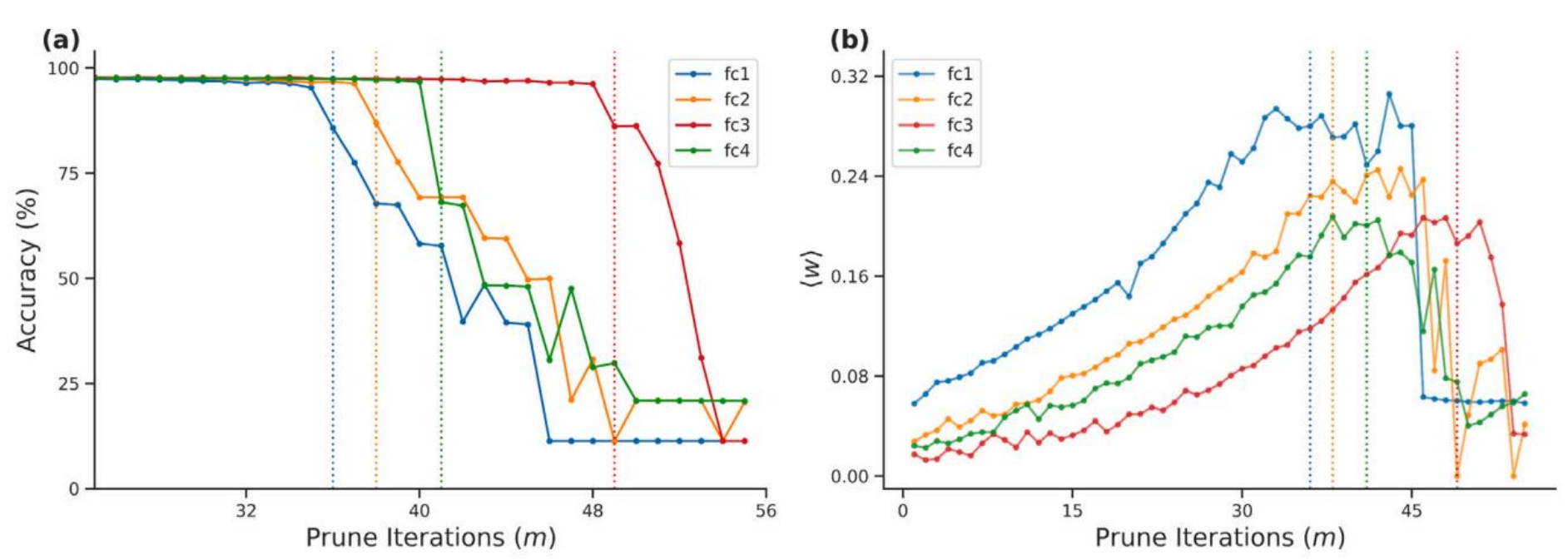




$$Y(v_i) = \sum_{j:W_{ji}\neq 0} \left(\frac{|W_{ji}|}{s_i}\right)^2 + \sum_{j:W_{ij}\neq 0} \left(\frac{|W_{ij}|}{s_i}\right)^2,$$

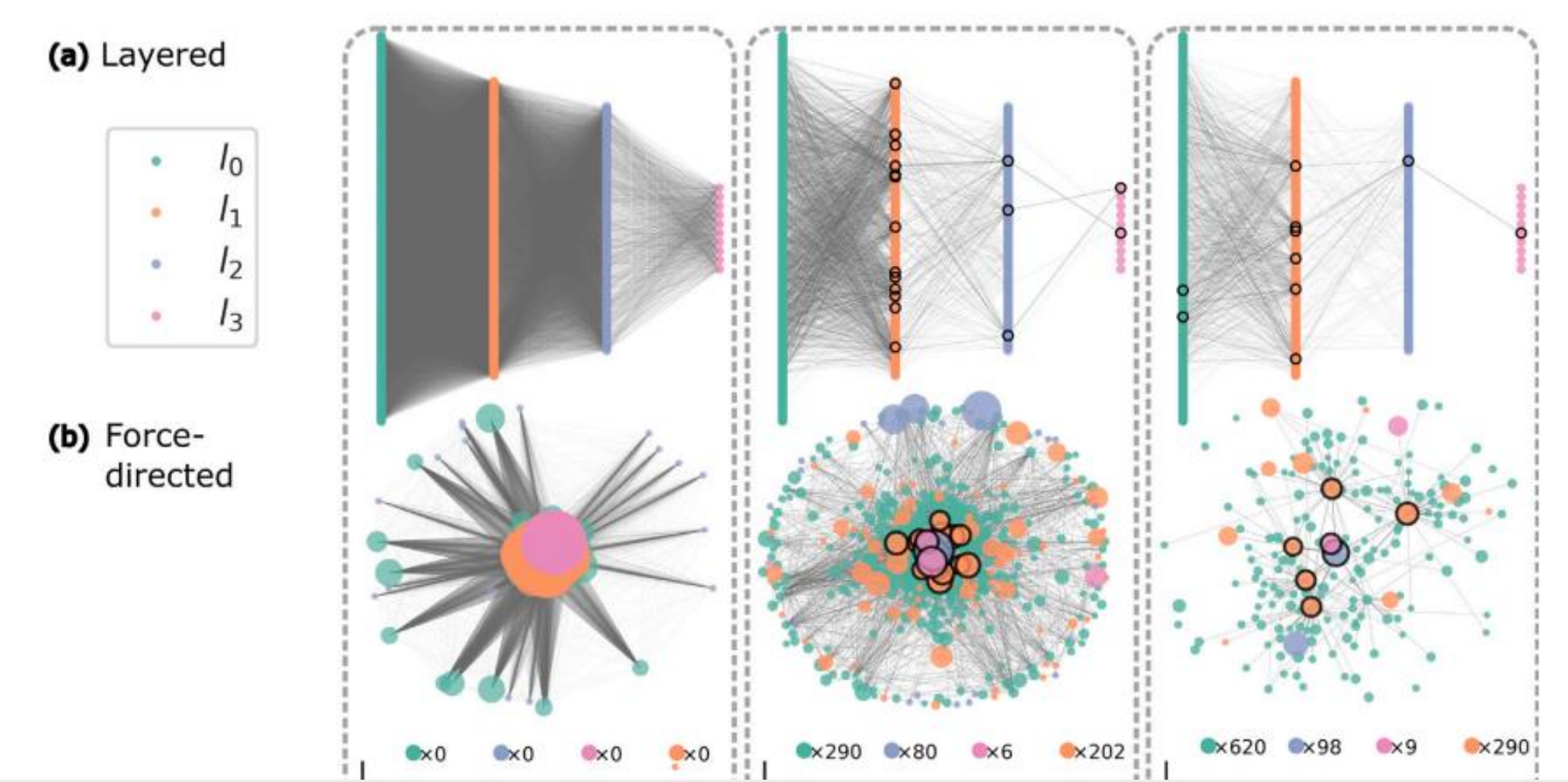
quantifies whether there are dominant local pathways, if pathways are more heterogeneous.

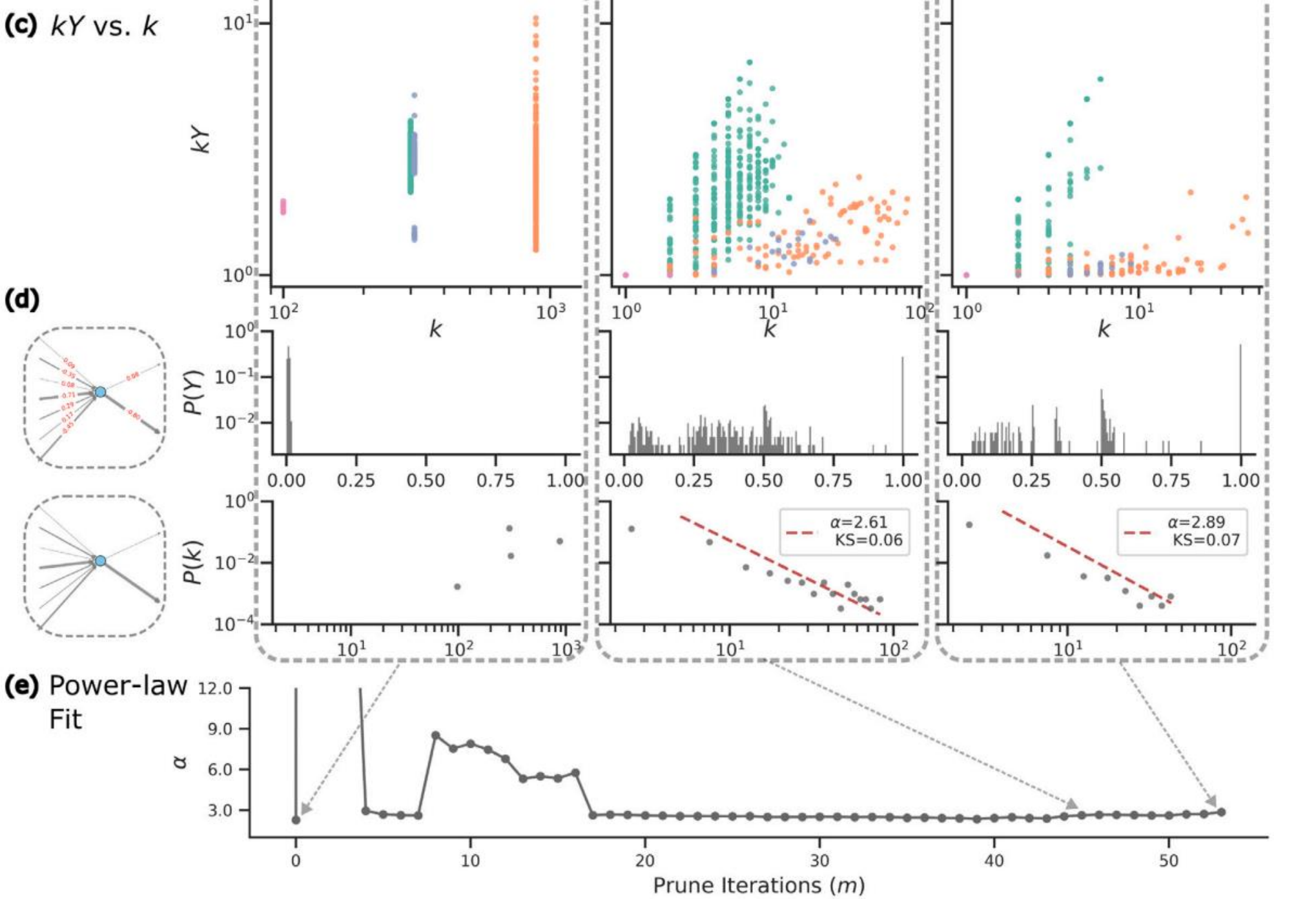
Performances of the LT MLPs trained on MNIST



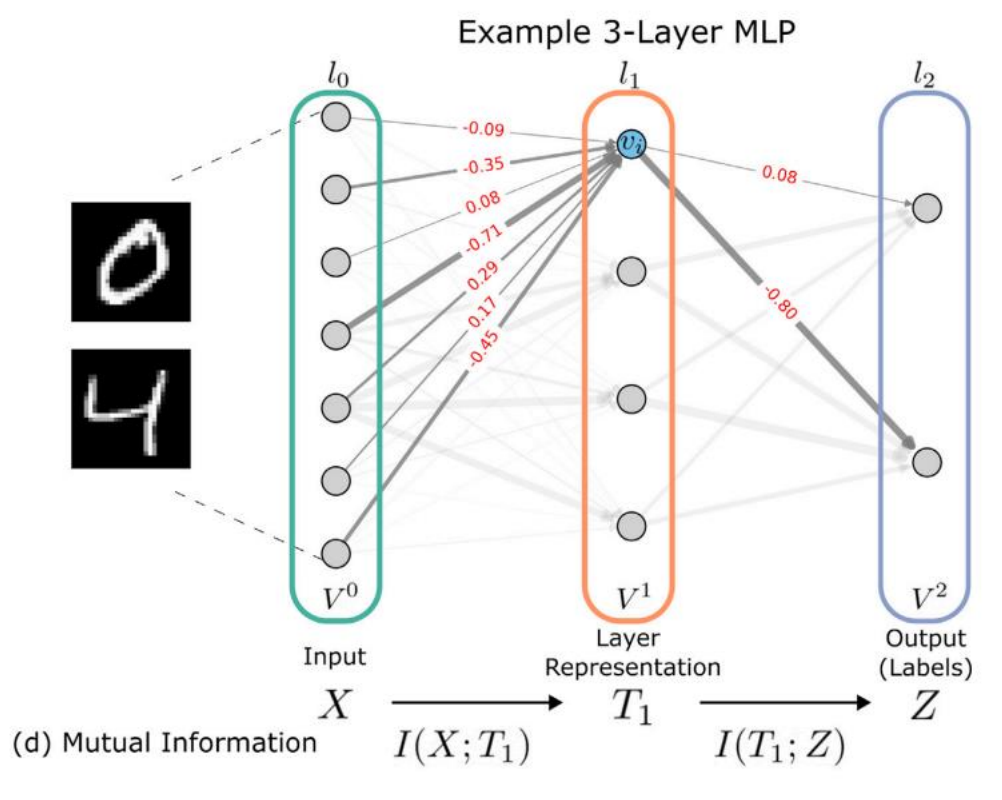
the near-maximum〈w〉values are attained at around the same iteration or prior to which the catastrophic LT MLP performances begin to decay.

Layered MLP representation

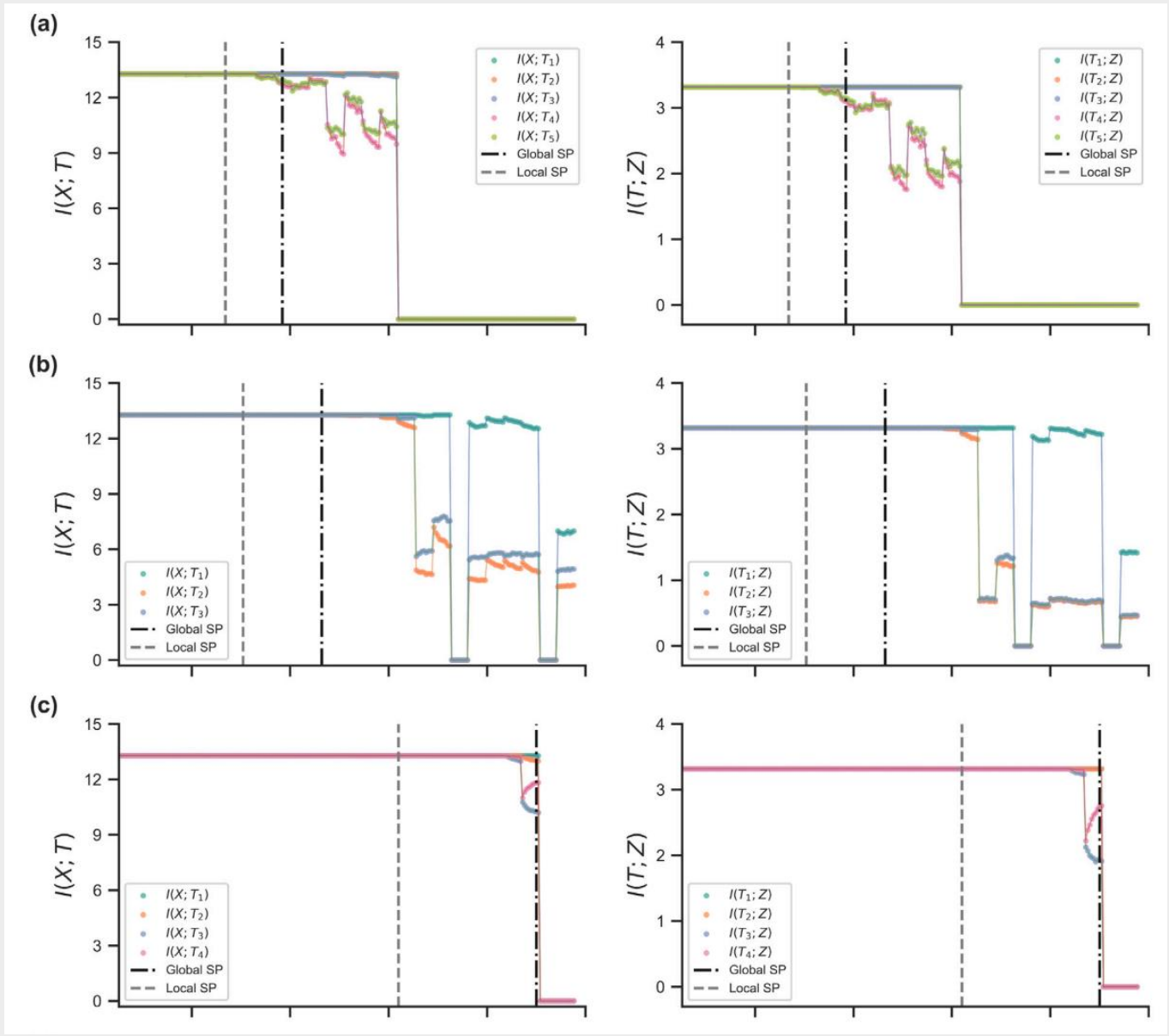



(c) kY vs. k
kY
k
(d)
P(Y)
P(k)
α=2.61
KS=0.06
α=2.89
KS=0.07
(e) Power-law Fit
α
Prune Iterations (m)

Mutual Information



$$I(X;Z) = \sum_{x \in X, z \in Z} p(x,z) \log \frac{p(x,z)}{p_X(x) p_Z(z)}.$$



2 Measure the characteristics of ANNs

Kang, Chris, et al. "Structural network measures reveal the emergence of heavy-tailed degree distributions in lottery ticket multilayer perceptrons." Neural Networks (2025): 107308.

pruned subnetworks in over-parameterized MLPs evolve to form heterogeneous network structures that fit a heavy-tailed distribution

May guide network pruning
the potential to investigate and improve the fidelity between ANNs and the human brain

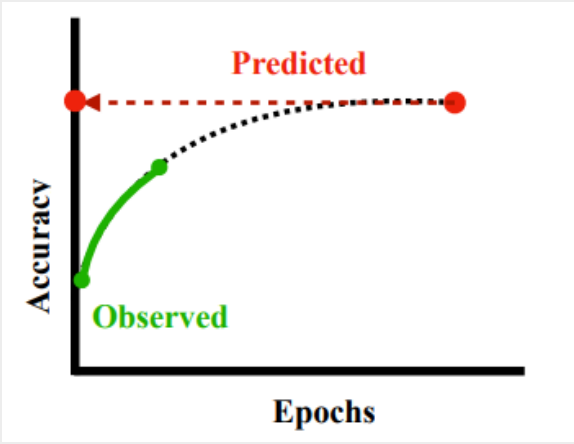
·3

A model selection problem a metric (from network resilience study) to know how well the model works through a few training epochs

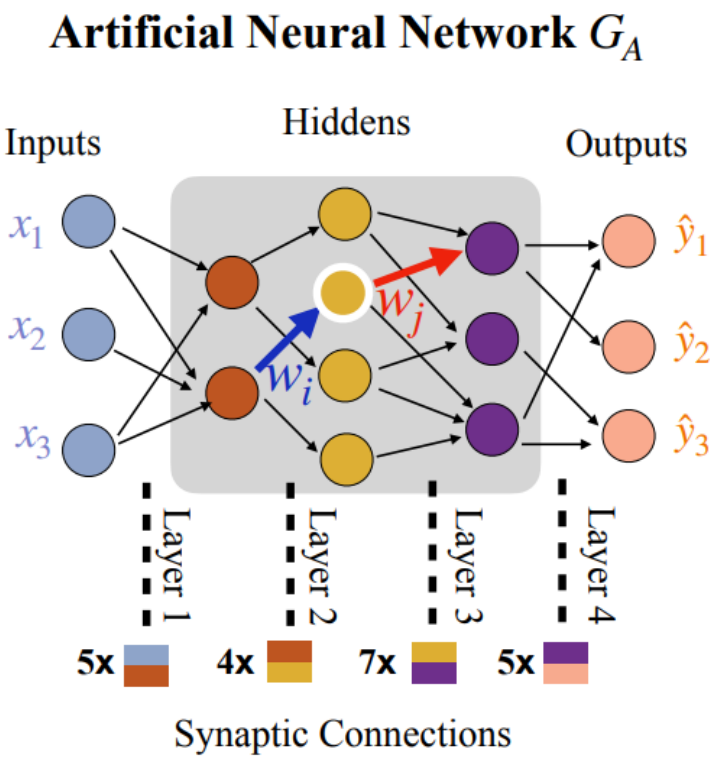
Problem: Model Selection

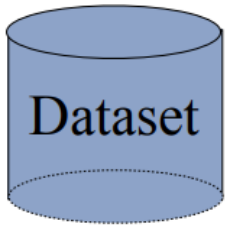
Model	Size (MB)	Top-1 Accuracy	Top-5 Accuracy	Parameters	Depth	Time (ms) per inference step (CPU)	Time (ms) per inference step (GPU)
Xception	88	79.0%	94.5%	22.9M	81	109.4	8.1
VGG16	528	71.3%	90.1%	138.4M	16	69.5	4.2
VGG19	549	71.3%	90.0%	143.7M	19	84.8	4.4
ResNet50	98	74.9%	92.1%	25.6M	107	58.2	4.6
ResNet50V2	98	76.0%	93.0%	25.6M	103	45.6	4.4
ResNet101	171	76.4%	92.8%	44.7M	209	89.6	5.2
ResNet101V2	171	77.2%	93.8%	44.7M	205	72.7	5.4
ResNet152	232	76.6%	93.1%	60.4M	311	127.4	6.5
ResNet152V2	232	78.0%	94.2%	60.4M	307	107.5	6.6
InceptionV3	92	77.9%	93.7%	23.9M	189	42.2	6.9
InceptionResNetV2	215	80.3%	95.3%	55.9M	449	130.2	10.0
MobileNet	16	70.4%	89.5%	4.3M	55	22.6	3.4
MobileNetV2	14	71.3%	90.1%	3.5M	105	25.9	3.8
DenseNet121	33	75.0%	92.3%	8.1M	242	77.1	5.4
DenseNet169	57	76.2%	93.2%	14.3M	338	96.4	6.3
DenseNet201	80	77.3%	93.6%	20.2M	402	127.2	6.7
NASNetMobile	23	74.4%	91.9%	5.3M	389	27.0	6.7

- **Given**:
  - Dataset $D = D_{\text{train}} \cup D_{\text{test}}$
  - Model set $\{F_1, F_2, ..., F_m\}$
- **Goal**:
  - Use early training signals to pick the best model F^*
  - F^* is expected to perform best on D_{test} after full training (e.g. 500 epochs)



NN Training is a Dynamical System





$$D = \{(X_i, Y_i)\}_{i=1}^{N},$$
$$X_i = (x_{i1}, x_{i2}, x_{i3}),$$
$$Y_i = (y_{i1}, y_{i2}, y_{i3})$$

- **Forward Propagation**

  $G_A : x \mapsto \hat{y} \mapsto C(y, \hat{y}; w)$

- **Back-propagation (stochastic gradient descent)**

  $\text{SGD} : w \leftarrow w - \alpha \nabla_w C$

Co-evolution of $\{w_1, w_2, \ldots\}$ ***during training?***

↓

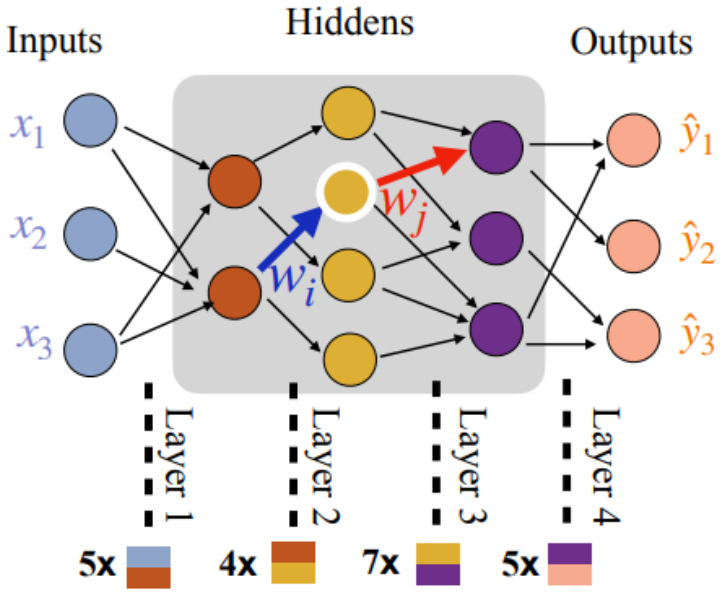
A complex networked system over the parameters $\{w_1, w_2, \ldots\}$ **of** G_A

7

Our Approach: Build a Line Graph G_B from G_A

Nepusz &Vicsek, Nature Physics, 2012

Artificial Neural Network G_A

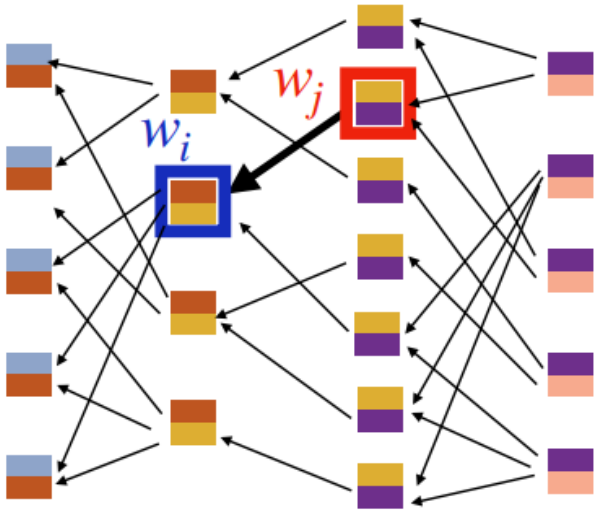


Synaptic Connections

$\phi : G_A \rightarrow G_B$

Network Mapping

Line Graph G_B



Edge Dynamics $\mathscr{B}$

$$\dot{w}_i = f(w_i) + \sum_j P_{ij} g(w_i, w_j)$$

Node dynamics

$$(\mathcal{A})\ d\mathbf{z}/dt \approx h_{\mathcal{A}}(\mathbf{z},t;\mathbf{w}) = -\nabla_{\mathbf{z}} C(\mathbf{z}(t)), \qquad (3)$$

$$(\mathcal{B})\ d\mathbf{w}/dt \approx h_{\mathcal{B}}(\mathbf{w},t;\mathbf{z}) = -\nabla_{\mathbf{w}} C(\mathbf{w}(t)), \qquad (4)$$

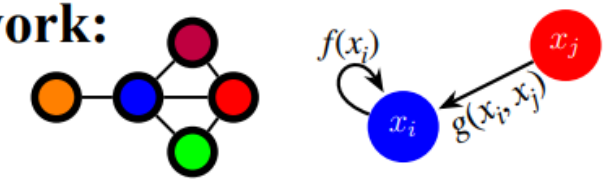
Edge dynamics

8

How to identify the edge dynamics $\mathscr{B}$, especially the matrix P?

Mean Field Approximation and Network Resilience

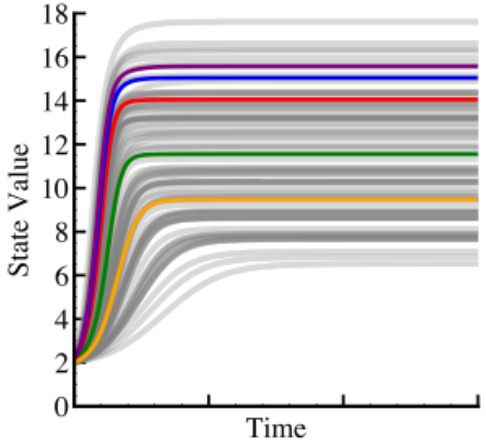
Network:



Governing dynamics:

$$\dot{x}_i = f(x_i) + \sum_j P_{ij} g(x_i, x_j)$$

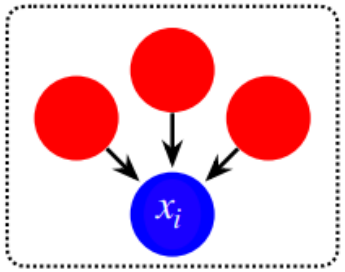
Forward Simulation



Solving $f(x_i) + \sum_j P_{ij} g(x_i, x_j) = 0$

Scalability

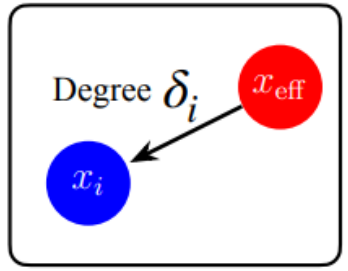
$N \mapsto$ millions, billions



$$\dot{x}_i = f(x_i) + \sum_j P_{ij} g(x_i, x_j)$$

Mean Filed Approximation

$$L_P(z) = \frac{1^T P z}{1^T P 1}$$



$$\dot{x}_i = f(x_i) + \delta_i g(x_i, x_{\text{eff}})$$

Dimension reduction: $N \to 1$

$$\dot{x}_{\text{eff}} = f(x_{\text{eff}}) + \beta_{\text{eff}}\, g(x_{\text{eff}}, x_{\text{eff}})$$

$$\beta_{\text{eff}} = L_P(\delta) = \langle \delta^2 \rangle / \langle \delta \rangle$$

resilience metric of a network

Gao et al., Nature, 2016

Jiang et al., AAAI, 2020

Jiang et al., IJCAI, 2020

Sanhedrai et al., arXiv, 2020

5

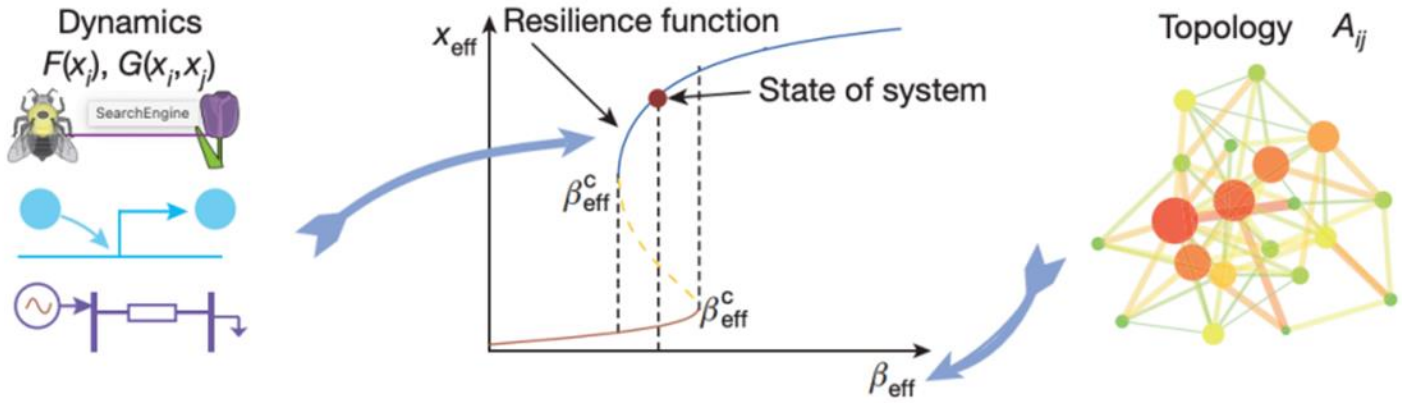
Mean Field Approximation and Network Resilience

Networks	Dynamics
Ecological (e.g. plant-pollinator networks)	$\dot{x}_i = B + x_i(1 - \frac{x_i}{K})(\frac{x_i}{C} - 1) + \sum_j A_{ij} \frac{x_i x_j}{D + Ex_i + Hx_j}$
Regulatory (e.g. genes networks)	$\dot{x}_i = -Bx_i^f + \sum_j A_{ij} R \frac{x_i^h}{x_j^h + 1}$
Epidemic (e.g. social networks)	$\dot{x}_i = -Bx_i + \sum_j A_{ij} R(1 - x_i)x_j$
Neuronal (e.g. brain networks)	$\dot{x}_i = -x_i + \omega \sum_j A_{ij} \frac{1}{1 + e^{\mu - \delta x_j}}$

Barabasi, Science, 1999

Gao et al., Nature, 2016

Sanhedrai et al., arXiv, 2020



What are the β_{eff}s of deep neural networks ?

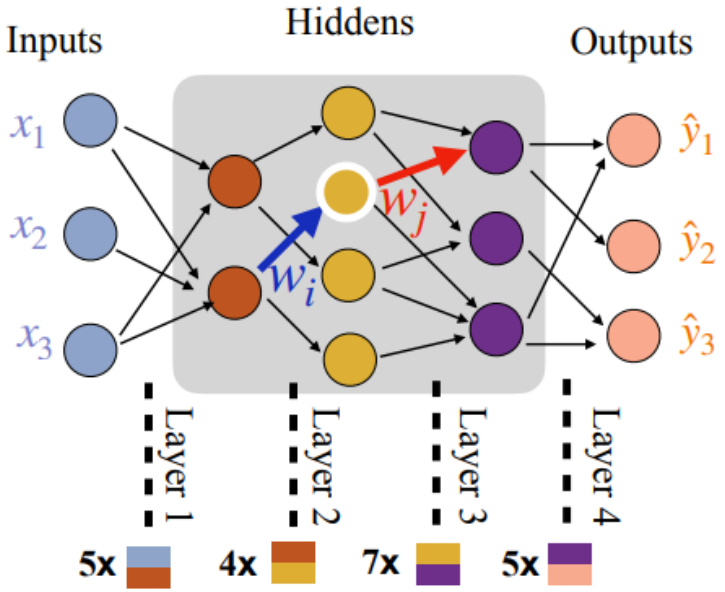
Gao et al. (2016). Universal resilience patterns in complex networks. *Nature*.

PowerPoint courtesy of the authors

Our Approach: Build a Line Graph G_B from G_A

Nepusz &Vicsek, Nature Physics, 2012

Artificial Neural Network G_A

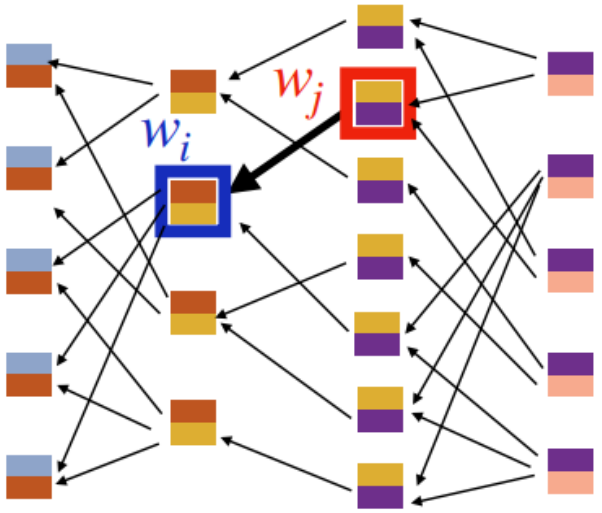


Synaptic Connections

$\phi : G_A \rightarrow G_B$

Network Mapping

Line Graph G_B



Edge Dynamics $\mathscr{B}$

$$\dot{w}_i = f(w_i) + \sum_j P_{ij} g(w_i, w_j)$$

Node dynamics

$$(\mathcal{A})\ d\mathbf{z}/dt \approx h_{\mathcal{A}}(\mathbf{z},t;\mathbf{w}) = -\nabla_{\mathbf{z}} C(\mathbf{z}(t)), \quad (3)$$

$$(\mathcal{B})\ d\mathbf{w}/dt \approx h_{\mathcal{B}}(\mathbf{w},t;\mathbf{z}) = -\nabla_{\mathbf{w}} C(\mathbf{w}(t)), \quad (4)$$

Edge dynamics

8

How to identify the edge dynamics $\mathscr{B}$, especially the matrix P?

Our Approach: Identify Edge Dynamics $\mathscr{B}$

Notations:

$$\delta^{(\ell)} = [\partial C/\partial z_1^{(\ell)}, \cdots, \partial C/\partial z_{n_\ell}^{(\ell)}]^T$$

$$\sigma'_\ell = [\sigma'_\ell(a_1^{(\ell)}), \cdots, \sigma'_\ell(a_{n_\ell}^{(\ell)})]^T$$

SGD:

$$\nabla_{W^{(\ell)}} = \Lambda(\sigma'_\ell) W^{(\ell+1)T} \Lambda(\sigma'_{\ell+1}) \delta^{(\ell+1)} z^{(\ell-1)T} = -F(W^{(\ell+1)})$$

↓ Taylor expansion

Edge Dynamics $\mathscr{B}$:

$$\dot{w}_i = f(w_i) + \sum_j P_{ij} g(w_i, w_j), \qquad (8)$$

$$P^{(l,l+1)} = \partial^2 C/\partial W^{(\ell)} \partial W^{(\ell+1)},$$

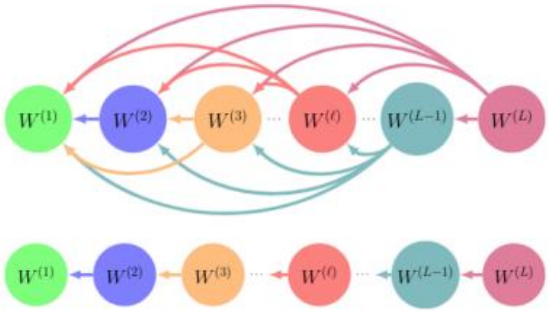
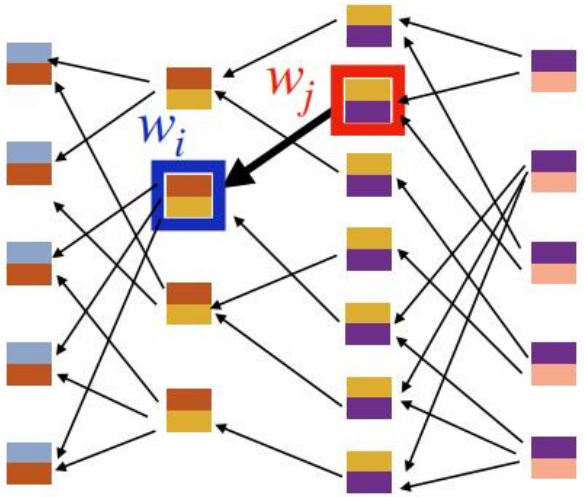


Figure 2: Exact (**top**) v.s. approximated (**bottom**) line graph to interpret the dependency of trainable weights $\boldsymbol{w}$ over synaptic connections.

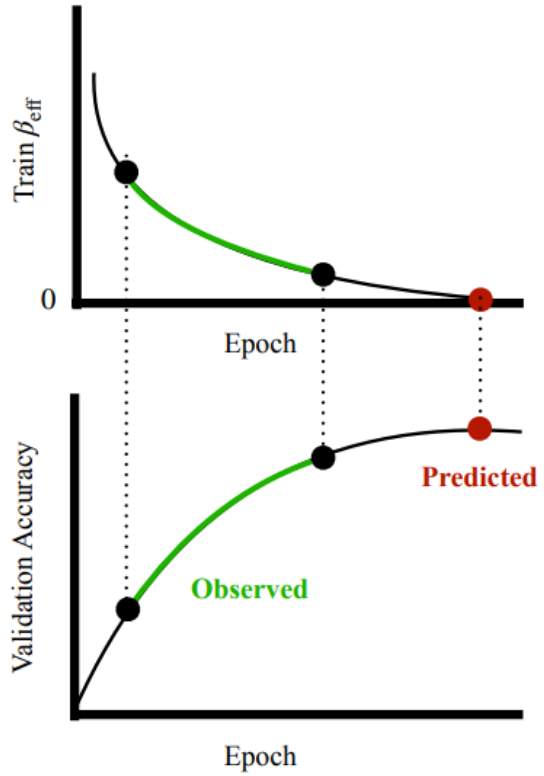


Line Graph G_B

Networked System (G_B, $\mathscr{B}$) ⟶ can β_{eff} reveal the performance of G_A?

Our Approach: Calculate β_{eff}

Theorem 1. Let ReLU be the activation function of G_A. When G_A converges, then $\beta_{\text{eff}} = 0$.

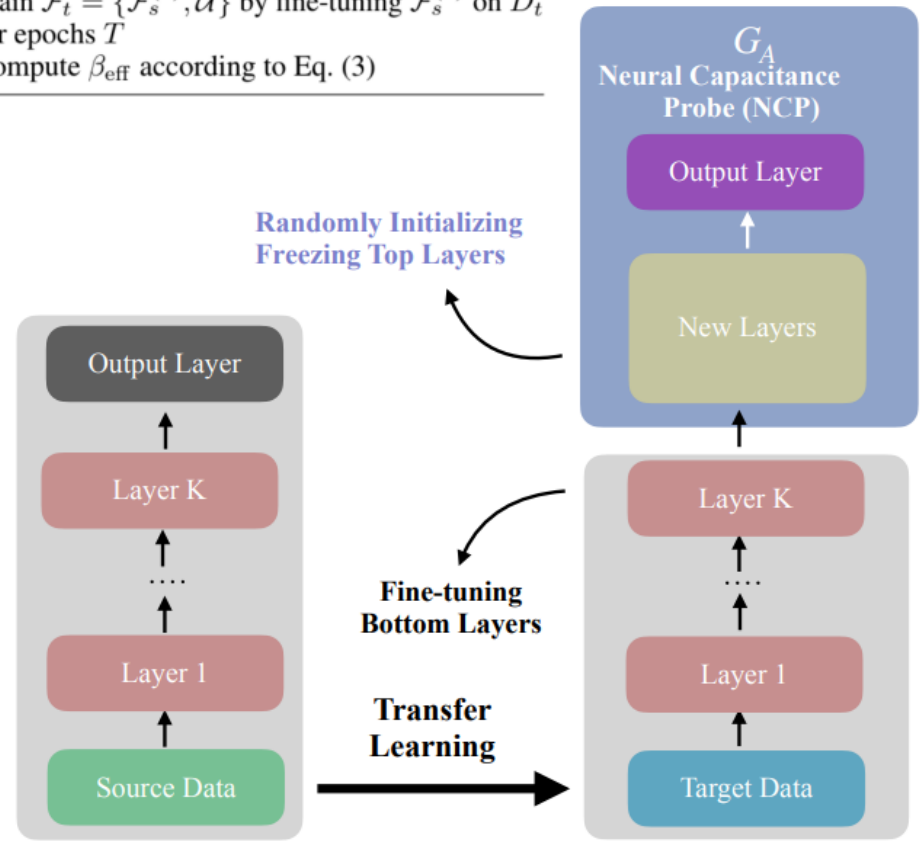


Learning Curve Prediction

Algorithm 1 Implement NCP and Compute β_{eff}

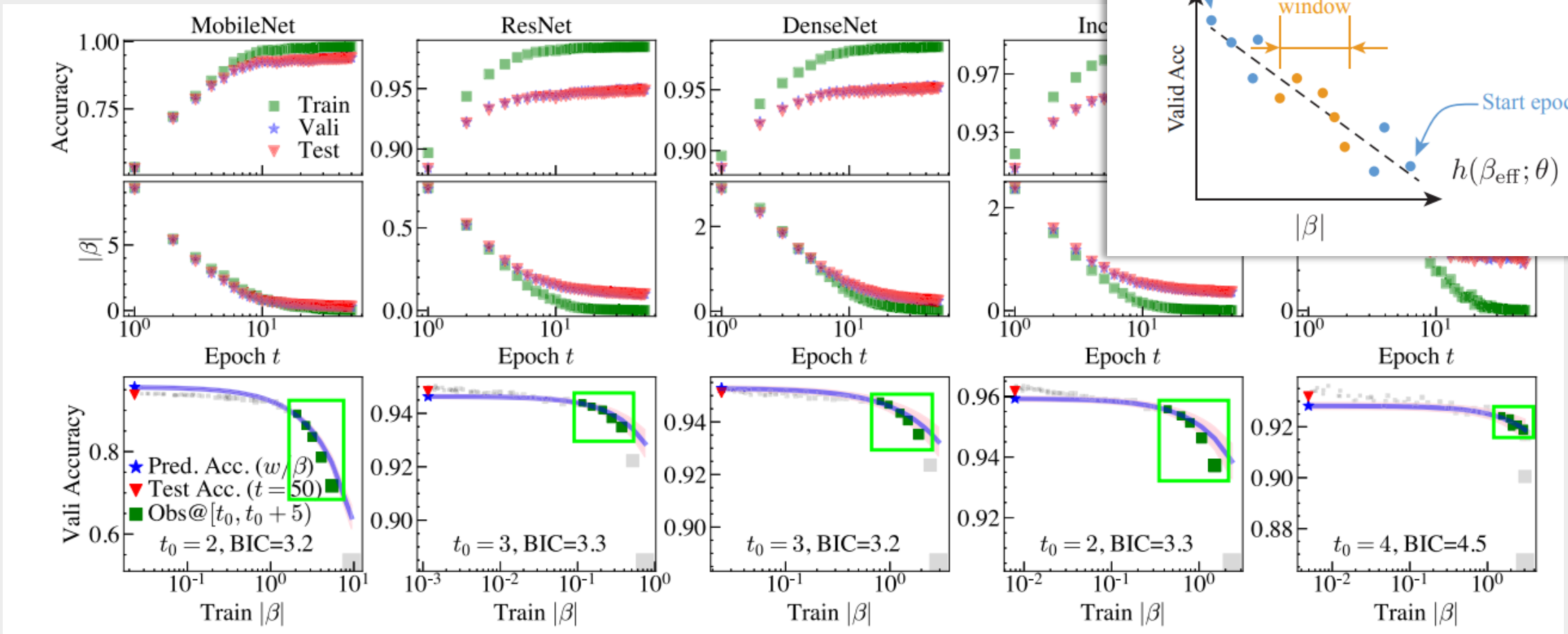
Input: A pre-trained model $\mathcal{F}_s = \{\mathcal{F}_s^{(1)}, \mathcal{F}_s^{(2)}\}$ with bottom layers $\mathcal{F}_s^{(1)}$ and output layer $\mathcal{F}_s^{(2)}$, a target dataset D_t, the maximum number of epochs T

1: Remove $\mathcal{F}_s^{(2)}$ from $\mathcal{F}_s$ and add on top of $\mathcal{F}_s^{(1)}$ an NCP unit $\mathcal{U}$ with multiple layers (Fig. 1**b**)
2: Initialize with random weights and freeze $\mathcal{U}$
3: Train $\mathcal{F}_t = \{\mathcal{F}_s^{(1)}, \mathcal{U}\}$ by fine-tuning $\mathcal{F}_s^{(1)}$ on D_t for epochs T
4: Compute β_{eff} according to Eq. (3)



10

Experiment Results



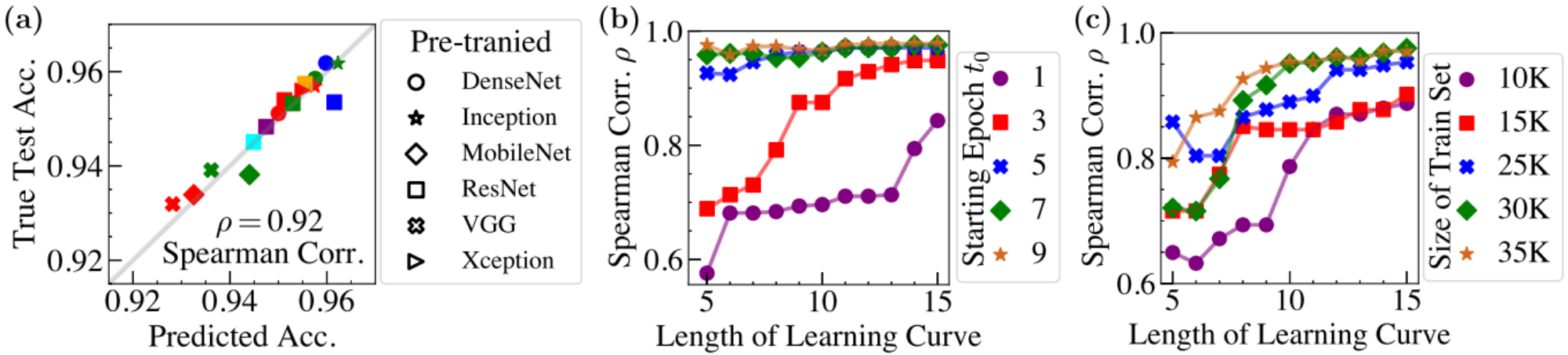


Fig. 3 | The sensitivity analysis of the neural capacitance's predictive capability. **a** Our β_{eff} based prediction of the validation accuracy versus the true test accuracy at epoch 50 of seven representative pre-trained models. Each shape is associated with one type of pre-trained models. Distinct models of the same type are marked in different colors. Because the accuracy of AlexNet is much lower than others, we exclude it for better visualization. Its predicted accuracy is 0.871, and the true test accuracy is 0.868. If it is included, $\rho = 0.93 > 0.92$. **b** Impacts of the starting epoch t_0 of the observations and (**c**) the number of training samples on the ranking performance of our β_{eff} based approach.

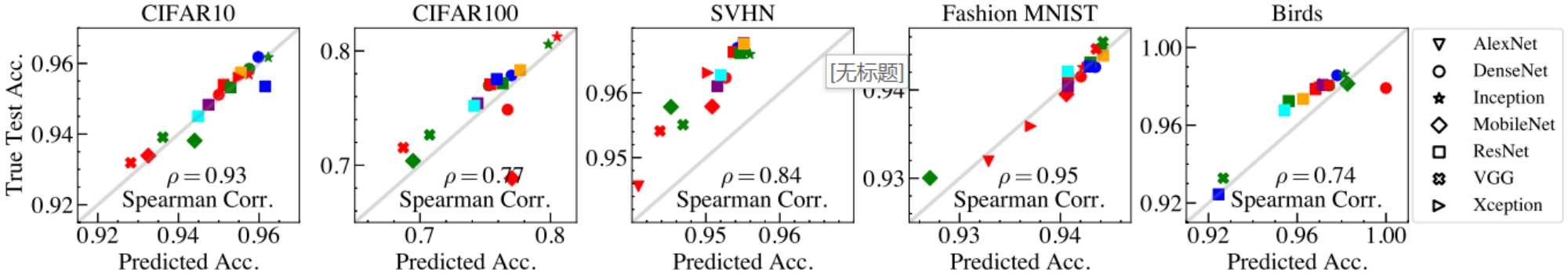


Fig. 4 | The validation accuracy prediction of pre-trained models on all five datasets. The validation accuracy based on β_{eff} is strongly correlated with the true test accuracy of these models after fine-tuning for $T = 50$ epochs. The Spearman's ranking correlation ρ is used to quantify the performance in model selection. Each shape is associated with one type of pre-trained models. Distinct models of the same type are marked in different colors. To be noted, each includes AlexNet in computing ρs.

·3 Model selection

Jiang, Chunheng, et al. "Network properties determine neural network performance." *Nature Communications* 15.1 (2024)

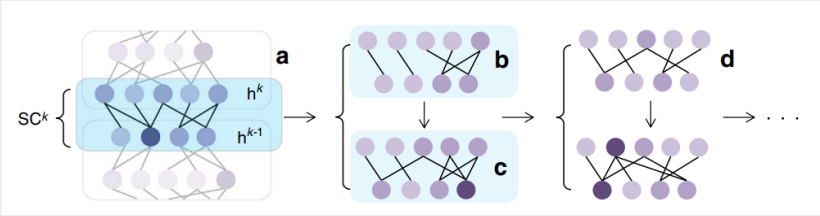
converting a neural network G_A to a line graph G_B

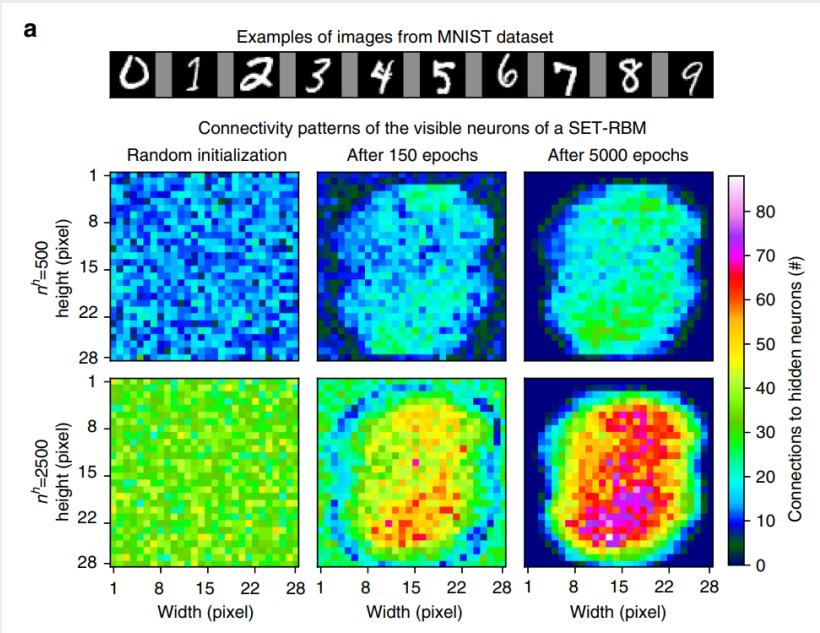
reformulates SGD-based neural network training dynamics as an edge dynamics

a topological property β_{eff} of G_B predicts model performance from early training

Sparse training

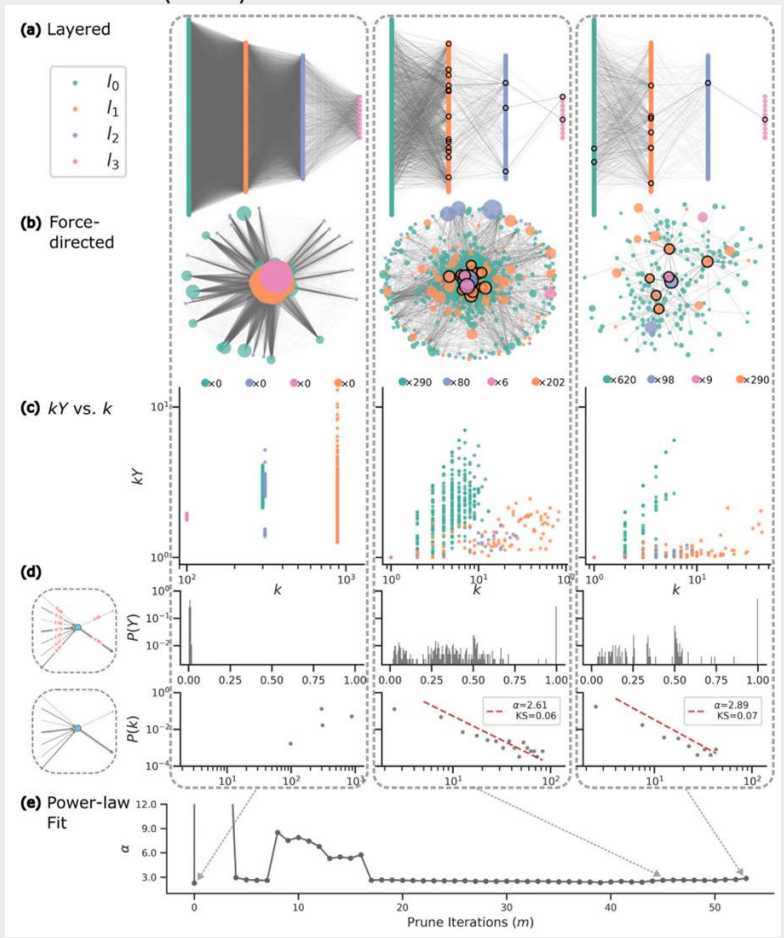
Mocanu, Decebal Constantin, et al. "Scalable training of artificial neural networks with adaptive sparse connectivity inspired by network science." Nature communications 9.1 (2018): 2383.





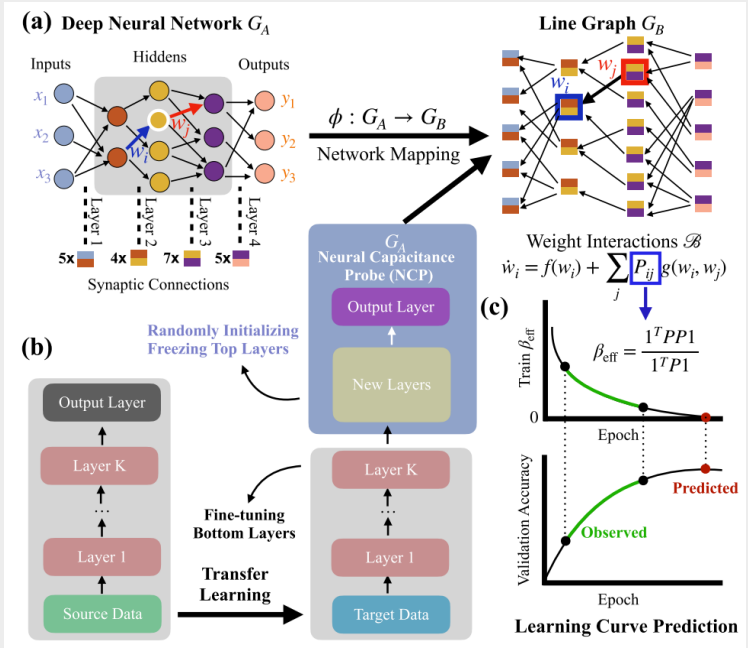
Measurement

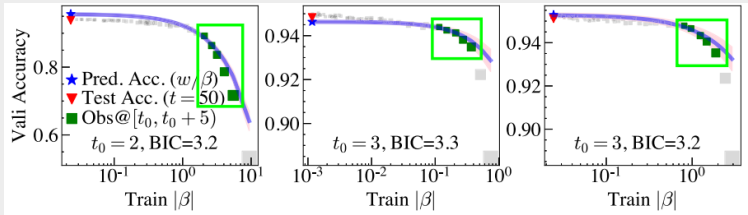
Kang, Chris, et al. “Structural network measures reveal the emergence of heavy-tailed degree distributions in lottery ticket multilayer perceptrons.” Neural Networks (2025)



Model selection

Jiang, Chunheng, et al. "Network properties determine neural network performance." *Nature Communications* 15.1 (2024)





HKU Musketeers Foundation
Institute of Data Science
香港大學同心基金數據科學研究院

Thanks

Jiayu Weng

Neural Networks Through the Lens of Network Science: Structure and Performance