Quantum Ridgelet Transform: Winning Lottery Ticket of Neural Networks with Quantum Computation

Hayata Yamasaki[1], Sathyawageeswar Subramanian[2], Satoshi Hayakawa[3], Sho Sonoda[4]

1: The University of Tokyo

2: University of Warwick, University of Cambridge

3: University of Oxford

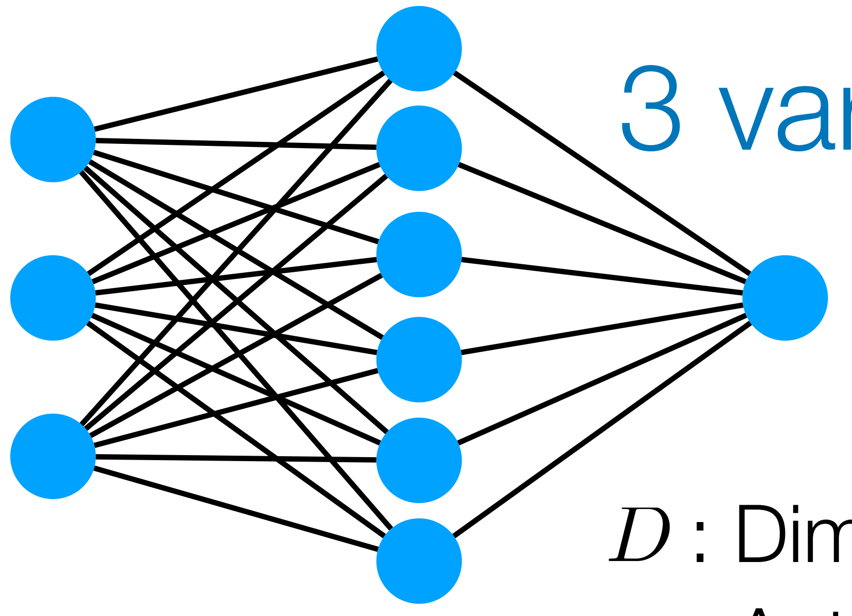
4: RIKEN AIP

arXiv:2301.11936 ICML2023

Background: Integral Representation of Neural Networks

Neural network

$$\boldsymbol{x} = \begin{pmatrix} x_1 \\ \vdots \\ x_D \end{pmatrix}$$



y

3 variables, nonconvex

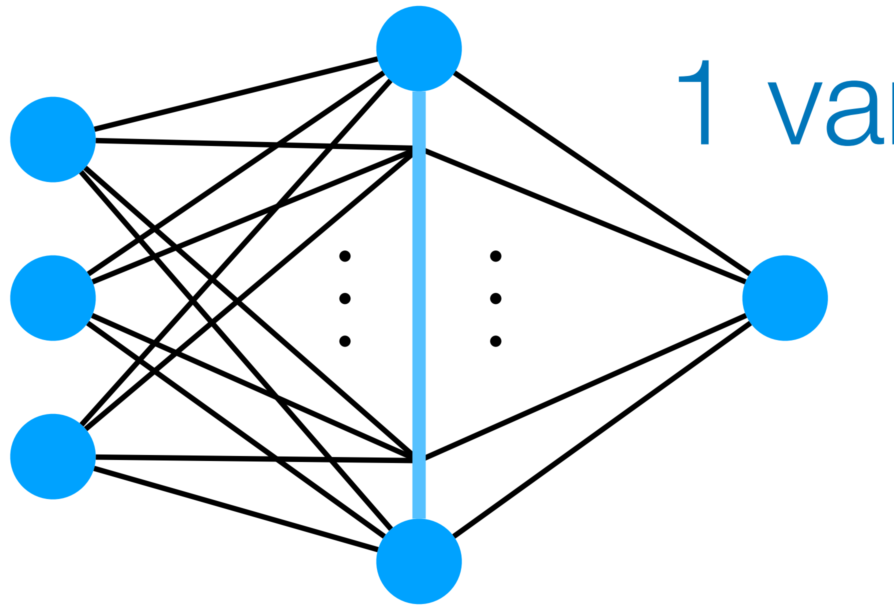
D : Dimension of input

g : Activation function

$$y = f(\boldsymbol{x}) \approx_\epsilon \sum_{n=1}^{N} w_n g(\boldsymbol{a}_n^\top \boldsymbol{x} - \boldsymbol{b}_n)$$

Overparameterization

Integral representation



1 variable, linear

$$f(\boldsymbol{x}) = S[w](\boldsymbol{x}) := \int_{\mathbb{R}^D \times \mathbb{R}} d\boldsymbol{a} db\, w(\boldsymbol{a}, b) g(\boldsymbol{a}^\top \boldsymbol{x} - b)$$

Weight Basis

Ridgelet transform of f: weight that can be used for reconstructing f

$$w(\boldsymbol{a}, b) = R[f](\boldsymbol{a}, b) := \int_{\mathbb{R}^D} d\boldsymbol{x}\, f(\boldsymbol{x}) r(\boldsymbol{a}^\top \boldsymbol{x} - b)$$

r : Ridgelet function

$f \propto S[R[f]]$ if g and r satisfy an admissibility condition

- **<u>Existing work</u>**: Ridgelet transform as **a tool for theoretical analysis** of neural networks
- **<u>Problem</u>**: **Hard to use in practice** due to $\exp(O(D))$ runtime by existing classical algorithms

arXiv:2301.11936

Result:1 Discrete Ridgelet Transform

Definition: Replace **integral** with **sum** over discretized space with **prime** P

$$f(\boldsymbol{x}) = S[w](\boldsymbol{x}) := \int_{\mathbb{R}^D \times \mathbb{R}} d\boldsymbol{a} db \, w(\boldsymbol{a}, b) g(\boldsymbol{a}^\top \boldsymbol{x} - b)$$

$$\mathbb{R}^D \to \mathbb{Z}_P^D := \{0, \dots, P-1\}^D$$

$$w(\boldsymbol{a}, b) = R[f](\boldsymbol{a}, b) := \int_{\mathbb{R}^D} d\boldsymbol{x} \, f(\boldsymbol{x}) r(\boldsymbol{a}^\top \boldsymbol{x} - b)$$

Discretized neural network: $\mathcal{S}[w](\boldsymbol{x}) := \frac{1}{\sqrt{P^D}} \sum_{(\boldsymbol{a}, b) \in \mathbb{Z}_P^D \times \mathbb{Z}_P} w(\boldsymbol{a}, b) g((\boldsymbol{a}^\top \boldsymbol{x} - b) \bmod P)$

Discrete ridgelet transform: $\mathcal{R}[f](\boldsymbol{a}, b) := \frac{1}{\sqrt{P^D}} \sum_{\boldsymbol{x} \in \mathbb{Z}_P^D} f(\boldsymbol{x}) r((\boldsymbol{a}^\top \boldsymbol{x} - b) \bmod P)$

Exact representation $f(\boldsymbol{x}) \propto \mathcal{S}[\mathcal{R}[f]](\boldsymbol{x})$ & Implementable with classical and quantum bits

arXiv:2301.11936

Result 2: Quantum Ridgelet Transform (QRT)

QRT = Discrete ridgelet transform of quantum states

$$|\psi\rangle = \sum_{\boldsymbol{x}} \psi(\boldsymbol{x}) |\boldsymbol{x}\rangle \mapsto \boldsymbol{R} |\psi\rangle = \sum_{\boldsymbol{a},b} \mathcal{R}[\psi](\boldsymbol{a}, b) |\boldsymbol{a}, b\rangle$$

$$\boldsymbol{R} := \frac{1}{\sqrt{P^D}} \sum_{\boldsymbol{a},b} \sum_{\boldsymbol{x}} r((\boldsymbol{a}^\top \boldsymbol{x} - b) \bmod P) |\boldsymbol{a}, b\rangle \langle \boldsymbol{x}|$$: isometry transformation

<u>Thm</u>: We show a quantum algorithm achieving **QRT within runtime** $O(D \times \mathrm{polylog} P)$

$$\mathbb{R}^D \to \mathbb{Z}_P^D := \{0, \ldots, P-1\}^D$$

- **Exponentially faster in D** than the existing classical algorithms for ridgelet transform
- Classical algorithms provide a sequence of values, but **QRT outputs a quantum state**

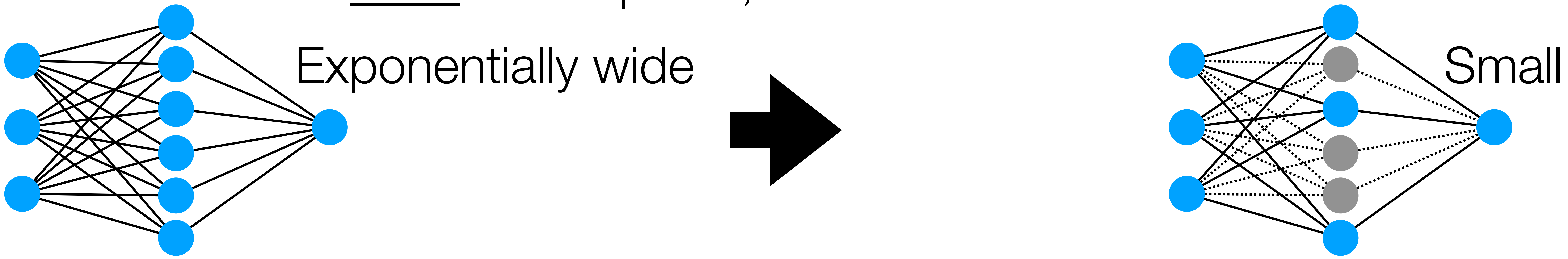
→ Also important to find an end-to-end **application that does not cancel out the speedup**

$$\sum_{\boldsymbol{a},b} \mathcal{R}[\psi](\boldsymbol{a}, b) |\boldsymbol{a}, b\rangle \xrightarrow{\text{measurement}} p(\boldsymbol{a}, b) \propto |\mathcal{R}[\psi](\boldsymbol{a}, b)|^2$$

<u>arXiv:2301.11936</u>

Result 3: Application to Lottery Ticket Hypothesis

Task: Find sparse, trainable subnetwork



Original network: High accuracy

Subnetwork: Achieving same accuracy

$$\frac{1}{\sqrt{P^D}} \sum_{(\boldsymbol{a},b)\in\mathbb{Z}_P^D\times\mathbb{Z}_P} w(\boldsymbol{a},b)\, g((\boldsymbol{a}^\top \boldsymbol{x} - b) \bmod P)$$

Discretized neural network

if we find an appropriate subnetwork

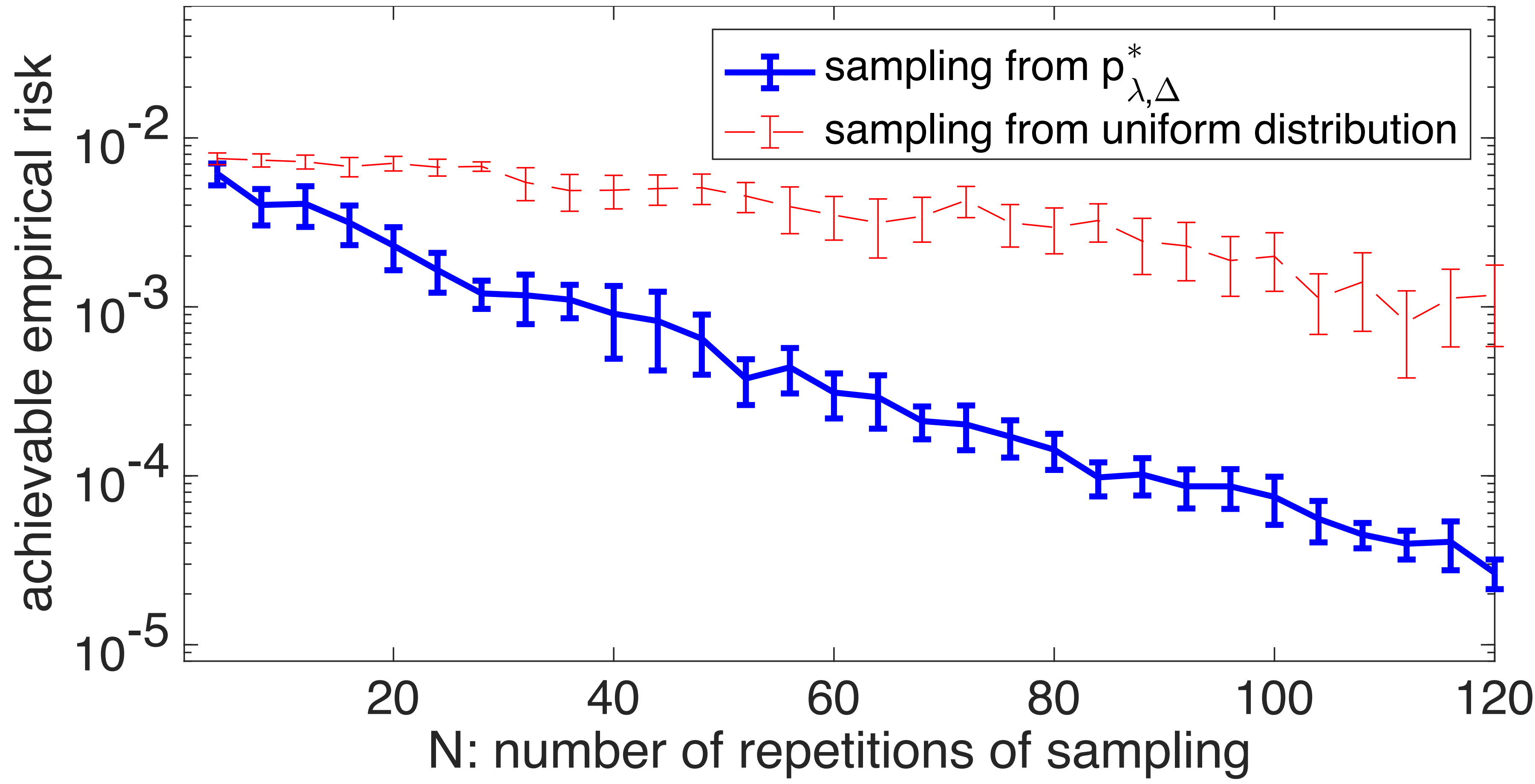
Idea: Find **high-weight nodes** → Use **QRT** to sample parameters of high-weight nodes

Optimized probability distribution

$$p^*(\boldsymbol{a},b) \propto \frac{\frac{1}{\sqrt{P^D}}|w(\boldsymbol{a},b)|^2}{\frac{1}{\sqrt{P^D}}|w(\boldsymbol{a},b)|^2 + \Delta}$$

$(\boldsymbol{a},b)$: Parameters of nodes in the hidden layer

- Represent weight as **amplitude of quantum state**
- Obtain weight by **ridgelet transform** for given examples



arXiv:2301.11936

Summary

- Result 1: Formulation of **discrete ridgelet transform**
- Result 2: Development of quantum algorithm = **quantum ridgelet transform (QRT)**
- Result 3: Application to finding sparse, trainable neural networks in **lottery ticket hypothesis**

Establishing QRT as **a fundamental subroutine for quantum machine learning**
with an end-to-end application to the task in learning with conventional neural networks

arXiv:2301.11936 ICML2023

Hayata Yamasaki, Sathyawageeswar Subramanian, Satoshi Hayakawa, Sho Sonoda,

Quantum Ridgelet Transform: Winning Lottery Ticket of Neural Networks with Quantum Computation

hayata.yamasaki@gmail.com

https://www.hayatayamasaki.com/

See you at the conference

This work was supported by JSPS Overseas Research Fellowship, JST PRESTO Grant Number JPMJPR201A, JPMJPR2125, MEXT Quantum Leap Flagship Program (MEXT QLEAP) JPMXS0118069605, JPMXS0120351339, and Royal Commission for the Exhibition of 1851 Research Fellowship in Science and Engineering