

# Randomness-enhanced expressivity of quantum neural networks

Yadong Wu,<sup>1,2,3</sup> Juan Yao,<sup>4,5,6</sup> Pengfei Zhang,<sup>1,3,\*</sup> and Xiaopeng Li<sup>1,2,3,7,8,†</sup>

<sup>1</sup>Department of Physics, Fudan University, Shanghai, 200438, China

<sup>2</sup>State Key Laboratory of Surface Physics, Key Laboratory of Micro and Nano Photonic Structures (MOE), Institute for Nanoelectronic Devices and Quantum Computing, Fudan University, Shanghai 200438, China

<sup>3</sup>Shanghai Qi Zhi Institute, AI Tower, Xuhui District, Shanghai 200232, China

<sup>4</sup>Shenzhen Institute for Quantum Science and Engineering, Southern University of Science and Technology, Shenzhen 518055, Guangdong, China

<sup>5</sup>International Quantum Academy, Shenzhen 518048, Guangdong, China

<sup>6</sup>Guangdong Provincial Key Laboratory of Quantum Science and Engineering, Southern University of Science and Technology, Shenzhen 518055, Guangdong, China

<sup>7</sup>Shanghai Artificial Intelligence Laboratory, Shanghai 200232, China

<sup>8</sup>Shanghai Research Center for Quantum Sciences, Shanghai 201315, China

As a hybrid of artificial intelligence and quantum computing, quantum neural networks (QNNs) have gained significant attention as a promising application on near-term, noisy intermediate-scale quantum (NISQ) devices. Conventional QNNs are described by parametrized quantum circuits, which perform unitary operations and measurements on quantum states. In this work, we propose a novel approach to enhance the expressivity of QNNs by incorporating randomness into quantum circuits. Specifically, we introduce a random layer, which contains single-qubit gates sampled from an trainable ensemble pooling. The prediction of QNN is then represented by an ensemble average over a classical function of measurement outcomes. We prove that our approach can accurately approximate arbitrary target operators using Uhlmann’s theorem for majorization, which enables observable learning. Our proposal is demonstrated with extensive numerical experiments, including observable learning, Rényi entropy measurement, and image recognition. We find the expressivity of QNNs is enhanced by introducing randomness for multiple learning tasks, which could have broad application in quantum machine learning.

**Introduction.**— In recent years, significant breakthroughs have been made in the field of artificial intelligence. Among various machine learning algorithms, neural networks have played a vital role, thanks to their universal expressivity for deep architectures. As a quantum generalization of neural networks, quantum neural networks (QNNs) have been proposed based on parameterized quantum circuits. QNNs use quantum states instead of classical numbers as inputs[1–4]. However, the evolution of the input quantum states is constrained to be unitary, which limits the expressivity of QNNs. For physical observables, which are linear functions of the input quantum states or density matrices, QNNs can achieve high accuracy only if the target operator shares the same eigenvalues with the measurement operator. For a general situation, it requires introducing auxiliary qubits, as proposed in [5]. To express non-linear functions of the input density matrices, such as purities, traditional approaches introduce multiple replicas, which is unfavorable on near-term, noisy intermediate-scale quantum (NISQ) devices with a limited number of logical qubits. Previous studies have also reported moderate accuracy for more general machine learning tasks, including image recognition [6–10].

In this work, we propose a universal scheme to overcome the expressivity obstacle without the need for additional replicas. Our main inspiration comes from the recent development of the randomized measurement toolbox for quantum simulators [11–35]. In all of these protocols, a measurement is performed after a random unitary gate, and the desired property is predicted through a classical computer after collecting sufficient measurement outcomes. In particular, the random mea-

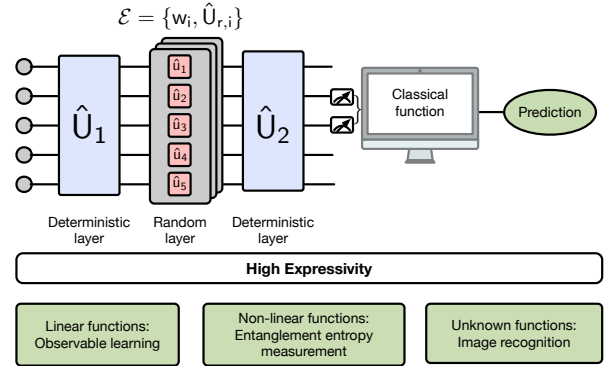


FIG. 1. An illustration is provided for the proposed architecture of randomized quantum neural networks. In this example, the circuit contains two deterministic layers  $\hat{U}_{1(2)}$  and one random layer  $\hat{U}_r$  in between, with the final measurement performed on two qubits. As demonstrated in this work, this architecture shows randomness-enhanced expressivity for a variety of general learning tasks.

surement has been experimentally realized in [36–41]. These developments unveil that randomness plays a central role in extracting information from complex quantum systems efficiently. From a machine learning perspective, this implies that introducing random unitaries can enhance the expressivity of QNNs. This naturally leads to the concept of randomized quantum neural networks, where we collect measurement outcomes from an ensemble of parametrized quantum circuits to make final predictions. Analogous to the different types of layers in classical neural networks, randomized QNNs consist of deterministic layers and random layers. In determinis-

tic layers, the quantum gates contain parameterized quantum gates as in traditional QNNs, while in random layers, they are sampled from trainable ensembles of single-qubit gates. This is illustrated in FIG. 1. We demonstrate the high expressivity of the proposed architecture using several different tasks, including both linear and nonlinear functions of the input density matrix. Our results pave the way towards realizing the universal expressivity ability for QNNs.

**Architecture.**— We begin with a detailed description of randomized QNNs. To be concrete, we focus on the architecture illustrated in FIG. 1 for  $N_{\text{sys}} = 5$  qubits, which comprises two deterministic layers, namely  $\hat{U}_1$  and  $\hat{U}_2$ , with a random single qubit gate layer  $\hat{U}_r$  in between.

Each deterministic layer  $\hat{U}_{l_d}$  ( $l_d = 1, 2$ ) contains a number of units  $\hat{V}_{l_d}^l(\theta_{l_d}^l)$  ( $l \in \{1, 2, \dots, L_{l_d}\}$ ) and each deterministic layer is constructed as

$$\hat{U}_{l_d} = \hat{V}_{l_d}^{L_{l_d}}(\theta_{l_d}^{L_{l_d}}) \dots \hat{V}_{l_d}^2(\theta_{l_d}^2) \hat{V}_{l_d}^1(\theta_{l_d}^1), \quad (1)$$

where  $\{\theta_{l_d}^l\}$  are the parameters of the deterministic layers. In general, the arrangement of two-qubit gates in each deterministic layer allows for a large degree of freedom. In this work, we focus on the standard brick wall architecture with spatial locality. Each unit  $\hat{V}_{l_d}^l$  contains  $N_{\text{sys}} - 1$  two qubit gates and each two qubit gate is a SU(4) matrix which can be parameterized as  $\exp(\sum_j c_j \hat{g}_j)$ . Here  $\hat{g}_j$  is the generator of SU(4) group and  $\{\theta_{l_d}^l\}$  denotes parameters  $\{\mathbf{c}\}$  of all two qubit gates [42]. Nonetheless, alternative choices for each deterministic layer have the potential to enhance the expressivity of QNNs for a fixed number of gates [6].

For the sake of experimental convenience, the random layer  $\hat{U}_r$  comprises a tensor product of single-qubit gates, denoted as  $\hat{u}_1 \otimes \hat{u}_2 \dots \otimes \hat{u}_{N_{\text{sys}}}$ . These gates are sampled from an ensemble

$$\mathcal{E} = \{(w_i, \hat{U}_{r,i} = \hat{u}_1^i(\alpha_i^1) \otimes \hat{u}_2^i(\alpha_i^2) \dots \otimes \hat{u}_{N_{\text{sys}}}^i(\alpha_i^{N_{\text{sys}}}))\}, \quad (2)$$

where  $i = 1, 2, \dots, N_r$  labels different elements and  $w_i$  is the corresponding weight with  $\sum_i w_i = 1$ . Each single qubit gate is parameterized by generators of SU(2) with 3 dimensional real vector  $\alpha_i^q$  ( $q \in \{1, 2, \dots, N_{\text{sys}}\}$ ). Both  $\{w_i\}$  and  $\{\alpha_i^q\}$  are trainable parameters. It is also straightforward to introduce multiple random layers into the full architecture of QNNs. Importantly, it is worth noting the differences between our definition and typical random measurement protocols. Firstly, our random layer can be added at any point in the quantum circuit, not necessarily before the final measurement. Secondly, our definition of  $\mathcal{E}$  allows for non-trivial correlations between single-qubit gates on different sites, which is typically absent in random measurement protocols. Both features are necessary for achieving a high expressivity in QNNs.

We consider a dataset  $\{(|\psi_m\rangle, \mathcal{T}_m)\}$ , in which  $m \in \{1, 2, \dots, N_D\}$  labels different data and  $\mathcal{T}_m$  is the target information for the corresponding state  $|\psi_m\rangle$ . For each unitary  $\hat{U}_{r,i}$  in the ensemble  $\mathcal{E}$ , we perform projective measurements in the computational basis for  $k \sim O(1)$  qubits. The small

number of measured qubits would avoid the barren plateaus, which can be caused by global measurements [43]. In FIG. 1, we set  $k = 2$ , and the measurement yields the probability distribution given by:

$$p_{i,m}^{s,s'} = \langle \psi_m | \hat{U}_1^\dagger \hat{U}_{r,i}^\dagger \hat{U}_2^\dagger (\hat{P}_s^2 \otimes \hat{P}_{s'}^3) \hat{U}_2 \hat{U}_{r,i} \hat{U}_1 | \psi_m \rangle, \quad (3)$$

where the projection operator  $\hat{P}_s^q = \frac{1+s\hat{\sigma}_z^q}{2}$  for  $s = \pm 1$ . Due to the constraint  $\sum_{s,s'} p_{i,m}^{s,s'} = 1$ , there are only 3 non-trivial components of  $p_{i,m}^{s,s'}$ , denoted by the vector  $\mathbf{p}_{i,m}$ . We then use a classical computer to apply a general function  $f_\beta(\cdot)$ , parametrized by  $\beta$ , to the probability distribution  $p_{i,m}^{s,s'}$ , which yields a single outcome denoted by  $\mathcal{P}_{i,m} = f_\beta(\mathbf{p}_{i,m})$ . The classical function can be described by elementary functions in the simplest setting, but is more generally described by classical neural networks. We further average the outcome over the ensemble  $\mathcal{E}$  to obtain the final prediction for the input state  $|\psi_m\rangle$  as:

$$\mathcal{P}_m = \sum_{i=1}^{N_r} w_i \mathcal{P}_{i,m} = \sum_{i=1}^{N_r} w_i f_\beta(\mathbf{p}_{i,m}). \quad (4)$$

We use the mean square error (MSE) as the loss function  $\mathcal{L} = \frac{1}{N_D} \sum_m (\mathcal{P}_m - \mathcal{T}_m)^2$  with a data size of  $N_D$  during the training process. We apply the gradient descent algorithm to optimize the parameters  $\{\theta_{l_d}^l, w_i, \alpha_i^q, \beta\}$  to minimize the loss function  $\mathcal{L}$ , and set the numerical criteria as  $\mathcal{L} < 10^{-5}$  to characterize the accurate prediction. Our method to compute gradients of parameters is explained in the Supplementary Material [42]. In the following sections, we focus on demonstrating high expressivity for randomized QNNs. Our examples range from simple physical tasks including observable learning and Rényi entropy measurement, to standard machine learning tasks such as image recognition.

**Observable learning.**— To show the high expressivity of randomized QNNs, let us consider a simple scenario where the target,  $\mathcal{T}_m$ , is an expectation of a physical observable  $\hat{O}$  with  $\mathcal{T}_m = \langle \psi_m | \hat{O} | \psi_m \rangle$ . For simplicity, focusing on single-qubit measurement with  $k = 1$ , we first investigate whether the randomized QNNs as proposed as in FIG. 1 can approximate the target function  $\mathcal{T}_m$  as accurate as possible for sufficiently deep circuit structures with sufficiently large  $N_r$ . As physical observables are linear in density matrices, a linear function  $f_\beta(x) = \beta_0 + \beta_1 x$  will be applied to the measurement result. Explicitly, we introduce  $\hat{U}_{\text{tot},i}$  for a random realization  $i$  of the quantum circuit. As an example, we have  $\hat{U}_{\text{tot},i} = \hat{U}_2 \hat{U}_{r,i} \hat{U}_1$ . An accurate prediction of the target function requires that

$$\sum_{i=1}^{N_r} w_i \hat{U}_{\text{tot},i}^\dagger (\beta_0 \hat{\sigma}_0^1 + \beta_1 \hat{\sigma}_z^1) \hat{U}_{\text{tot},i} = \hat{O}, \quad (5)$$

where  $\hat{\sigma}_0$  is the identity operator and Pauli matrix  $\hat{\sigma}_z$  is the single-qubit's measurement operator.

For the case of  $N_r = 1$  and  $w_1 = 1$ , our setup reduces to the traditional QNN without randomness. In this scenario, Eq.(5)

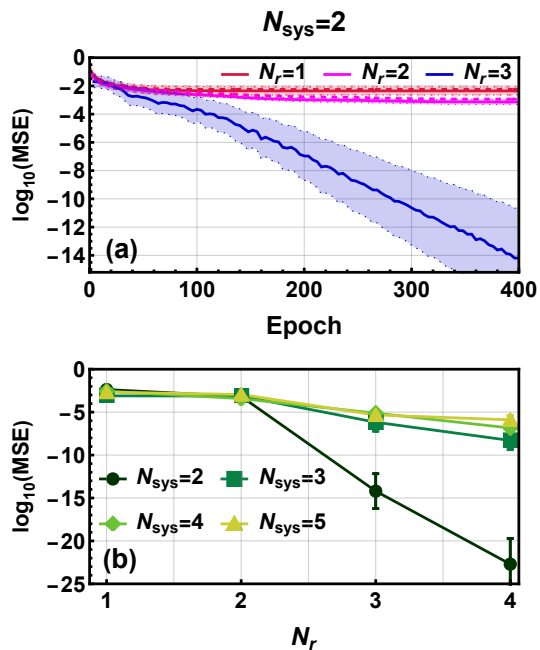


FIG. 2. Predicting observables using QNN with a random layer. (a) The logarithmic training mean square error is shown as a function of the training epoch for observable learning with  $N_{\text{sys}} = 2$ . The solid lines represent the averaged over the training process for 10 different random target operators with independent runs, while the shaded region represents the standard deviation. The dashed lines are the validation loss with the dataset containing 200 samples. (b) The logarithmic mean square error of training dataset for  $N_{\text{sys}} \in \{2, 3, 4, 5\}$  and  $N_r \in \{1, 2, 3, 4\}$ . The markers represents the average over 10 different random target operators with random initializations, and error bars are the standard deviation.

requires that  $(\beta_0 \hat{\sigma}_0^1 + \beta_1 \hat{\sigma}_z^1)$  and  $\hat{O}$  be related by a unitary transformation. Since the unitary transformation preserves the eigenvalues of the operator, the requirement cannot be satisfied for a general operator  $\hat{O}$ . When  $N_r > 1$ , Eq.(5) can be expressed as  $\Phi(\hat{\Sigma}) = \hat{O}$ , where  $\hat{\Sigma} \equiv \beta_1 \hat{\sigma}_z^1 + \beta_0 \hat{\sigma}_0^1$  and  $\Phi(\hat{X})$  is a mixed-unitary channel [44]. For sufficiently complex circuit structures, we expect  $\Phi$  to be generic. In comparison to the  $N_r = 1$  case, there is no constraint from unitarity. However, we still need to ask whether Eq.(5) can be satisfied for an arbitrary operator  $\hat{O}$ . In the following, we prove that the answer to this question is affirmative:

**Step 1.** Mathematically, if there exists a mixed-unitary channel  $\Phi$  such that  $Y = \Phi(X)$ , we say that  $X$  majorizes  $Y$ , denoted by  $Y \prec X$  [45]. Thus, for a randomized QNNs which can accurately predict any observable  $\hat{O}$ , we need to find values of  $\beta_0$  and  $\beta_1$  such that  $\hat{O} \prec \hat{\Sigma}$  for any  $\hat{O}$ .

**Step 2.** According to Uhlmann's theorem for majorization [45, 46],  $\hat{O} \prec \hat{\Sigma}$  if and only if  $\lambda_{\hat{O}} \prec \lambda_{\hat{\Sigma}}$ , where  $\lambda_{\hat{X}}$  is the list of eigenvalues for the operator  $\hat{X}$  in descending order. Here the majorization between two real vectors  $\mathbf{y} \prec \mathbf{x}$  is defined as (i)  $\sum_{j=1}^q x_j \geq \sum_{j=1}^q y_j$  for arbitrary  $1 \leq q < \mathcal{D}$  and (ii)  $\sum_{j=1}^{\mathcal{D}} x_j = \sum_{j=1}^{\mathcal{D}} y_j$ . Here  $\mathcal{D}$  is the dimension of the

vectors. Noting that condition (ii) takes into account the trace-preserving property of mixed-unitary channels.

**Step 3.** We can always find  $\beta_0$  and  $\beta_1$  such that  $\lambda_{\hat{O}} \prec \lambda_{\hat{\Sigma}}$ . Assuming  $\beta_1 > 0$ , the first or last  $\mathcal{D}/2$  components of  $\lambda_{\hat{\Sigma}}$  correspond to the values  $\beta_0 + \beta_1$  or  $\beta_0 - \beta_1$ , respectively. The constant term  $\beta_0$  can then be determined using condition (ii), which gives  $\beta_0 = \mathcal{D}^{-1} \sum_{j=1}^{\mathcal{D}} \lambda_{\hat{O},j}$ . Moreover, condition (i) can always be satisfied for sufficiently large  $\beta_1$ . This proves the existence of  $\beta_0$  and  $\beta_1$  such that  $\hat{O} \prec \hat{\Sigma}$ .

Although randomized QNNs have the potential to express arbitrary operators, it is difficult to determine an upper bound or a required value for  $N_r$  in practical learning tasks. It is unfavorable to have large  $N_r$  or a large number of random layers, especially in NISQ devices. Therefore, we turn to numerical simulations of the randomized QNNs, and investigate practical requirements on  $N_r$ . Since the basis change can be efficiently captured by the deterministic layer  $\hat{U}_1$ , we focus on observables  $\hat{O}$  that are diagonal in the computational basis. For simplicity, we further set  $\hat{U}_1 = \hat{I}$  and  $\hat{U}_2$  composed by  $L_2$  units of a brick wall structure [42]. For each system size  $N_{\text{sys}}$ , we test whether a random diagonal operator  $\hat{O}$  can be predicted accurately for different values of  $N_r$  by monitoring the training loss for a sufficiently large dataset. As an example, we plot the logarithmic training mean square error  $\log_{10}(\text{MSE})$  as a function of the training epoch for  $N_{\text{sys}} = 2$  in FIG. 2 (a). The curves are averaged over 10 operators with random eigenvalues from the uniform distribution  $[-2.5, 2.5]$ . When we increase  $N_r$  from 1 to 3, there is a rapid decrease in the training loss for large training epochs. The result shows that  $N_r = 3$  is sufficient for learning general operators for  $N_{\text{sys}} = 2$  where the loss  $\mathcal{L}$  can be decreased to  $10^{-14}$ . We further extend the system size  $N_{\text{sys}}$  to study how it affects the number of required random gates. The results are shown in FIG. 2 (b). Although we are limited to small system sizes  $N_{\text{sys}} \in \{2, 3, 4, 5\}$ , the results clearly show weak dependence of  $N_r$  on  $N_{\text{sys}}$ . The training results show that  $N_r = 3$  already gives highly accurate predictions for  $N_{\text{sys}} = 5$ .

**Rényi entropy measurement.**— We now consider targets that are non-linear functions of density matrices. One example is the Rényi entropy, which is also of experimental interest. To compute the Rényi entropy for a subsystem  $A$  consisting of the central  $N_{\text{sub}}$  qubits, we first calculate the reduced density matrix  $\hat{\rho}_A$  of an input state  $|\psi_m\rangle$  by tracing out the degrees of freedom of the complementary subsystem  $\bar{A}$ . We then select the target as

$$\mathcal{T}_m = \text{Tr}_A[\hat{\rho}_A^n], \quad (6)$$

which is related to the  $n$ -th Rényi entropy through  $S_A^{(n)} = -\frac{1}{n-1} \ln(\mathcal{T}_m)$ . Since we are directly measuring a local property of the input wavefunction, it is reasonable to fix  $\hat{U}_1$  and  $\hat{U}_2$  to the identity matrix  $\hat{I}$  and focus on the random layer  $\hat{U}_r$  with  $k = N_{\text{sub}}$ . This approach provides a minimum guaranteed expressivity of randomized QNNs. Because the target  $\mathcal{T}_m$  is proportional to  $\rho^n$ , we choose the function  $f_{\beta}(\mathbf{x})$  to be a polynomial up to the  $n$ -th order. However, it is worth noting

that lower order polynomials may also work in certain cases [47]. We prepare a dataset with random states  $|\psi_m\rangle$ , the detailed description of which is provided in the Supplementary Material [42]. The numerical results for  $n = 2, 3$ ,  $N_{\text{sys}} = 5$  and  $N_{\text{sub}} = 1, 2$  are shown in FIG. 3. To achieve accurate predictions, we need  $N_r = 3$  for  $N_{\text{sub}} = 1$  and  $N_r = 9$  for  $N_{\text{sub}} = 2$ . The blue lines in FIG. 3 demonstrate that the loss  $\mathcal{L}$  is able to reach a value of  $10^{-5}$  and still keep decreasing, indicating the ability to make accurate predictions. We have also discussed the saturation of  $N_r$  for  $n = 2, N_{\text{sub}} = 2$  and the required number of  $N_r$  if we instead consider  $n = 3$ . The results are shown in the Supplementary Material [42].

It is interesting to compare our results to the proposed random measurement protocol for Rényi entropies. Our results indicate that  $N_r$  scales as  $3^{N_{\text{sub}}}$  when measuring Rényi entropies. In comparison, the previous protocol required each single-qubit gate  $\hat{u}_q^i$  to be sampled from the circular unitary ensemble [18, 40]. For  $n = 2$ , the circular unitary ensemble can be replaced by unitary 2-designs, which are known to be the Clifford group. Since the single-qubit Clifford group contains 24 elements, the total number of unitary matrices  $\hat{U}_{r,i}$  would naively scale as  $24^{N_{\text{sub}}}$ . However, in practice, this can be significantly reduced because randomized measurement protocols only require  $N_s$  snapshots sampled from the full ensemble. The theoretical bound of  $N_s$  for measuring general linear observables in a subsystem with  $N_{\text{sub}}$  qubits using random Pauli measurements up to an error  $\epsilon$  is given by  $N_s \gtrsim 3^{N_{\text{sub}}} / \epsilon^2$  [18, 19]. Consequently, in this quantum neural network structure, the number of unitary matrices  $\hat{U}_{r,i}$  that contribute is at most  $3^{N_{\text{sub}}}$ , as in our randomized QNNs.

**Image recognition.**— Finally, we turn our attention to image recognition, a more practical machine learning task, in order to demonstrate the enhanced expressivity of randomized QNNs. In this case, we use Google’s ‘Street View of House Number (SVHN)’ dataset as an example [48]. Each image in the dataset corresponds to an integer number. For demonstration purposes, we select two categories of images containing the numbers ‘1’ and ‘4’. Initially, we compress each image into an  $8 \times 8$  pixel format, resulting in a 64-dimensional real vector, which can be equivalently represented as a 32-dimensional complex vector. Subsequently, we encode the image into the input wave function using  $N_{\text{sys}} = 5$  qubits [42]. Unlike previous tasks, the mapping between the input and the output is highly complex and non-local, lacking a simple understanding. Consequently, we allow both  $\hat{U}_1$  and  $\hat{U}_2$  to be trainable. After measuring a single qubit, we choose a 5<sup>th</sup>-order polynomial for the function  $f_\beta(x)$ . Since the image recognition is a two category classification task, after obtaining the final ensemble average prediction  $\mathcal{P}_m$ , we apply a logistic-sigmoid function to restrict the prediction in the region (0,1) with  $\mathcal{G}_m = 1/(1 + \exp(-\mathcal{P}_m))$ . And we use the cross-entropy as the loss function  $\mathcal{L} = \frac{1}{N_D} \sum_m -\mathcal{T}_m \log(\mathcal{G}_m) - (1 - \mathcal{T}_m) \log(1 - \mathcal{G}_m)$  to optimize the parameters in the randomized QNN. The accuracy  $F = \frac{1}{N_D} \sum_m |[\text{sign}(\mathcal{G}_m - 0.5) + 1]/2 - \mathcal{T}_m|$  for  $N_r = 1$  and  $N_r = 4$

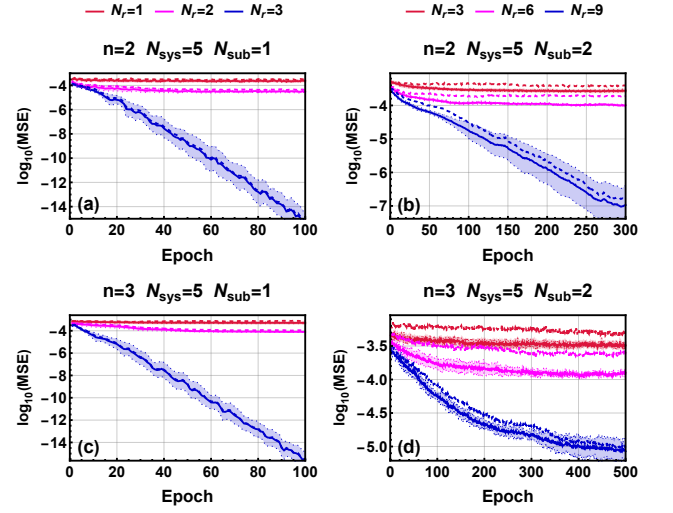


FIG. 3. Predicting Rényi entropies using QNN with a random layer. The logarithmic training mean square error is shown as a function of the training epoch for purity with  $N_{\text{sys}} = 5$  and (a)  $n = 2, N_{\text{sub}} = 1$ , (b)  $n = 2, N_{\text{sub}} = 2$ , (c)  $n = 3, N_{\text{sub}} = 1$  or (d)  $n = 3, N_{\text{sub}} = 2$ . The results are averaged over the training process for 10 different random initializations, and the shaded region represents the standard deviation. The dashed lines are the validation loss with the dataset containing 200 samples.

are shown in FIG. 4. For  $N_r = 1$ , the accuracy saturates at approximately 0.8 after a large number of epochs, while the averaged accuracy for the test dataset reaches 69.8%. The introduction of a single random layer with  $N_r = 4$  significantly enhances the accuracy of the predictions. In this case, the training dataset achieves an accuracy higher than 90%, and the average accuracy for the test dataset is 82.29%. The utilization of a non-trivial random layer with  $N_r = 4$  demonstrates a significant improvement in the prediction capabilities of QNNs, indicating the enhanced expressivity of our randomized QNN architecture.

**Outlook.**— This work introduces the concept of randomized quantum neural networks, which include random layers where quantum gates are selected from an ensemble of unitary matrices. It is proven that these random layers provide universal expressivity for general physical observables using Uhlmann’s theorem for majorization. Numerical simulations further show that this architecture achieves high expressivity for non-linear functions of the density matrix, such as Rényi entropies and image recognition, with small ensemble sizes  $N_r$ . These results indicate that the proposed method has potential for broad applications in NISQ devices. We remark that adding a random layer to the QNNs causes an extra computational cost proportional to  $N_r$ . Nonetheless, introducing randomness into QNNs while maintaining the same computational cost still improves the learning performance significantly [42]. We further highlight the differences between our architecture and the proposal presented in a very recent paper [49]. Their work also incorporates a series of parameter-

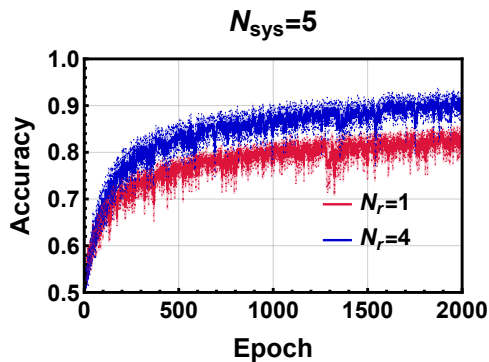


FIG. 4. *Image recognition by QNN with a random layer.* The accuracy is shown as a function of the training epoch for the image recognition task. The results are averaged over the training process for 10 different random initializations, and the shaded region represents the standard deviation.

ized quantum circuits, where the circuit consists of multiple parametrized (controlled-) rotations that share the same parameter. In contrast, our architecture features only a few random layers described by a tensor product of single-qubit gates, making its training process more efficient.

While the focus of this work is on parameterized quantum circuits with brick wall structures, it is straightforward to combine this novel architecture with other proposals to further improve expressivity or learning efficiency. For instance, it is possible to add ancilla qubits and explore more sophisticated architectures for the deterministic layers. Additionally, it would be interesting to investigate the impact of random layers on other quantum machine learning algorithms beyond traditional quantum neural networks [50–55], such as quantum autoencoders [50, 51].

**Acknowledgement.**— We thank Yingfei Gu, Ning Sun, Ce Wang, Hai Wang, and Yi-Zhuang You for helpful discussions. YW and XL are supported by National Program on Key Basic Research Project of China (Grant No. 2021YFA1400900), National Natural Science Foundation of China (Grants No. 11934002), Shanghai Municipal Science and Technology Major Project (Grant No. 2019SHZDZX01). YW is supported by the National Natural Science Foundation of China (Grant No. 12174236). JY is supported by the National Natural Science Foundation of China (Grant No. 11904190).

\* pengfeizhang.physics@gmail.com

† xiaopeng.li@fudan.edu.cn

- [1] Maria Schuld, Alex Bocharov, Krysta M. Svore, and Nathan Wiebe. Circuit-centric quantum classifiers. *Phys. Rev. A*, 101:032308, Mar 2020.
- [2] Jin-Guo Liu and Lei Wang. Differentiable learning of quantum circuit born machines. *Phys. Rev. A*, 98:062324, Dec 2018.
- [3] Edward Farhi and Hartmut Neven. Classification with quantum neural networks on near term processors, 2018.

- [4] Marcello Benedetti, Erika Lloyd, Stefan Sack, and Mattia Fiorentini. Parameterized quantum circuits as machine learning models. *Quantum Science and Technology*, 4(4):043001, nov 2019.
- [5] Yadong Wu, Juan Yao, Pengfei Zhang, and Hui Zhai. Expressivity of quantum neural networks. *Phys. Rev. Res.*, 3:L032049, Aug 2021.
- [6] Yadong Wu, Pengfei Zhang, and Hui Zhai. Scrambling ability of quantum neural network architectures. *Phys. Rev. Res.*, 3:L032057, Sep 2021.
- [7] Leonardo Bianchi, Jason Pereira, and Stefano Pirandola. Generalization in quantum machine learning: A quantum information standpoint. *PRX Quantum*, 2:040321, Nov 2021.
- [8] Matthias C. Caro, Hsin-Yuan Huang, M. Cerezo, Kunal Sharma, Andrew Sornborger, Lukasz Cincio, and Patrick J. Coles. Generalization in quantum machine learning from few training data. *Nature Communications*, 13(1):4919, 2022.
- [9] Amira Abbas, David Sutter, Christa Zoufal, Aurelien Lucchi, Alessio Figalli, and Stefan Woerner. The power of quantum neural networks. *Nature Computational Science*, 1(6):403–409, 2021.
- [10] Maria Schuld, Ryan Sweke, and Johannes Jakob Meyer. Effect of data encoding on the expressive power of variational quantum-machine-learning models. *Phys. Rev. A*, 103:032430, Mar 2021.
- [11] Xiao-Liang Qi, Emily J. Davis, Avikar Periwal, and Monika Schleier-Smith. Measuring operator size growth in quantum quench experiments. 6 2019.
- [12] Scott Aaronson. Shadow Tomography of Quantum States. *arXiv e-prints*, page arXiv:1711.01053, November 2017.
- [13] Andreas Ketterer, Nikolai Wyderka, and Otfried Gühne. Characterizing multipartite entanglement with moments of random correlations. *Phys. Rev. Lett.*, 122:120505, Mar 2019.
- [14] A. Elben, B. Vermersch, M. Dalmonte, J. I. Cirac, and P. Zoller. Rényi entropies from random quenches in atomic hubbard and spin models. *Phys. Rev. Lett.*, 120:050406, Feb 2018.
- [15] S. J. van Enk and C. W. J. Beenakker. Measuring  $\text{Tr}\rho^n$  on single copies of  $\rho$  using random measurements. *Phys. Rev. Lett.*, 108:110503, Mar 2012.
- [16] A. Elben, B. Vermersch, C. F. Roos, and P. Zoller. Statistical correlations between locally randomized measurements: A toolbox for probing entanglement in many-body quantum states. *Phys. Rev. A*, 99:052323, May 2019.
- [17] Lukas Knips, Jan Dziewior, Waldemar Klobus, Wiesław Laskowski, Tomasz Paterek, Peter J. Shadbolt, Harald Weinfurter, and Jasmin D. A. Meinecke. Multipartite entanglement analysis from random correlations. *npj Quantum Information*, 6(1):51, 2020.
- [18] Hsin-Yuan Huang, Richard Kueng, and John Preskill. Predicting many properties of a quantum system from very few measurements. *Nature Physics*, 16(10):1050–1057, 2020.
- [19] Andreas Elben, Steven T. Flammia, Hsin-Yuan Huang, Richard Kueng, John Preskill, Benoît Vermersch, and Peter Zoller. The randomized measurement toolbox. *Nature Reviews Physics*, 5(1):9–24, 2023.
- [20] Hong-Ye Hu and Yi-Zhuang You. Hamiltonian-driven shadow tomography of quantum states. *Phys. Rev. Res.*, 4:013054, Jan 2022.
- [21] Senrui Chen, Wenjun Yu, Pei Zeng, and Steven T. Flammia. Robust shadow estimation. *PRX Quantum*, 2:030348, Sep 2021.
- [22] Atithi Acharya, Siddhartha Saha, and Anirvan M. Sengupta. Informationally complete POVM-based shadow tomography. *arXiv e-prints*, page arXiv:2105.05992, May 2021.
- [23] Ryan Levy, Di Luo, and Bryan K. Clark. Classical Shadows for

- Quantum Process Tomography on Near-term Quantum Computers. *arXiv e-prints*, page arXiv:2110.02965, October 2021.
- [24] Andrew Zhao, Nicholas C. Rubin, and Akimasa Miyake. Fermionic partial tomography via classical shadows. *Phys. Rev. Lett.*, 127:110504, Sep 2021.
- [25] Kianna Wan, William J. Huggins, Joonho Lee, and Ryan Babush. Matchgate Shadows for Fermionic Quantum Simulation. *arXiv e-prints*, page arXiv:2207.13723, July 2022.
- [26] Hsin-Yuan Huang, Richard Kueng, Giacomo Torlai, Victor V. Albert, and John Preskill. Provably efficient machine learning for quantum many-body problems. *Science*, 377(6613):eabk3333, 2022.
- [27] Kaifeng Bu, Dax Enshan Koh, Roy J. Garcia, and Arthur Jaffe. Classical shadows with Pauli-invariant unitary ensembles. *arXiv e-prints*, page arXiv:2202.03272, February 2022.
- [28] Jonathan Kunjummen, Minh C. Tran, Daniel Carney, and Jacob M. Taylor. Shadow process tomography of quantum channels. *Phys. Rev. A*, 107(4):042403, April 2023.
- [29] Saumya Shivam, Curt W. von Keyserlingk, and Shivaji L. Sondhi. On classical and hybrid shadows of quantum states. *SciPost Physics*, 14(5):094, May 2023.
- [30] Matteo Ippoliti, Yaodong Li, Tibor Rakovszky, and Vedika Khemani. Operator relaxation and the optimal depth of classical shadows. *Phys. Rev. Lett.*, 130:230403, Jun 2023.
- [31] Christian Bertonì, Jonas Haferkamp, Marcel Hinsche, Marios Ioannou, Jens Eisert, and Hako Pashayan. Shallow shadows: Expectation estimation using low-depth random Clifford circuits. *arXiv e-prints*, page arXiv:2209.12924, September 2022.
- [32] Mirko Arienzo, Markus Heinrich, Ingo Roth, and Martin Kliesch. Closed-form analytic expressions for shadow estimation with brickwork circuits. *arXiv e-prints*, page arXiv:2211.09835, November 2022.
- [33] Ahmed A. Akhtar, Hong-Ye Hu, and Yi-Zhuang You. Scalable and Flexible Classical Shadow Tomography with Tensor Networks. *Quantum*, 7:1026, 2023.
- [34] Hong-Ye Hu, Soonwon Choi, and Yi-Zhuang You. Classical shadow tomography with locally scrambled quantum dynamics. *Phys. Rev. Res.*, 5:023027, Apr 2023.
- [35] Dax Enshan Koh and Sabee Grewal. Classical shadows with noise. *Quantum*, 6:776, 2022.
- [36] Joseph Vovrosh and Johannes Knolle. Confinement and entanglement dynamics on a digital quantum computer. *Scientific Reports*, 11(1):11577, 2021.
- [37] Min Yu, Dongxiao Li, Jingcheng Wang, Yaoming Chu, Pengcheng Yang, Musang Gong, Nathan Goldman, and Jianming Cai. Experimental estimation of the quantum fisher information from randomized measurements. *Phys. Rev. Res.*, 3:043122, Nov 2021.
- [38] Crystal Noel, Pradeep Niroula, Daiwei Zhu, Andrew Risinger, Laird Egan, Debopriyo Biswas, Marko Cetina, Alexey V. Gorshkov, Michael J. Gullans, David A. Huse, and Christopher Monroe. Measurement-induced quantum phases realized in a trapped-ion quantum computer. *Nature Physics*, 18(7):760–764, 2022.
- [39] Jin Ming Koh, Shi-Ning Sun, Mario Motta, and Austin J. Minnich. Experimental realization of a measurement-induced entanglement phase transition on a superconducting quantum processor, 2022.
- [40] Tiff Brydges, Andreas Elben, Petar Jurcevic, Benoît Vermersch, Christine Maier, Ben P. Lanyon, Peter Zoller, Rainer Blatt, and Christian F. Roos. Probing renyi entanglement entropy via randomized measurements. *Science*, 364(6437):260–263, 2019.
- [41] G.I. Struchalin, Ya. A. Zagorovskii, E.V. Kovlakov, S.S. Straupe, and S.P. Kulik. Experimental estimation of quantum state properties from classical shadows. *PRX Quantum*, 2:010307, Jan 2021.
- [42] See Supplementary Material for: (i). Structures of deterministic layers; (ii). Gradient decent method of QNN’s parameters. (iii). Training details of the observables learning task; (iv). Training details of the Rényi entropy measurement task; (v). Training details of the pattern recognition task; (vi). Comparison with fixed Computational Cost.
- [43] Marco Cerezo, Akira Sone, Tyler Volkoff, Lukasz Cincio, and Patrick J Coles. Cost function dependent barren plateaus in shallow parametrized quantum circuits. *Nature communications*, 12(1):1791, 2021.
- [44] John Watrous. *The Theory of Quantum Information*. Cambridge University Press, 2018.
- [45] Michael A Nielsen. An introduction to majorization and its applications to quantum mechanics. *Lecture Notes, Department of Physics, University of Queensland, Australia*, 2002.
- [46] Peter Alberti and Armin Uhlmann. Stochasticity and partial order. doubly stochastic maps and unitary mixing. 01 1982.
- [47] The third order of single qubit reduced density matrix  $tr[\hat{\rho}_a^3]$  is the second order of the elements of the density matrix.
- [48] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. 2011.
- [49] Xiaokai Hou, Guanyu Zhou, Qingyu Li, Shan Jin, and Xiaoting Wang. A duplication-free quantum neural network for universal approximation. *Science China Physics, Mechanics & Astronomy*, 66(7):270362, 2023.
- [50] Jonathan Romero, Jonathan P Olson, and Alan Aspuru-Guzik. Quantum autoencoders for efficient compression of quantum data. *Quantum Science and Technology*, 2(4):045001, 2017.
- [51] Dmytro Bondarenko and Polina Feldmann. Quantum autoencoders to denoise quantum data. *Physical review letters*, 124(13):130502, 2020.
- [52] Seunghyeok Oh, Jaeho Choi, and Joongheon Kim. A tutorial on quantum convolutional neural networks (qcnn). In *2020 International Conference on Information and Communication Technology Convergence (ICTC)*, pages 236–239. IEEE, 2020.
- [53] Tak Hur, Leeseok Kim, and Daniel K Park. Quantum convolutional neural network for classical data classification. *Quantum Machine Intelligence*, 4(1):3, 2022.
- [54] Iris Cong, Soonwon Choi, and Mikhail D Lukin. Quantum convolutional neural networks. *Nature Physics*, 15(12):1273–1278, 2019.
- [55] Matthias C. Caro, Hsin-Yuan Huang, M. Cerezo, Kunal Sharma, Andrew Sornborger, Lukasz Cincio, and Patrick J. Coles. Generalization in quantum machine learning from few training data. *Nature Communications*, 13(1):4919, 2022.

## Supplementary Material for "Randomness-enhanced Expressivity of Quantum Neural Networks"

Yadong Wu,<sup>1,2,3</sup> Juan Yao,<sup>4,5,6</sup> Pengfei Zhang,<sup>1,3,\*</sup> and Xiaopeng Li<sup>1,2,3,7,8,†</sup>

<sup>1</sup>Department of Physics, Fudan University, Shanghai, 200438, China

<sup>2</sup>State Key Laboratory of Surface Physics, Key Laboratory of Micro and Nano Photonic Structures (MOE), Institute for Nanoelectronic Devices and Quantum Computing, Fudan University, Shanghai 200438, China

<sup>3</sup>Shanghai Qi Zhi Institute, AI Tower, Xuhui District, Shanghai 200232, China

<sup>4</sup>Shenzhen Institute for Quantum Science and Engineering, Southern University of Science and Technology, Shenzhen 518055, Guangdong, China

<sup>5</sup>International Quantum Academy, Shenzhen 518048, Guangdong, China

<sup>6</sup>Guangdong Provincial Key Laboratory of Quantum Science and Engineering, Southern University of Science and Technology, Shenzhen 518055, Guangdong, China

<sup>7</sup>Shanghai Artificial Intelligence Laboratory, Shanghai 200232, China

<sup>8</sup>Shanghai Research Center for Quantum Sciences, Shanghai 201315, China

### STRUCTURES OF DETERMINISTIC LAYERS

In this section, we'll present the architecture of deterministic layers. The entire deterministic layer  $\hat{U}$  is composed of a number of unites, i.e.  $\hat{U} = \hat{V}^L \hat{V}^{L-1} \dots \hat{V}^1$ . Each unit  $\hat{V}^a$  contains multiple two-qubit gates  $\hat{v}_{ij}$  where  $i, j$  represent the qubits indices. Each two-qubit gate is parameterized as  $\hat{v}_{ij} = \exp(\sum_w \theta_{ij}^w \hat{g}_w)$  where  $\{\hat{g}_k\}$  are the generators of the SU(4) group. Fig.[1] illustrates the brick wall architecture of one unit. The explicit form of this single-unit neural network is as follow:

$$\hat{V} = [\hat{\sigma}_0^1 \otimes \hat{v}_{23} \otimes \hat{v}_{45}][\hat{v}_{12} \otimes \hat{v}_{34} \otimes \hat{\sigma}_0^5], \quad (1)$$

where  $\hat{\sigma}_0$  is the identity of the single qubit.

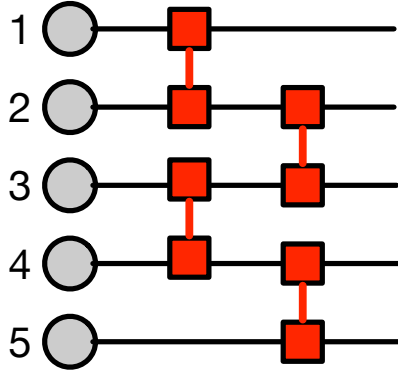


FIG. 1: The brick wall structure of 5-qubit system.

### GRADIENT DECENT METHOD OF QNN'S PARAMETERS

In this section, we present the calculation details of the gradients of parameters in QNNs with the random layer. The QNN architecture consists of three components: deterministic layers  $\hat{U}_1$  and  $\hat{U}_2$ , a random layer  $\mathcal{E} = \{w_i, \hat{U}_{r,i}\}$  situated between deterministic layers, and a classical function  $f_\beta$ . The final prediction for the input state  $|\psi_m\rangle$  is computed as follows:

$$\mathcal{P}_m = \sum_{i=1}^{N_r} w_i \mathcal{P}_{i,m} = \sum_{i=1}^{N_r} w_i f_\beta(\mathbf{p}_{i,m}). \quad (2)$$

where  $\{w_i\}$  represents the probabilities of the random unitary operators in the ensemble  $\mathcal{E}$ ,  $\{\beta\}$  are variational parameters in the classical function, and  $\mathbf{p}_{i,m}$  denotes the measurements from the quantum circuit. In quantum machine learning tasks, the loss

function  $\mathcal{L}$  is a function of  $\mathcal{P}_m$ , and the gradient with respect to  $\{w_i, \beta\}$  can be easily obtained.

$$\frac{\partial \mathcal{L}}{\partial w_i} = \sum_m \frac{\partial \mathcal{L}}{\partial \mathcal{P}_m} \mathcal{P}_{i,m} \quad (3)$$

$$\frac{\partial \mathcal{L}}{\partial \beta} = \sum_m \frac{\partial \mathcal{L}}{\partial \mathcal{P}_m} \sum_i^{N_r} w_i \frac{\partial f_{\beta}(\mathbf{p}_{i,m})}{\partial \beta} \quad (4)$$

Each element of measurements is given by  $p_{i,m}^s = \langle \psi_m | \hat{U}_1^\dagger \hat{U}_{r,i}^\dagger \hat{U}_2^\dagger \hat{M}_s \hat{U}_2 \hat{U}_{r,i} \hat{U}_1 | \psi_m \rangle$  where  $\hat{M}_s$  is a general measurement operator.  $\hat{U}_{r,i}$  is parametrized by  $\{\alpha_i^q\}$  that

$$\begin{aligned} \hat{U}_{r,i} &= \hat{u}_1^i(\alpha_i^1) \otimes \hat{u}_2^i(\alpha_i^2) \dots \otimes \hat{u}_{N_{\text{sys}}}^i(\alpha_i^{N_{\text{sys}}}) \\ \hat{u}_q^i &= \exp(\alpha_i^q \cdot \sigma), \end{aligned} \quad (5)$$

where  $\alpha_i^q$  is a three-dimensional real vector and  $\sigma$  is a three-dimensional vector composed of the generators of the SU(2) group. The gradient of  $\alpha_i^q$  can be written as:

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial \alpha_i^q} &= \sum_m \frac{\partial \mathcal{L}}{\partial \mathcal{P}_m} \sum_i^{N_r} w_i \sum_s \frac{\partial f_{\beta}(\mathbf{p}_{i,m})}{\partial p_{i,m}^s} \\ \frac{\partial p_{i,m}^s}{\partial \alpha_i^q} &= \langle \psi_m | \hat{U}_1^\dagger \hat{U}_{r,i}^\dagger \hat{U}_2^\dagger \hat{M}_s \hat{U}_2 \frac{\partial \hat{U}_{r,i}}{\partial \alpha_i^q} \hat{U}_1 | \psi_m \rangle + h.c. \\ \frac{\partial \hat{U}_{r,i}}{\partial \alpha_i^q} &= \hat{u}_1^i(\alpha_i^1) \otimes \hat{u}_2^i(\alpha_i^2) \dots \otimes \frac{\partial \hat{u}_q^i}{\partial \alpha_i^q} \otimes \dots \otimes \hat{u}_{N_{\text{sys}}}^i(\alpha_i^{N_{\text{sys}}}) \end{aligned} \quad (6)$$

Then we can utilize the matrix exponential gradient [1] to calculate the gradient:  $\frac{\partial \hat{u}_q^i}{\partial \alpha_i^q}$

Similarly, for the parameters  $\{\theta_{i_d}^l\}$  in the circuit of deterministic layers  $\hat{U}_1, \hat{U}_2$ , the gradient of parameters in this neural network can also be obtained from the matrix exponential gradient [1].

## TRAINING DETAILS OF OBSERVABLES LEARNING TASK

### Training Details

In this section, we provide details about the observables learning task. While the quantum neural network contains  $N_{\text{sys}}$  qubits, the Hilbert space dimension is  $\mathcal{D} = 2^{N_{\text{sys}}}$ . Each input wave function  $|\psi\rangle$  can be expanded as:

$$|\psi\rangle = \frac{1}{\mathcal{N}} \sum_s^{\mathcal{D}} (a_s + ib_s) |s\rangle, \quad (7)$$

where  $\{|s\rangle\}$  represents the bases of this Hilbert space,  $\{a_s, b_s\}$  are randomly sampled from the uniform distribution  $[-1, 1]$ , and  $\mathcal{N}$  is the normalization factor. The target physical observables are diagonal in these bases with eigenvalues randomly sampled from the uniform distribution  $[-2.5, 2.5]$ . To avoid overfitting, the size of the training set  $N_{\mathcal{D}}$  must be larger than the degree of freedom of  $\mathcal{D}$ -dimensional hermitian matrices, i.e.  $N_{\mathcal{D}} > \mathcal{D}^2 = 2^{2N_{\text{sys}}}$ . The output of the randomized neural network is  $\mathcal{P}_m = \langle \psi_m | \sum_{i=1}^{N_r} p_i \hat{U}_{\text{tot},i}^\dagger (\beta_0 \hat{I} + \beta_1 \hat{\sigma}_z) \hat{U}_{\text{tot},i} | \psi_m \rangle$ . We use Mean Square Error (MSE) as the loss function, denoted as  $\mathcal{L} = \frac{1}{N_{\mathcal{D}}} \sum_m (\mathcal{P}_m - \mathcal{T}_m)^2$ . To escape from the local minima, we use mini-batch Adam method to update parameters and set the learning rate  $\eta = 0.01$ . Table [1] shows the training dataset size for different quantum system size.

We also apply the optimized neural network to the testing dataset containing  $N_{\text{test}} = 200$  samples. Fig.[2] shows the logarithmical mean square error of test dataset for different system size.

### Opening the black box

Based on the training results presented in the main text, it was found that for the two-qubit system, by setting  $N_r = 3$ , the loss can be significantly reduced as low as  $10^{-15}$ . We further investigate the effectiveness of the quantum neural network



TABLE I: Hyper parameters of QNN

$N_{\text{sys}}$	$\mathcal{D}$	$L_2$	$N_D$	$N_{\text{batch}}$	Epoch <sub>max</sub>
2	4	1	100	20	1000
3	8	4	200	40	1500
4	16	6	500	100	2000
5	32	8	1500	300	3000

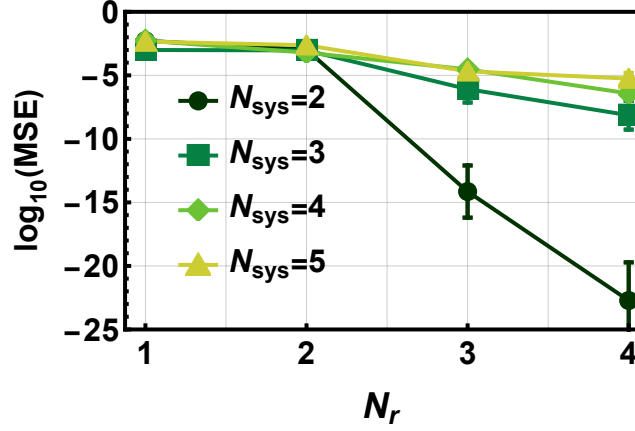


FIG. 2: Logarithmical mean square error of test dataset with 200 samples for  $N_{\text{sys}} = \{2, 3, 4, 5\}$  and  $N_r = \{1, 2, 3, 4\}$ . Markers are the averaged loss of 10 different training processes and error bars are the standard deviation.

in learning observable expectations. As shown in Fig.[2] in the main text, when  $N_{\text{sys}} = 2$ ,  $N_r = 3$ , we can predict the observable's expectation with extremely high accuracy. After training, we find that the probabilities of each random unitary are almost same  $w_i = 1/N_r$ . Consequently, the final prediction of the neural network is  $\mathcal{P} = \frac{1}{3} \sum_{i=1}^{N_r=3} \mathcal{P}_i$  where  $\mathcal{P}_i = \langle \psi | \hat{U}_{\text{tot},i}^\dagger (\beta_0 \hat{\sigma}_0 + \beta_1 \hat{\sigma}_z) \hat{U}_{\text{tot},i} | \psi \rangle$ . This results in the predicted operator reading as:

$$\hat{O}_{\text{pre}} = \frac{1}{3} \sum_{i=1}^{N_r=3} \hat{o}_i, \quad (8)$$

with  $\hat{o}_i = \hat{U}_{\text{tot},i}^\dagger (\beta_0 \hat{\sigma}_0 + \beta_1 \hat{\sigma}_z) \hat{U}_{\text{tot},i}$ . Noticing that the target observable is diagonal, it can be found that, after training, each element of  $\hat{O}_{\text{pre}}$  is very close to that of  $\hat{O}$ .

We expand each predicted operator  $\hat{o}_i$  in Pauli string operator bases  $\hat{o}_i = \sum_{ab} C_{ab}^i \hat{\sigma}_a \otimes \hat{\sigma}_b$  where  $a, b = x, y, z, 0$ .  $C_{00}$  represents the trace of  $\hat{O}$  and  $C_{zz}, C_{z0}, C_{0z}, C_{00}$  can reconstruct the diagonal elements of  $\hat{O}_{\text{pre}}$ . The random unitaries are tensor products of unitaries applied to each single-qubit, i.e.  $\hat{U}_{r,i} = \hat{u}_1^i \otimes \hat{u}_2^i$ . For the sake of simplicity,  $\hat{U}_{r,1}$  can be absorbed into  $\hat{U}_2$ . We then found that  $\hat{U}_{r,2}$  and  $\hat{U}_{r,3}$  commute with the Pauli operator  $\hat{\sigma}_z \otimes \hat{\sigma}_0$  and  $\hat{\sigma}_0 \otimes \hat{\sigma}_z$ . This implies that  $\hat{U}_{r,2}$  and  $\hat{U}_{r,3}$  rotate each qubit in its own Bloch sphere along  $z$  axis. So from the coefficients obtained through expansion, we can obtain the angles projected in the  $x - y$  plane:

$$\xi_{1,z} = \arctan(C_{zy}/C_{zx}), \quad \xi_{1,0} = \arctan(C_{0y}/C_{0x}), \quad (9)$$

$$\xi_{2,z} = \arctan(C_{yz}/C_{xz}), \quad \xi_{2,0} = \arctan(C_{y0}/C_{x0}). \quad (10)$$

For  $N_r = 3$ , the angle difference of each qubit between different  $\hat{o}_i$  is  $2\pi/3$ . This implies that  $\xi_{l,a}^1 - \xi_{l,a}^2 = \xi_{l,a}^2 - \xi_{l,a}^3 =$

$\xi_{l,a}^3 - \xi_{l,a}^1 = 2\pi/3$  where  $l = \{1, 2\}$ ,  $a = \{z, 0\}$ . Furthermore, table[II] illustrates certain relations between coefficients, ensuring the summation of off-diagonal elements of  $\hat{o}_i$  vanishes.

TABLE II: Relations between coefficients of predicted operators

a \ b	x	y	z	0
x	$C_{xx}$	$C_{xy}$	$C_{xz}$	$C_{xz}$
y	$-C_{xy}$	$C_{xx}$	$C_{yz}$	$C_{yz}$
z	$C_{zx}$	$C_{zy}$	$C_{zz}$	$C_{z0}$
0	$C_{zx}$	$C_{zy}$	$C_{0z}$	$C_{00}$

## TRAINING DETAILS OF RÉNYI ENTROPIES LEARNING TASK

### Training Details

In this section, we'll describe the data generation process for the Rényi entropies learning task and the optimization method for parameters. Similar to the observable learning task, we consider a quantum system containing  $N_{\text{sys}} = 5$  qubits. The Hilbert space dimension is  $\mathcal{D} = 2^5$  and each input wave function is generated as given by eq. (7). After the evolution, we measure the subsystem in the bases  $x_s = |\langle s | \hat{U}_{r,i} | \psi \rangle|^2$ ,  $s = 1, 2, \dots, 2^{N_{\text{sub}}}$ . When learning the purity,  $\mathcal{T} = \text{Tr}[\hat{\rho}_{\text{sub}}^2]$  is a quadratic function of reduced density matrices. Therefore, in this case we set the classical non-linear function to also be a quadratic function of measurement results  $\mathbf{x}$ .

$$f_{\beta}(\vec{x}) = \beta_0 + \vec{\beta}_1^T \vec{x} + \vec{x}^T \beta_2 \vec{x} \quad (11)$$

We also use mini-batch Adams to optimize parameters in these randomized quantum neural networks. Specifically, we set  $N_D = 100$ ,  $N_{\text{batch}} = 20$  for  $N_{\text{sub}} = 1$  and  $N_D = 200$ ,  $N_{\text{batch}} = 40$  for  $N_{\text{sub}} = 2$ . The learning rate is set to  $\eta = 0.01$ . After training, we apply this neural network to a test dataset consisting of  $N_{\text{test}} = 200$  samples to ensure that the neural network has learnt the purities.

### Convergence of Randomness for n=2 Rényi Entropy

In the main text, we demonstrated that for  $N_{\text{sub}} = 1$ , we need  $N_r = 3$  random unitary operators, and for  $N_{\text{sub}} = 2$ , we require  $N_r = 9$  random unitary operators to predict  $n = 2$  Rényi entropy with high accuracy. Here, for  $N_{\text{sub}} = 2$ , we further support our conclusion by comparing the learning performance of  $N_r = 9$  and  $N_r = 10$ . Both loss functions decay rapidly with exponential behavior and converged to nearly the same value, as shown in Fig. 3(a). After training, we extracted the probabilities  $\{w_i\}$  of each random unitary operator. They are all approximately equal to  $1/9$  for  $N_r = 9$ , and one probability would vanish for  $N_r = 10$ . This implies that for  $N_{\text{sub}} = 2$ ,  $N_r = 3^2$  is enough for achieving good performance.

### Third Order Entropy Learning

We also applied the randomized quantum neural networks to learn higher-order Rényi entropies, specifically,  $\mathcal{T} = \text{Tr}[\hat{\rho}_{\text{sub}}^3]$ . In this case, the label is a third order polynomial function of the density matrix. As a result, we set the classical non-linear function up to be a third-order polynomial of measurement results:

$$f_{\beta}(\vec{x}) = \beta_0 + \vec{\beta}_1^T \vec{x} + \vec{\beta}_2^T \vec{x}^{\otimes 2} + \vec{\beta}_3^T \vec{x}^{\otimes 3} \quad (12)$$

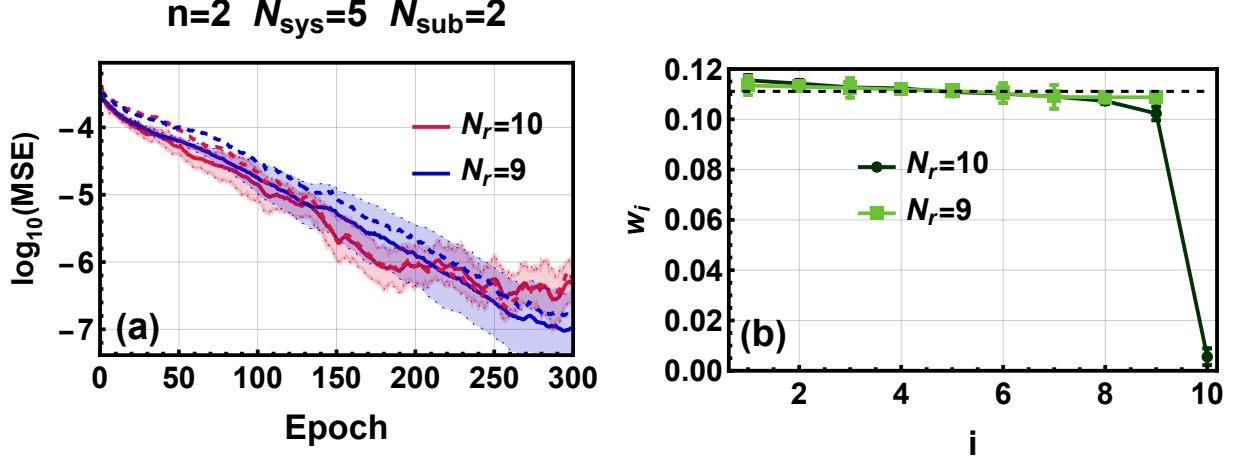


FIG. 3: The results for learning the second order  $\text{Tr}[\hat{\rho}_{\text{sub}}^2]$  for  $N_{\text{sub}} = 2$ . (a) The logarithmical mean square error for  $N_r = 9$  and  $N_r = 10$  is averaged over the training process for 10 different random initializations, with the shaded region representing the standard deviation. The dashed lines represent the validation loss using a dataset containing 200 samples. (b) Probabilities of random unitary operators for  $N_r = 9$  and  $N_r = 10$  in decreasing order. These optimal parameters  $\{w_i\}$  are averaged over the training process for 10 different random initializations with error bars included.

However, after training, we observed that the third-order coefficients  $\vec{\beta}_3$  vanish for the subsystem with one qubit. The loss for  $N_r = 3$  also decreases exponentially which is consistent with the analytical calculation for a general single-qubit density matrix:

$$\hat{\rho}_{\text{sub}} = \begin{pmatrix} \rho_{11} & \rho_{12}^r + i\rho_{12}^i \\ \rho_{12}^r - i\rho_{12}^i & 1 - \rho_{11} \end{pmatrix}$$

$$\text{Tr}[\hat{\rho}_{\text{sub}}^3] = 1 - 3\rho_{11} + 3\rho_{11}^2 + 3|\rho_{12}|^2 \quad (13)$$

where  $\rho_{12} = \rho_{12}^r + i\rho_{12}^i$ , and  $\hat{\rho}_{\text{sub}}$  is semi-positive with unit trace. The single-qubit Rényi entropies with  $n = 2, 3$  have a similar form.  $\text{Tr}[\hat{\rho}_{\text{sub}}^2] = 1 - 2\rho_{11} + 2\rho_{11}^2 + 2|\rho_{12}|^2$ . Therefore,  $N_r = 3$  is also suitable for  $n = 3$  Rényi entropy. However, for  $N_{\text{sub}} = 2$ ,  $\vec{\beta}_3$  makes a great contribution. As shown in Fig. 3(d) in the main text, increasing  $N_r$  leads to the loss converging to a lower value.

## TRAINING DETAILS OF PATTERN RECOGNITION TASK

### Training Details

In this section, we'll provide details on how we encode the "Street View of House Number (SVHN)" data into the 5-qubit quantum system, explain the explicit form of the nonlinear function, and describe the optimization methods used.

Each pattern in the original SVHN dataset contains a grid of  $32 \times 32$  pixels, with each pixel having three RGB channels. Initially, we convert these RGB images to grayscale using a weighted combination:  $0.2989R + 0.5870G + 0.1140B$ . Next, we resize the  $32 \times 32$  pixel images to  $8 \times 8$  pixels by averaging the nearest 4-by-4 neighborhood. Then, we reshape this resulting  $8 \times 8$  matrix into a  $32 \times 2$  matrix, with the first column representing the real part of the coefficients of the basis and the second column representing the imaginary part of the coefficients in eq.(7). For demonstration, we selected two patterns from all the images. We labeled one pattern as '1' and the other as '0'. After measuring the single qubit, we set the non-linear function as a polynomial of the measurement output  $x$ :

$$f_{\beta}(x) = \sum_{k=0}^5 \beta_k x^k \quad (14)$$

And for this classification task, we take cross-entropy as the loss function:

$$\mathcal{L} = \frac{1}{N_D} \sum_m^{N_D} -\mathcal{T}_m \log(\mathcal{G}_m) - (1 - \mathcal{T}_m) \log(1 - \mathcal{G}_m) \quad (15)$$

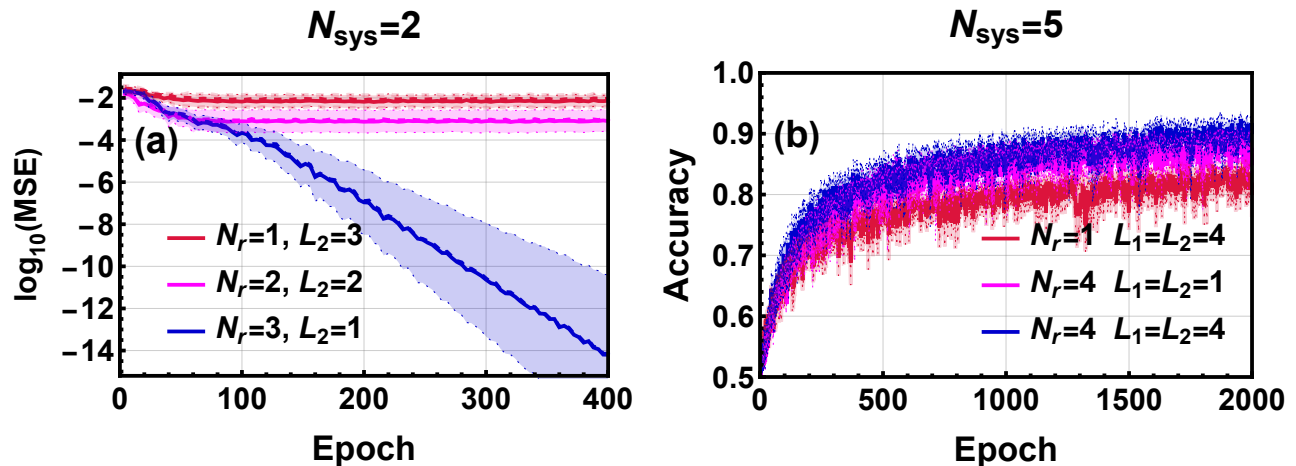


FIG. 4: Training results comparison with the same computational cost. (a) The observable learning result for  $N_{\text{sys}} = 2$  with save  $N_r L_2$ . (b) The patten recognition result for  $N_{\text{sys}} = 5$ . The red line and the pink line contain same computational cost  $N_r(L_1 + L_2) = 8$ . The training results are averaged over the training process for 10 different random initializations, and the shaded region represents the standard deviation. The dashed lines are the validation loss with the dataset containing 200 samples.

where  $\mathcal{G}_m = 1/1 + \exp(-\mathcal{P}_m)$  in the region  $(0,1)$ . We also use mini-batch Adams to optimize parameters in this randomized quantum neural network with  $N_D = 1200, L_1 = L_2 = 4, N_{\text{batch}} = 300, \eta = 0.01$ .

### COMPARISON WITH FIXED COMPUTATIONAL COST

In this section, we discuss the computational cost of the QNN with a random layer. In our approach, we introduce one random layer into the QNN, and the final prediction is based on the ensemble average of this random layer. The computational cost of the QNN scales linearly with both  $N_r$  and the depth of the deterministic layers  $L_1, L_2$ . Therefore, to make a fair comparison, we consider the results of different  $N_r$  with the same  $N_r(L_1 + L_2)$ .

For the observable learning task we set the deterministic layer  $\hat{U}_1$  as the identity, i.e. In this context, using  $L_1 = 0, N_r = 3, L_2 = 1$  is sufficient to predict the observables' expectations with high accuracy for a quantum system with  $N_{\text{sys}} = 2$ . In our investigation, we set  $L_2 = 3, 2$  for  $N_r = 1, 2$  while keeping  $N_r L_2$  approximately same. FIG. 4(a) illustrates that, for the same computational cost, introducing appropriate randomness can enhance the expressivity of the QNN. This enhancement occurs without the necessity of increasing the depth of the QNN. Even when considering  $N_r = 2, L_2 = 2$ , the computational cost is higher than when using  $N_r = 3, L_2 = 1$ .

Similar results are observed for the pattern recognition task. We set the deterministic layers  $\hat{U}_1, \hat{U}_2$  as variational with the same circuit depth  $L_1 = L_2$ . Pattern recognition is a typical machine learning task, where the both data and the function are classical. Thus FIG.4(b) shows that, even with just one brick wall unit in the deterministic, the configuration with  $N_r = 4$  also has better learning performance than that of  $N_r = 1, L_1 = L_2 = 4$ .

\* Electronic address: [pengfeizhang.physics@gmail.com](mailto:pengfeizhang.physics@gmail.com)

† Electronic address: [xiaopeng.li@fudan.edu.cn](mailto:xiaopeng.li@fudan.edu.cn)

[1] Yadong Wu, Pengfei Zhang, and Hui Zhai, "Scrambling ability of quantum neural network architectures", Phys. Rev. Res. 3:L032057 (2021).