

Supplementary Material for “Efficient Learning for Linear Properties of Bounded-Gate Quantum Circuits”

Yuxuan Du,^{1,*} Min-Hsiu Hsieh,^{2,†} and Dacheng Tao^{1,‡}

¹*College of Computing and Data Science, Nanyang Technological University, Singapore 639798, Singapore*

²*Hon Hai (Foxconn) Research Institute, Taipei, Taiwan*

CONTENTS

A. Preliminary and literature review	2
1. Classical shadow and its application in estimating incoherent dynamics of quantum states	2
2. Pauli transfer matrix and Trigonometric expansion of RZ + CI quantum circuits	3
3. Generality of exploring the learnability of $\mathcal{F}$	4
4. Literature review	5
B. Proof of Theorem 1	6
C. Lower bound of learning bounded-gate quantum circuits with incoherent measurement (Proof of Theorem B.1)	7
D. Upper bound of the sample complexity when incoherently learning the dynamic with bounded-gates (Proof of Theorem B.2)	13
E. Exponential separation between computational complexity and sample complexity (Proof of Theorem B.3)	15
F. Learnability of the proposed kernel-based ML model (Proof of Theorem 2)	16
1. Truncation error of the classical learning model (Proof of Lemma F.1)	18
2. Estimation error of the classical learning model (Proof of Lemma F.2)	21
G. Computational time for training and prediction	23
H. Classical prediction model with the full expansion (Proof of Corollary 1)	24
1. Implementation of the classical prediction model	24
2. Proof of Corollary 1	25
I. Learning bounded-gate quantum circuit with CI gates and parameterized multi-qubit gates	26
J. Connection and differences with Ref. [1] and Pauli path simulators	27
1. Complementation with Ref. [1]	27
a. Insight I: Computational complexity analysis	28
b. Insight II: Learning algorithm design	29
2. Connections and differences with Pauli path simulators	30
K. More numerical simulations	31
1. Heuristic evidence about the truncation value Λ	31
2. Numerical simulations of N -qubit rotational GHZ states	33
3. More details of synthetic global Hamiltonian simulation	34
4. More simulation results of enhanced variational quantum algorithms by pretraining	35
a. Pretraining a 3-qubit HEA for both VQE and QNN	37
b. Pretraining Hamiltonian-variational ansatz towards a class of Transverse-field Ising model	39
c. Pretraining Hamiltonian-variational ansatz towards a class of H_5 molecules	45

* duyuxuan123@gmail.com

† minhsiuh@gmail.com

‡ dacheng.tao@gmail.com

SM A: Preliminary and literature review

This section contains the necessary backgrounds of the whole work. Specifically, in Supplementary Material (SM) A 1, we present the basic information of classical shadow. Then, in SM A 2, we recap the trigonometric expansion of quantum circuits implemented by RZ and Clifford gates. Afterward, in SM A 3, we elaborate on the generality of the explored problem in our work. Last, we provide a literature review in SM A 4.

1. Classical shadow and its application in estimating incoherent dynamics of quantum states

Classical shadow represents a computationally and memory-efficient approach for storing quantum states on classical computers, primarily used for estimating the expectation values of local observables [2]. The fundamental principle of classical shadow lies in the ‘measure first and ask questions later’ strategy. In this subsection, we outline the utilization of classical shadow to estimate linear functions under Pauli-based measurements. Interested readers can refer to tutorials and surveys [3, 4] for more comprehensive details.

Formalism of classical shadow. The general scheme of classical shadow for an unknown N -qubit state ρ is repeating the following procedure T times. At each time, the state ρ is first operated with a unitary U randomly sampled from the predefined unitary ensemble $\mathcal{U}$ and then each qubit is measured under the Z basis to obtain an N -bit string denoted by $\mathbf{b} \in \{0, 1\}^N$. There exists a linear map $\mathcal{M}(\cdot)$ satisfying

$$\mathcal{M}(\rho) = \mathbb{E}_{U \sim \mathcal{U}} \mathbb{E}_{\mathbf{b} \sim \mathbb{P}(\mathbf{b})} U^\dagger |\mathbf{b}\rangle \langle \mathbf{b}| U = \sum_{\mathbf{b}} \int dU U^\dagger |\mathbf{b}\rangle \langle \mathbf{b}| U \langle \mathbf{b}| U \rho U^\dagger |\mathbf{b}\rangle, \quad (\text{A1})$$

where $\mathbb{P}(\mathbf{b}) = \langle \mathbf{b}| U \rho U^\dagger |\mathbf{b}\rangle$. Such a linear map implies that the unknown state ρ can be formulated as

$$\rho = \sum_{\mathbf{b}} \int dU \mathcal{M}^{-1} \left(U^\dagger |\mathbf{b}\rangle \langle \mathbf{b}| U \right) \langle \mathbf{b}| U \rho U^\dagger |\mathbf{b}\rangle. \quad (\text{A2})$$

In other words, the state ρ can be estimated by sampling the snapshot with M times following the distribution $\mathbb{P}(\mathbf{b})$. Define the t -th snapshot as $U_t^\dagger |\mathbf{b}_t\rangle \langle \mathbf{b}_t| U_t$ with $t \in [T]$ and $U_t \sim \mathcal{U}$. The estimated state of ρ is

$$\hat{\rho}_T = \frac{1}{T} \sum_{t=1}^T \hat{\rho}_t, \text{ with } \hat{\rho}_t = \mathcal{M}^{-1}(U_t^\dagger |\mathbf{b}_t\rangle \langle \mathbf{b}_t| U_t). \quad (\text{A3})$$

As pointed out in Ref. [2], the random Pauli basis is not only experimentally friendly but also enables a succinct form of the classical shadow. When Pauli-based measurements are adopted, it is equivalent to setting the unitary ensemble $\mathcal{U}$ as single-qubit Clifford gates, i.e., $U_t = U_{1,t} \otimes \cdots \otimes U_{j,t} \cdots \otimes U_{N,t} \sim \mathcal{U} = \text{CI}(2)^{\otimes N}$ with uniform weights. In this case, the inverse snapshot takes the form as

$$\hat{\rho}_t = \mathcal{M}^{-1} \left(\bigotimes_{j=1}^N U_{j,t}^\dagger |\mathbf{b}_{j,t}\rangle \langle \mathbf{b}_{j,t}| U_{j,t} \right) = \bigotimes_{j=1}^N \mathcal{D}_{1/3}^{-1} \left(U_{j,t}^\dagger |\mathbf{b}_{j,t}\rangle \langle \mathbf{b}_{j,t}| U_{j,t} \right) = \bigotimes_{j=1}^N \left(3U_{j,t}^\dagger |\mathbf{b}_{j,t}\rangle \langle \mathbf{b}_{j,t}| U_{j,t} - \mathbb{I}_2 \right), \quad (\text{A4})$$

where $\mathcal{D}_{1/3}^{-1}(Y) = 3Y - \text{Tr}(Y)\mathbb{I}$.

Estimate incoherent dynamics (linear properties of quantum states). The tensor product form of the classical shadow in Eq. (A4) allows an efficient procedure to predict linear properties of the state ρ . A typical instance is estimating the expectation value $\text{Tr}(\rho O)$ with O being a local observable. Mathematically, suppose the local observable to be a Pauli string, i.e., $O = P_1 \otimes \cdots \otimes P_N$ with $P_i \in \{X, Y, Z, \mathbb{I}\}$ for $\forall i \in [N]$, the estimation of the classical shadow is

$$\text{Tr}(\hat{\rho}_T(P_1 \otimes \cdots \otimes P_N)) = \frac{1}{T} \sum_{t=1}^T \prod_{j=1}^N \text{Tr} \left(\left(3U_{j,t}^\dagger |\mathbf{b}_{j,t}\rangle \langle \mathbf{b}_{j,t}| U_{j,t} - \mathbb{I}_2 \right) P_j \right), \quad (\text{A5})$$

which is memory and computation efficient. Namely, one only needs $\mathcal{O}(NT)$ bits to store $\hat{\rho}_T$ and $\mathcal{O}(NT)$ computational time to load $\hat{\rho}_T$ to the classical memory. Next, when the locality of the observable $O = \sum_i O_i$ is $K \sim \mathcal{O}(1)$, the shadow estimation of the expectation value, i.e., $\text{Tr}(\hat{\rho} O_i)$, can be performed in $\mathcal{O}(T)$ time after $\hat{\rho}_T$ is loaded into the classical memory [2].

2. Pauli transfer matrix and Trigonometric expansion of RZ + CI quantum circuits

Pauli Transfer Matrix. Here we use Pauli-Liouville representation to reformulate the quantum state and the observable. Specifically, an N -qubit state can be represented as a 4^N -dimensional vector, whose i -th entry is

$$|\cdot\rangle\rangle = \text{Tr}(\cdot P_i) \text{ with } P_i \in \{\mathbb{I}, X, Y, Z\}^N. \quad (\text{A6})$$

For example, the state $|0\rangle^{\otimes N}$ satisfies $(|0\rangle\langle 0|)^{\otimes N} = ((\mathbb{I} + Z)/2)^{\otimes N}$, which indicates that its representation under the Pauli basis yields

$$|0\rangle^{\otimes N} \equiv |\mathbf{0}\rangle\rangle = ((\mathbb{I} + Z)/2)^{\otimes N} = ([1, 0, 0, 1]^\top)^{\otimes N}. \quad (\text{A7})$$

Similarly, the normalized Pauli operator O under the Pauli basis yields

$$|O\rangle\rangle = [\text{Tr}(OP_1), \dots, \text{Tr}(OP_{4^N})]^\top. \quad (\text{A8})$$

The unitary operator can also be represented by Pauli basis, i.e., given a parameterized circuit $U(\boldsymbol{\theta})$, its Pauli Transfer Matrix (PTM) $\mathfrak{U}(\boldsymbol{\theta})$ yields

$$[\mathfrak{U}(\boldsymbol{\theta})]_{ij} = \langle\langle P_i | \mathfrak{U}(\boldsymbol{\theta}) | P_j \rangle\rangle = \text{Tr}(P_i U(\boldsymbol{\theta}) P_j U(\boldsymbol{\theta})^\dagger). \quad (\text{A9})$$

For example, the PTM representation of RZ($\mathbf{x}_j$) gate is

$$\mathfrak{R}_Z(\mathbf{x}_j) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\mathbf{x}_j) & \sin(\mathbf{x}_j) & 0 \\ 0 & -\sin(\mathbf{x}_j) & \cos(\mathbf{x}_j) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = D_0 + \cos(\mathbf{x}_j)D_1 + \sin(\mathbf{x}_j)D_{-1}, \quad (\text{A10})$$

$$\text{where } D_0 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, D_1 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \text{ and } D_{-1} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

Trigonometric expansion of RZ + CI quantum circuits. A critical research line in quantum computing involves determining if variational quantum algorithms can provide meaningful advantages over state-of-the-art classical methods through dequantization via Fourier expansion [5, 6]. Despite differing objectives, the techniques developed, particularly the Low-Weight Efficient Simulation Algorithm (LOWESA) [7, 8], serve as inspiration for our approach to predicting the incoherent dynamics of bounded-gate quantum circuits. In the following context, we delve into the mechanics of LOWESA and postpone the discussion on the connection and difference between our work and this research line to SM A 4.

We now recap the mechanism of LOWESA. When an N -qubit quantum circuit $U(\mathbf{x})$ is composed of d RZ gates and $G - d$ CI gates, the state representation under Pauli-basis expansion yields

$$\rho(\mathbf{x}) = U(\mathbf{x}) (|0\rangle\langle 0|)^{\otimes N} U(\mathbf{x})^\dagger = \sum_{\boldsymbol{\omega}} \Phi_{\boldsymbol{\omega}}(\mathbf{x}) \langle\langle 0 | \mathfrak{U}_{\boldsymbol{\omega}}^\dagger \equiv \sum_{\boldsymbol{\omega}} \Phi_{\boldsymbol{\omega}}(\mathbf{x}) \rho_{\boldsymbol{\omega}}. \quad (\text{A11})$$

The notation $\Phi_{\boldsymbol{\omega}}(\mathbf{x})$ with $\boldsymbol{\omega} \in \{0, \pm 1\}^d$ refers to the *trigonometric monomial basis* with values

$$\Phi_{\boldsymbol{\omega}}(\mathbf{x}) = \prod_{i=1}^d \begin{cases} 1 & \text{if } \omega_i = 0 \\ \cos(\mathbf{x}_i) & \text{if } \omega_i = 1 \\ \sin(\mathbf{x}_i) & \text{if } \omega_i = -1 \end{cases}. \quad (\text{A12})$$

Moreover, $\mathfrak{U}_{\boldsymbol{\omega}}$ in Eq. (A11) refers to the *purely-Clifford circuit* for the path indexed by $\boldsymbol{\omega}$ in the sense that in the path $\boldsymbol{\omega}$, each RZ gate at the i -th position with $i \in [d]$ is replaced by the operator D_{ω_i} in Eq. (A10). The quantum mean value under an observable O is

$$f(\mathbf{x}, O) \equiv \text{Tr}(\rho(\mathbf{x})O) = \sum_{\boldsymbol{\omega}} \Phi_{\boldsymbol{\omega}}(\mathbf{x}) \langle\langle 0 | \mathfrak{U}_{\boldsymbol{\omega}}^\dagger | O \rangle\rangle. \quad (\text{A13})$$

It is noteworthy that LOWESA is a purely classical approach to estimating expectation values of simple variational quantum circuits. The term ‘simple’ is reflected by the fact that even for low dimensional classical inputs (i.e., a small d), its computational complexity is prohibited by other two key factors: the initial state and the number of CI gates $G - d$. This is because when the initial state $(|0\rangle\langle 0|)^{\otimes N}$ is substitute a complicated one (e.g., a state with exponentially many terms under Pauli basis), and $G - d$ becomes large, the computational overhead of calculating $\langle\langle 0 | \mathfrak{U}_{\boldsymbol{\omega}}^\dagger | O \rangle\rangle$ is unaffordable by classical simulators.

3. Generality of exploring the learnability of $\mathcal{F}$

Here we discuss the generality of the explored mean-value space in the main text, i.e.,

$$\mathcal{F} = \left\{ f(\mathbf{x}, O) = \text{Tr}(U(\mathbf{x})\rho_0 U(\mathbf{x})^\dagger O) \mid \mathbf{x} \in [-\pi, \pi]^d \right\}, \quad (\text{A14})$$

where $U(\mathbf{x}) = \prod_{l=1}^d (\text{RZ}(\mathbf{x}_l)u_e)$ is composed of d $\text{RZ}(\mathbf{x}_l)$ gates, $G-d$ CI gates denoted by U_e with $\text{CI} = \{H, S, \text{CNOT}\}$, O is constituted by multiple local observables with a bounded norm, i.e., $O = \sum_{i=1}^q O_i$ and $\sum_l \|O_i\|_\infty \leq B$, and the maximum locality of $\{O_i\}$ is K . This formalism encompasses many applications for quantum computing, including variational quantum algorithms, classical-shadow-based algorithms, and quantum system certification. A common feature of these applications is that the exploited quantum circuit can be hardware-efficient. As a result, for a specified quantum device, its executable quantum circuits can be described by $\text{RZ} + \text{CI}$ gates with a fixed layout but with different angles, i.e., $U(\mathbf{x})$ in Eq. (A14). Given an unknown state ρ_0 evolved by $U(\mathbf{x})$ and incoherently measured by an observable O , it forms the mean-value space $\mathcal{F}$ explored in this work. In the following, we detail how $\mathcal{F}$ relates to variational quantum algorithms, classical-shadow-based applications, and quantum system certification, respectively.

Variational quantum algorithms. We briefly review the mechanism of variational quantum algorithms (VQAs). Interested readers can refer to the surveys [9, 10] for detailed information. VQAs generally consist of an N -qubit circuit and a classical optimizer. In the training stage, VQAs follow an iterative manner to proceed with optimization, where the optimizer continuously leverages the output of the quantum circuit to update trainable parameters of the adopted ansatz, i.e., $\mathbf{x}$ of $U(\mathbf{x})$, to minimize a predefined objective function $\mathcal{L}(\cdot)$. Mathematically, at the t -th iteration, the updating rule for the trainable parameters $\mathbf{x}$ is

$$\mathbf{x}^{(t+1)} = \mathbf{x}^{(t)} - \eta \frac{\partial \mathcal{L}(f(\mathbf{x}^{(t)}, O), c_1)}{\partial \mathbf{x}}, \quad (\text{A15})$$

where η is the learning rate, $c_1 \in \mathbb{R}$ is the target label, and $f(\mathbf{x}^{(t)}, O)$ amounts to the output of the quantum circuit defined in Eq. (A14). The optimization is terminated when the training loss is lower than a threshold or the total number of iterations T exceeds a predefined value [11–13]. Two main protocols of VQAs are quantum neural networks (QNNs) and variational quantum Eigen-solver (VQEs). The former is utilized to solve machine learning tasks such as classification and regression, and the latter is exploited to estimate the ground state energy of a given Hamiltonian. Notably, when QNNs are applied, the classical input $\mathbf{x}$ should be divided into two parts, where the first part is used to encode training examples (without updating) and the second part refers to the trainable parameters. Refer to SM K for details.

Classical-shadow-based applications. A major application of classical shadow, as introduced in SM A1, is estimating linear properties of quantum states, i.e., $\text{Tr}(\rho O)$. When the measured states ρ are generated by large-qubit quantum devices, whose circuit can be described by $\text{RZ} + \text{CI}$ gates with a fixed layout but with different angles, the formed function space coincides with $\mathcal{F}$ in Eq. (A14). It is noteworthy that classical-shadow-based applications are highly entangled with variational quantum algorithms [14, 15], quantum system certification (explained below), and two novel quantum machine learning protocols—flipped model and shadow model [16]. For the flipped model, the training examples $\mathbf{z}$ are encoded into the measurement observables rather than the quantum state, i.e., the prediction of the flipped model yields $\text{Tr}(U(\mathbf{x})\rho_0 U(\mathbf{x})^\dagger O(\mathbf{z}))$, where $\mathbf{x}$ refers to trainable parameters. As the mean-value space $\mathcal{F}$ in Eq. (A14) supports a class of measurement operators O sampled from a distribution $\mathbb{D}_O$, the achieved results in this work can be harnessed to empower the flipped model. Similarly, the key concept of the shadow model is using classical shadow to obtain the classical description of the state $U(\mathbf{x})\rho_0 U(\mathbf{x})^\dagger$, followed by classically estimating $\text{Tr}(U(\mathbf{x})\rho_0 U(\mathbf{x})^\dagger O(\mathbf{z}))$, which is also under the framework of $\mathcal{F}$ in Eq. (A14). Taken together, the developed algorithm in this work can greatly reduce the required quantum resources for developing the relevant algorithms.

Quantum system certification. As quantum devices scale up to larger system sizes, the demand for application-specific certification tools becomes apparent. These tools must surpass standard approaches, such as fully simulating a device on a classical computer or performing full tomographic reconstruction, which incurs exponential computational overhead with the qubit count. In response, various certification protocols have been developed to extract different levels of information from the explored quantum chip by using minimal quantum resources. Among them, a representative class of certification protocols is estimating the linear property of quantum states generated by the employed quantum chip, such as direct fidelity estimation [17], entanglement witnessing [18], and two-point correlator [19]. Notably, all of these tasks can be described by the mean-value space $\mathcal{F}$ in Eq. (A14).

4. Literature review

Relevant prior literature to our study can be categorized into three groups: solving quantum many-body problems using machine learning, learnability of quantum circuits, and the combination of quantum computing (e.g., variational quantum algorithms and quantum system certification) and classical machine learning. In the following, we elucidate how our study aligns with and distinguishes itself from these earlier works.

Machine learning for quantum many-body problems. The seminal work in this context is presented in Huang et al. [19], demonstrating that machine learning algorithms, informed by data collected in physical experiments, can effectively tackle certain quantum many-body problems that are challenging for classical algorithms. In particular, the l_2 Dirichlet kernel is proposed to predict the ground state properties of a class of Hamiltonians.

Recent follow-up works have further augmented the capacity of machine learning in addressing quantum many-body problems. Specifically, Ref. [20] explored a specific class of scenarios in quantum many-body problems and devised *positive good kernels*, achieving a polynomial sample complexity for predicting quantum many-body states and their properties. Furthermore, when concentrating on learning the average of observables with a locality assumption, Refs. [21, 22] achieved a quasi-polynomial sample complexity. Additionally, Ref. [23] extended the results of learning phases of quantum matter characterized by exponential decay of correlations, to the task of learning local expectation values for all states within a phase.

Our study diverges from this line of research in terms of its distinct objectives. While previous works concentrate on predicting the properties of quantum many-body states, our focus is on predicting the incoherent dynamics of bounded-gate quantum circuits. This disparity in objectives results in the utilization of *different prior information*, with the former relying primarily on assumptions about the explored Hamiltonian, such as locality, while the latter leverages the properties of quantum gates, such as RZ + CI. As a result, the proposed trigonometric monomial kernel in our proposal is more succinct compared to the l_2 Dirichlet kernel in terms of the frequency set, i.e., $\omega \in \{0, \pm 1\}^d$ in Eq. (6) versus $\mathbf{k} \in \mathbb{Z}^d$ in Dirichlet kernel. Furthermore, the learning efficiency of the algorithm proposed in Ref. [23] relies on *the smoothness of the Hamiltonians* in the quantum many-body systems under exploration, an assumption that does not apply to the learnability of bounded-gate quantum circuits. A key technical contribution of this work is establishing a connection between the effectiveness of the proposed ML model and the gradients of bounded-gate quantum circuits with respect to the classical inputs.

Learning quantum circuits. Several studies have been undertaken to explore the learnability of quantum circuits, a critical aspect of quantum learning theory [24, 25]. Notably, in Ref. [1], it was demonstrated that when learning a state generated by a quantum circuit with G two-qubit gates to a small trace distance, a sample complexity scaling linearly in G is both necessary and sufficient. However, the computational complexity for learning states and unitaries must scale exponentially in G . While our work shares similar complexity scaling, it diverges in two key aspects: (1) *we address a different problem*, focusing on predicting linear properties of bounded-gate quantum circuits with a *classical learner*, while Ref. [1] investigated the quantum learner such that quantum resources are allowed in both the training and inference time; and (2) we provide concrete algorithms to balance the trade-off between sample and computational complexities.

In Ref. [26], a polynomial-time classical algorithm was devised to learn the description of any unknown N -qubit shallow quantum circuit U . Furthermore, they developed another polynomial-time classical algorithm to learn the description of any unknown N -qubit state prepared by a shallow quantum circuit U on a 2D lattice. However, unlike their study, our proposed algorithm does not necessitate the circuit to be shallow. Additionally, the key components of our algorithm include classical shadow, kernel method, and trigonometric expansion, whereas their algorithm relies on a quantum circuit representation based on local inversions and a technique to combine these inversions.

In Ref. [27], the authors examined the power and limitations of learning unitary dynamics using quantum neural networks through both coherent and incoherent measurements. Similar to Ref. [1], the learner in this study is still quantum, meaning that quantum resources are required during inference. In contrast, our work focuses on a different problem where the learner is classical, indicating that the techniques employed to achieve the final results are fundamentally different.

Combination between quantum computing and machine learning. For clarity, here we separately discuss how previous studies harness machine learning to enhance variational quantum algorithms and quantum system certification.

Variational quantum algorithms. The incorporation of machine learning to improve VQAs follows two main approaches. The first approach involves integrating deep neural networks and variational quantum circuits to form a hybrid learning model capable of addressing diverse computational tasks [28–32]. These learning protocols often lack theoretical guarantees and fall outside the scope of our study.

The second approach entails designing *classical surrogates* capable of inferring the output of VQAs. The outcomes of this approach not only benefit from assessing whether VQAs can offer meaningful advantages over state-of-the-art

classical methods but also contribute to conserving quantum resources for the development of novel VQAs. A notable paradigm in this context, complementary to tensor network methods and Clifford-based simulators, involves the dequantization of VQAs via Fourier expansion [6]. In particular, Refs. [7, 8, 33, 34] proposed Fourier-based algorithms to simulate variational quantum algorithms when applied to simulate the expectation value of an observable for an initial state evolved under a unitary quantum circuit, e.g., the tasks covered by variational quantum Eigen-solvers and quantum approximate optimization algorithms. Besides, Refs. [35–38] harness Fourier features to dequantize VQAs, i.e., quantum neural networks, when applied to solve classical machine learning tasks.

Our approach distinguishes itself from previous Fourier-based methods by offering broader applicability beyond the classical simulation of VQAs. A specific illustration is that prior Fourier-based approaches often impose constraints on the initial state, such as requiring it to be a simple product state $|0\rangle\langle 0|^{\otimes N}$ interpreted in SM A 2. In contrast, our proposal eliminates this requirement. This advancement is rooted in the hybrid nature of our approach, which utilizes shadow information from quantum computers to build the training dataset. Technically, our method integrates three distinct techniques—classical shadow, kernel method, and trigonometric expansion—paving the way for the development of novel strategies to enhance variational quantum algorithms with provable guarantees.

Quantum system certification. We next explain how our work relates to learning-based quantum system certification. Prior literature related to this topic can be classified into two categories. The first category is using deep learning to improve quantum tomography [39], which has two distinct research lines. The first line involves explicit state reconstruction, where the output of neural networks represents the density matrix of the target quantum state [40–42]. This line exploits the generalization ability of neural networks, where the optimized neural networks can predict the density matrix of unseen states when sampled from the same distribution of the training data. The second line focuses on implicit state reconstruction, wherein neural networks are optimized to emulate the behavior of a given quantum state [43–46]. Note that the second line merely exploits the fitting power of neural networks, where the optimized neural networks can only reconstruct a single state and do not possess the generalization ability. As such, this research line is beyond the scope of our work. The second category is using deep learning to predict the properties of quantum states, including fidelity estimation [47], energy estimation [48], entropy estimation [49], cross-platform verification [50], and similarity testing [51].

Despite distinct applications, the learning paradigm of these two categories is very similar. In particular, a labeled dataset needs to be first collected to conduct supervised learning. The data features of training examples generally are random measurement results, and the label corresponds to the specific tasks, i.e., the density matrix for quantum state tomography and entanglement entropy for entropy estimation. After training, the optimized neural network can predict the unseen state by feeding into the random measurement results. As addressed in the main text, a critical caveat of learning-based quantum system certification is that it lacks a theoretical guarantee. Our work fills this knowledge gap and provides concrete evidence of using various machine learning techniques to comprehend quantum systems.

Remark. We conclude this section by highlighting the complementary nature of our work to quantum tomography and classical simulators in comprehending the behavior of large-qubit quantum devices. In contrast to tomography-based approaches, the offline capability of our proposal could markedly reduce the quantum resource overhead. Moreover, compared to classical simulators, our proposal offers two notable advantages. Firstly, it outperforms classical simulation methods in terms of runtime complexity by removing the reliance on the number of Clifford gates. Secondly, our approach accommodates arbitrarily complex input states, whereas classical simulators often necessitate simple input states such as the product state. A summary of the key focuses between different works is shown in the table below.

Supplementary Table 1: **Summary of the key focuses between different works.**

	Learner Type	Access Quantum Resources at Inference	Learning Model
Huang et al. [19]	Classical	No	Dirichlet Kernel
Zhao et al. [1]	Quantum	Yes	-
Jerbi et al. [27]	Quantum	Yes	Quantum Neural Networks
Schreiber et al. [35]	Classical	No	Least square regression for Fourier-based models
This work	Classical	No	Trigonometric Monomial Kernel

SM B: Proof of Theorem 1

Theorem (Formal statement of Theorem 1). *Following notations in Eq. (3), denote a dataset $\{\mathbf{x}^{(i)}, \tilde{f}_T(\mathbf{x}^{(i)}, O)\}_{i=1}^n$ containing n training examples with $\mathbf{x}^{(i)} \in [-\pi, \pi]^d$ and $\tilde{f}_T(\mathbf{x}^{(i)}, O)$ being the estimation of $f(\mathbf{x}^{(i)}, O)$ using T inco-*

herent measurements. Then, when $T \leq \frac{(1-\epsilon^2)}{\epsilon} N \log 2$ and $\epsilon \ll 1$, the training data size

$$\frac{(1-\epsilon)(C_1 d - \log 2)}{\epsilon T} \leq n \leq \tilde{O}\left(\frac{B^2 d + B^2 N G}{\epsilon}\right) \quad (\text{B1})$$

is sufficient and necessary to achieve $\mathbb{E}_{\mathbf{x} \sim \mathbb{D}_{\mathcal{X}}} |h(\mathbf{x}, O) - \text{Tr}(\rho(\mathbf{x}), O)| \leq \epsilon$ with high probability. However, unless $\text{BQP} \subseteq \text{P/poly}$, there exists a class of G -bounded-gate quantum circuits that no algorithm can achieve $\mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} |h(\mathbf{x}, O) - \text{Tr}(\rho(\mathbf{x}), O)| \leq \epsilon$ in a polynomial time.

The proof of Theorem 1 can be broken down into three parts: the lower bound of sample complexity, the upper bound of sample complexity, and the lower bound of runtime complexity. The corresponding bounds are established in the following three theorems, and the proofs are presented in SM C, SM D, and SM E, respectively.

Theorem B.1 (Lower bound of sample complexity). *Consider that a learner collects a dataset $\mathcal{T} = \{\mathbf{x}^{(i)}, \tilde{\rho}_T(\mathbf{x}^{(i)})\}_{i=1}^n$ containing n training examples to predict unseen states $\rho(\mathbf{x})$ with $\mathbf{x} \in [-\pi, \pi]^d$. Suppose that with high probability, the learned model $h(\cdot, \cdot)$ can achieve*

$$\mathbb{E}_{\mathbf{x} \sim \mathbb{D}_{\mathcal{X}}} |h(\mathbf{x}, O) - \text{Tr}(\rho(\mathbf{x}), O)| \leq \epsilon. \quad (\text{B2})$$

Then, when $T \leq \frac{(1-\epsilon)}{\epsilon} N \log 2$, the training data size must obey

$$n \geq \frac{(1-\epsilon)(C_1 d - \log 2)}{\epsilon T}, \quad (\text{B3})$$

where $C_1 \in (0, 1)$ is a constant.

Theorem B.2 (Upper bound of sample complexity). *Consider a learner quires an N -qubit quantum circuits and collects a dataset $\mathcal{T} = \{\mathbf{x}^{(i)}, \tilde{\rho}_T(\mathbf{x}^{(i)})\}_{i=1}^n$ containing n training examples to predict unseen states $\rho(\mathbf{x})$ with $\mathbf{x} \in [-\pi, \pi]^d$. Suppose with high probability, there exists a learned model $h(\cdot, \cdot)$ that can achieve*

$$\mathbb{E}_{\mathbf{x} \sim \mathbb{D}_{\mathcal{X}}} |h(\mathbf{x}, O) - \text{Tr}(\rho(\mathbf{x}), O)|^2 \leq \epsilon. \quad (\text{B4})$$

using training data of size

$$n \leq \tilde{O}\left(\frac{B^2 d + B^2 N G}{\epsilon}\right). \quad (\text{B5})$$

Theorem B.3 (Lower bound of runtime complexity). *There exists a class of bounded-gate quantum circuits composed of d RZ gates and $G - d$ CI gates such that no polynomial-time algorithm exists to achieve*

$$\mathbb{E}_{\mathbf{x} \sim \mathbb{D}_{\mathcal{X}}} |h(\mathbf{x}, O) - \text{Tr}(\rho(\mathbf{x}), O)|^2 \leq \epsilon, \quad (\text{B6})$$

unless $\text{BQP} \subseteq \text{P/poly}$.

Proof of Theorem 1. The proof of Theorem 1 can be immediately obtained by combining the results of Theorems B.1, B.2, and B.3. $\square$

SM C: Lower bound of learning bounded-gate quantum circuits with incoherent measurement (Proof of Theorem B.1)

In this section, we comprehend the fundamental limitation of learning quantum circuits with a bounded number of non-Clifford gates when only the classical input control and incoherent measurements are allowed, and the circuit layout is unknown to the learner. As elaborated on in the main text, these restrictions echo most experiments for large-qubit quantum computing, i.e., the prediction solely relies on shadow information about the output state without harnessing any prior knowledge about the circuit layout. In addition, as the data distribution $\mathbb{D}_{\mathcal{X}}$ is arbitrary, here we consider a specific setting in which the classical inputs $\mathbf{x}$ are sampled from the uniform distribution, i.e., $\mathbf{x} \sim [-\pi, \pi]^d$.

Theorem (Restatement of Theorem B.1). *Consider that a learner collects a dataset $\mathcal{T} = \{\mathbf{x}^{(i)}, \tilde{\rho}_T(\mathbf{x}^{(i)})\}_{i=1}^n$ containing n training examples to predict unseen states $\rho(\mathbf{x})$ with $\mathbf{x} \in [-\pi, \pi]^d$. Suppose that with high probability, the learned model $h(\cdot, \cdot)$ can achieve*

$$\mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} |h(\mathbf{x}, O) - \text{Tr}(\rho(\mathbf{x}), O)| \leq \epsilon. \quad (\text{C1})$$

Then, when $T \leq \frac{(1-\epsilon)}{\epsilon} N \log 2$, the training data size must obey

$$n \geq \frac{(1-\epsilon)(C_1 d - \log 2)}{\epsilon T}, \quad (\text{C2})$$

where $C_1 \in (0, 1)$ is a constant.

To reach Theorem B.1, we constitute a class of quantum circuits and analyze the lower bound of the required sample complexity for a learner to achieve the average prediction error ϵ . In particular, we consider a simple family of two-qubit quantum circuits illustrated in Fig. 1(a), which is composed of d RZ gates and at most $6d$ CNOT gates, as each SWAP gate can be decomposed into three CNOT gates. The explicit expression of this class of quantum circuits is

$$\mathcal{F} = \{f_{\mathbf{a}}(\mathbf{x}) = \text{Tr}(W(\mathbf{x}, \mathbf{a})(\rho_0 \otimes |0\rangle\langle 0|)W(\mathbf{x}, \mathbf{a})^\dagger O) \mid \mathbf{a} \in \{0, 1\}^d, \mathbf{x} \sim [-\pi, \pi]^d\}, \quad (\text{C3})$$

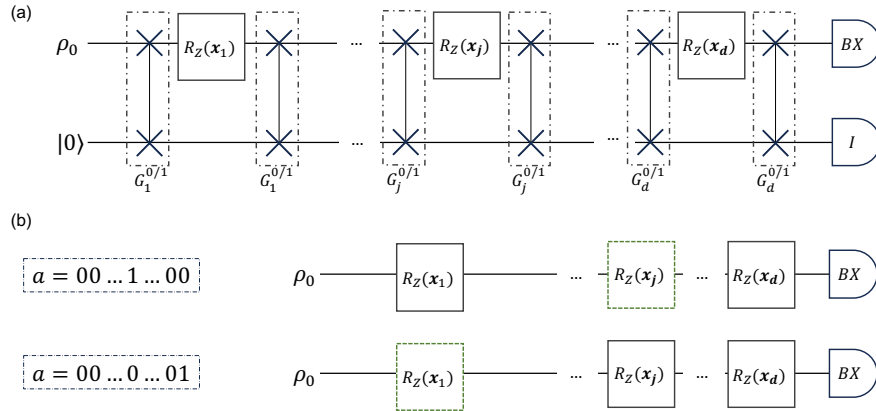
where $\rho_0 \otimes |0\rangle\langle 0|$ is the initial two-qubit state with

$$\rho_0 = \begin{bmatrix} \left(\left(\sqrt{1-2\sqrt{\epsilon}} + \sqrt{1+2\sqrt{\epsilon}} \right) / 2 \right)^2 & \sqrt{2\epsilon}/2 \\ \sqrt{2\epsilon}/2 & \left(\left(\sqrt{1-2\sqrt{\epsilon}} - \sqrt{1+2\sqrt{\epsilon}} \right) / 2 \right)^2 \end{bmatrix}, \quad (\text{C4})$$

$W(\mathbf{x}, \mathbf{a})$ refers to the parameterized unitary depending on the classical input $\mathbf{x}$ and the label $\mathbf{a}$ with

$$W(\mathbf{x}, \mathbf{a}) = G_d^{\mathbf{a}_d}(\text{RZ}(\mathbf{x}_d) \otimes \mathbb{I})G_d^{\mathbf{a}_d} \cdots G_j^{\mathbf{a}_j}(\text{RZ}(\mathbf{x}_j) \otimes \mathbb{I})G_j^{\mathbf{a}_j} \cdots G_1^{\mathbf{a}_1}(\text{RZ}(\mathbf{x}_1) \otimes \mathbb{I})G_1^{\mathbf{a}_1} \quad (\text{C5})$$

and $G_0^0 = \cdots = G_d^0 = \mathbb{I}_4$ and $G_1^1 = \cdots = G_d^1 = \text{SWAP}$, and $O = B(X \otimes \mathbb{I}_2)$ is the observable.



Supplementary Fig. 1: **Some intuitions about the proof of Theorem B.1.** (a) ILLUSTRATION OF THE CONSTRUCTED FAMILY OF QUANTUM CIRCUITS. The dashed box refers that $\forall i \in [d]$, the SWAP gate G_i is optionally applied to the circuit if the bit string satisfies $\mathbf{a}_i = 1$. (b) THE VISUALIZATION OF THE SIMPLIFIED CIRCUIT WHEN $\mathbf{a} = 00\dots 1\dots 00$ (UPPER) AND $\mathbf{a} = 00\dots 0\dots 01$ (LOWER). The dashed green box stands for replacing the RZ gate with the identity gate.

Note that any candidate in $\mathcal{F}$ can be simplified to a single-qubit circuit, as demonstrated in Fig. 1(b). That is, the equivalent single-qubit circuit only consists of a sequence of $\text{RZ}(\mathbf{x}_j)$ gates whose index $\mathbf{a}_j$ is zero applied to the initial state ρ_0 , followed by the measurement operator BX . Thus, the simplified expression of this family of circuits is

$$\mathcal{F} = \{f_{\mathbf{a}}(\mathbf{x}) = B \text{Tr}(V^{\mathbf{a}_d}(\mathbf{x}_d) \cdots V^{\mathbf{a}_1}(\mathbf{x}_1) \rho_0 V^{\mathbf{a}_1}(\mathbf{x}_1)^\dagger \cdots V^{\mathbf{a}_d}(\mathbf{x}_d)^\dagger X) \mid \mathbf{a} \in \{0, 1\}^d, \mathbf{x} \sim [-\pi, \pi]^d\}, \quad (\text{C6})$$

where $V^0(\mathbf{x}_j) = \text{RZ}(\mathbf{x}_j)$ and $V^1(\mathbf{x}_j) = \mathbb{I}_2$ for $\forall j \in [d]$. It is evident that the cardinality of $\mathcal{F}$ is $|\mathcal{F}| = 2^d$.

To proceed further analysis, we now quantify the average distance of different functions in $\mathcal{F}$. To do so, we reformulate $f_{\mathbf{a}}(\mathbf{x}) \in \mathcal{F}$ based on the trigonometric expansion. According to PTM representations in Eq. (A9), the initial state ρ_0 and Pauli- X can be written as

$$|\rho_0\rangle\rangle = [1, \sqrt{2\epsilon}, 0, 0]^\top \quad \text{and} \quad |X\rangle\rangle = [0, 1, 0, 0]^\top, \quad \text{respectively.} \quad (\text{C7})$$

Moreover, following the PTM representations of RZ in Eq. (A10), every function in $\mathcal{F}$ satisfies

$$f_{\mathbf{a}}(\mathbf{x}) = B \left\langle \left\langle \epsilon \middle| \Re_Z^\dagger \left(\sum_{j, \mathbf{a}_j=0} \mathbf{x}_j \right) \middle| X \right\rangle \right\rangle \quad (\text{C8a})$$

$$= \sqrt{2\epsilon} B \cos \left(\sum_{j, \mathbf{a}_j=0} \mathbf{x}_j \right) \quad (\text{C8b})$$

$$= \sqrt{2\epsilon} B \cos((1 - \mathbf{a}_1)\mathbf{x}_1 + (1 - \mathbf{a}_2)\mathbf{x}_2 + \dots + (1 - \mathbf{a}_d)\mathbf{x}_d). \quad (\text{C8c})$$

The derived explicit form allows us to quantify the average discrepancy of any two candidates in $\mathcal{F}$, i.e., given $\forall f_{\mathbf{a}}, f_{\mathbf{a}'} \in \mathcal{F}$ with $\mathbf{a} \neq \mathbf{a}'$, we have

$$\mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} |f_{\mathbf{a}}(\mathbf{x}, O) - f_{\mathbf{a}'}(\mathbf{x}, O)|^2 \quad (\text{C9a})$$

$$= 2\epsilon B^2 \mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} \left| \cos((1 - \mathbf{a}_1)\mathbf{x}_1 + \dots + (1 - \mathbf{a}_d)\mathbf{x}_d) - \cos((1 - \mathbf{a}'_1)\mathbf{x}_1 + \dots + (1 - \mathbf{a}'_d)\mathbf{x}_d) \right|^2 \quad (\text{C9b})$$

$$= 2\epsilon B^2 \left(\frac{1}{2} + \frac{1}{2} \right) - 2\epsilon B^2 \mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} \cos((1 - \mathbf{a}_1)\mathbf{x}_1 + \dots + (1 - \mathbf{a}_d)\mathbf{x}_d) \cos((1 - \mathbf{a}'_1)\mathbf{x}_1 + \dots + (1 - \mathbf{a}'_d)\mathbf{x}_d) \quad (\text{C9c})$$

$$= 2\epsilon B^2, \quad (\text{C9d})$$

where the first equality follows the explicit form of $f_{\mathbf{a}}$ and $f_{\mathbf{a}'}$ in Eq. (C8a), the second equality employs the two facts: (i) $\forall \mathbf{a} \in \{0, 1\}^d$, there are $d - \|\mathbf{a}\|_0$ effective and independent variables $\{\mathbf{x}_j\}$; (ii) these effective variables satisfy

$$\mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} |\cos((1 - \mathbf{a}_1)\mathbf{x}_1 + \dots + (1 - \mathbf{a}_d)\mathbf{x}_d)|^2 \quad (\text{C10a})$$

$$= \mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^{d - \|\mathbf{a}_0\|}} \cos^2(\mathbf{x}_1 + \mathbf{x}_2 + \dots + \mathbf{x}_{d - \|\mathbf{a}_0\|}) \quad (\text{C10b})$$

$$= \frac{1}{(2\pi)^{d - \|\mathbf{a}_0\|}} \int \frac{1 + \cos(2\mathbf{x}_1 + 2\mathbf{x}_2 + \dots + 2\mathbf{x}_{d - \|\mathbf{a}_0\|})}{2} d\mathbf{x} \quad (\text{C10c})$$

$$= \frac{1}{2} + \frac{1}{(2\pi)^{d - \|\mathbf{a}_0\|}} \int \frac{\cos(2\mathbf{x}_1) \cos(2\mathbf{x}_2 + \dots + 2\mathbf{x}_{d - \|\mathbf{a}_0\|}) - \sin(2\mathbf{x}_1) \sin(2\mathbf{x}_2 + \dots + 2\mathbf{x}_{d - \|\mathbf{a}_0\|})}{2} d\mathbf{x} \quad (\text{C10d})$$

$$= \frac{1}{2} + \frac{1}{(2\pi)^{d - \|\mathbf{a}_0\|}} \int_{-\pi}^{\pi} \cos(2\mathbf{x}_1) d\mathbf{x}_1 \int \frac{\cos(2\mathbf{x}_2 + \dots + 2\mathbf{x}_{d - \|\mathbf{a}_0\|})}{2} d\mathbf{x} \\ - \frac{1}{(2\pi)^{d - \|\mathbf{a}_0\|}} \int_{-\pi}^{\pi} \sin(2\mathbf{x}_1) d\mathbf{x}_1 \int \frac{\sin(2\mathbf{x}_2 + \dots + 2\mathbf{x}_{d - \|\mathbf{a}_0\|})}{2} d\mathbf{x} \quad (\text{C10e})$$

$$= \frac{1}{2}, \quad (\text{C10f})$$

and the last equality exploits the fact that the second term in Eq. (C9c) is zero as explained below. Specifically, we partition $\mathbf{x}_1, \dots, \mathbf{x}_d$ into two groups, depending on the index list $\mathbf{a}$ and $\mathbf{a}'$. Denote $R = \sum_{i, \mathbf{a}_i = \mathbf{a}'_i = 0} \mathbf{x}_i$ as the summation of entries whose index is zero in both $\mathbf{a}$ and $\mathbf{a}'$, and $\bar{R} = (1 - \mathbf{a}_1)\mathbf{x}_1 + \dots + (1 - \mathbf{a}_d)\mathbf{x}_d - R$ and $\bar{R}' = (1 - \mathbf{a}'_1)\mathbf{x}_1 + \dots + (1 - \mathbf{a}'_d)\mathbf{x}_d - R$ as the summation of the rest effective variables with the index list $\mathbf{a}$ and $\mathbf{a}'$, respectively. Without loss of generality, we suppose $\|\mathbf{a}\|_0 > \|\mathbf{a}'\|_0$, implying that R contains at least one effective variable. Then, the second term in Eq. (C9c) follows

$$\mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} \cos((1 - \mathbf{a}_1)\mathbf{x}_1 + \dots + (1 - \mathbf{a}_d)\mathbf{x}_d) \cos((1 - \mathbf{a}'_1)\mathbf{x}_1 + \dots + (1 - \mathbf{a}'_d)\mathbf{x}_d) \quad (\text{C11a})$$

$$= \mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} \cos(R + \bar{R}) \cos(R + \bar{R}') \quad (\text{C11b})$$

$$= \mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} [\cos(\bar{R}) \cos(R) - \sin(\bar{R}) \sin(R)] [\cos(R) \cos(\bar{R}') - \sin(R) \sin(\bar{R}')] \quad (\text{C11c})$$

$$= \mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} [\cos(\bar{R}) \cos^2(R) \cos(\bar{R}') - \cos(\bar{R}) \cos(R) \sin(R) \sin(\bar{R}') \\ - \sin(\bar{R}) \sin(R) \cos(R) \cos(\bar{R}') + \sin(\bar{R}) \sin^2(R) \sin(\bar{R}')] \quad (\text{C11d})$$

$$= 0, \quad (\text{C11e})$$

where the last equality exploits the fact that when R contains at least one effective variable, the symmetric property of integral with respect to trigonometric functions gives $\mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} \cos(R) = \mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} \sin(R) = 0$.

The result in Eq. (C9d) indicates that taking the expectation over $\mathbf{x}$, all concept functions in $\mathcal{F}$ are equally distant by $2\epsilon B^2$. Learning this family of circuits can be reduced to the following *multiple hypothesis testing problem*:

1. Alice randomly and uniformly chooses a target concept $f_{\mathbf{a}^*} \in \mathcal{F}$, or equivalently $\mathbf{a}^* \in \{0, 1\}^d$;
2. The training dataset $\mathcal{T} = \{(\mathbf{x}^{(i)}, \hat{\mathbf{o}}_{\mathbf{a}^*}^{(i)})\}_{i=1}^n$ with the size n is collected based on $f_{\mathbf{a}^*}$, i.e., for the i -th training example, the classical input $\mathbf{x}^{(i)}$ randomly sampled from $[-\pi, \pi]^d$ is fed into the quantum system described by $f_{\mathbf{a}^*}$ and the statistical estimation of $f_{\mathbf{a}^*}(\mathbf{x}^{(i)}, O)$ is obtained by T shots with $\hat{\mathbf{o}}_{\mathbf{a}^*}^{(i)} = \frac{1}{T} \sum_t \hat{\mathbf{o}}_{\mathbf{a}^*, t}^{(i)}$ and $\mathbb{E}(\hat{\mathbf{o}}_{\mathbf{a}^*}^{(i)}) = f_{\mathbf{a}^*}(\mathbf{x}^{(i)}, O)$;
3. The learner leverages the training dataset $\mathcal{T}$ to conduct the empirical risk minimization, i.e.,

$$h_{\mathcal{T}} = \arg \min_{h \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^n \ell \left(h(\mathbf{x}^{(i)}) - \hat{\mathbf{o}}_{\mathbf{a}^*}^{(i)} \right), \quad (\text{C12})$$

where ℓ refers to the loss function measuring the difference between the prediction $h(\mathbf{x}^{(i)})$ and the label $\hat{\mathbf{o}}_{\mathbf{a}^*}^{(i)}$;

4. The hypothesis testing is conducted to infer the target concept using the learned $h_{\mathcal{T}}$, i.e., the inferred index is

$$\bar{\mathbf{a}} = \arg \min_{\mathbf{a}, f_{\mathbf{a}} \in \mathcal{F}} \mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} |h_{\mathcal{T}}(\mathbf{x}) - f_{\mathbf{a}}(\mathbf{x})|^2 \quad (\text{C13})$$

and the associated error probability is $\mathbb{P}(\bar{\mathbf{a}} \neq \mathbf{a}^*)$.

Since the average discrepancy between any two concepts in $\mathcal{F}$ is $2\epsilon B^2$ as indicated in Eq. (C9d), the error probability $\mathbb{P}(\bar{\mathbf{a}} \neq \mathbf{a}^*)$ becomes zero when $\mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} |h_{\mathcal{T}}(\mathbf{x}) - f_{\mathbf{a}^*}(\mathbf{x}, O)|^2 < \epsilon B^2$. In this scenario, $h_{\mathcal{T}}$ has a large average prediction error for the data sampled from other $f_{\mathbf{a}'} \in \mathcal{F} \setminus \{f_{\mathbf{a}^*}\}$ in which the average error is at least ϵB^2 , i.e.,

$$\mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} |h_{\mathcal{T}}(\mathbf{x}) - f_{\mathbf{a}'}(\mathbf{x}, O)|^2 \geq \mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} |f_{\mathbf{a}'}(\mathbf{x}, O) - f_{\mathbf{a}^*}(\mathbf{x}, O)|^2 - \mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} |h_{\mathcal{T}}(\mathbf{x}) - f_{\mathbf{a}^*}(\mathbf{x}, O)|^2 > 2\epsilon B^2 - \epsilon B^2 = \epsilon B^2. \quad (\text{C14})$$

The multiple hypothesis testing problem reformulated above enables us to use Fano's inequality to derive the lower bound of sample complexity in learning $f_{\mathbf{a}^*} \in \mathcal{F}$. To be concrete, Fano's lemma [52] states that

$$\mathbb{P}[\mathbf{a}^* \neq \bar{\mathbf{a}}] \geq 1 - \frac{I(\mathbf{a}^*; \bar{\mathbf{a}}) + \log 2}{\log |\mathcal{F}|}, \quad (\text{C15})$$

where $I(\mathbf{a}^*; \bar{\mathbf{a}})$ refers to the mutual information between random variables $\mathbf{a}^*$ and $\bar{\mathbf{a}}$, and $|\mathcal{F}|$ denotes the cardinality of $\mathcal{F}$. In other words, the derivation of the lower bound of the sample complexity n amounts to quantifying the upper bound of $I(\mathbf{a}^*; \bar{\mathbf{a}})$ and the lower bound of $|\mathcal{F}|$, which motivates the following proof of Theorem B.1.

Remark. For ease of analysis, in the proof of Theorem B.1, we consider the shot number T is *sufficiently large* such that the measured results can be approximated by the normal distribution with the mean $\mu_{\mathbf{a}} = f_{\mathbf{a}}(\mathbf{x})$. This assumption is widely utilized in quantum computation and quantum information theory [53–56]. Besides, for each $\mathbf{a}$, the corresponding variance of measured results is assumed to be equal with the varied $\mathbf{x}$, i.e., for all $\mathbf{x}$, the variance is $\nu_{\mathbf{a}} = \mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} \nu_{\mathbf{a}}(\mathbf{x})$. Note that the assumption can be omitted if the primary focus is solely on the number of RZ gates d , without considering the measurement cost T . In this case, the upper bound of the mutual information $I(\mathbf{a}^*; \bar{\mathbf{a}})$ can be effectively derived using Holevo's theorem [57, 58], where a similar lower bound of the sample complexity can still be obtained with a linear dependence on d .

Proof of Theorem B.1. According to the above elaboration, we next separately quantify $|\mathcal{F}|$ and the mutual information $I(\mathbf{a}^*; \bar{\mathbf{a}})$, followed by Eq. (C15) to attain the lower bound of sample complexity in learning the incoherent dynamics of bounded-gate quantum circuits.

Cardinality of $\mathcal{F}$. The cardinality of $\mathcal{F}$ can be obtained following its definition in Eq. (C3), i.e.,

$$|\mathcal{F}| = 2^d. \quad (\text{C16})$$

Upper bound of mutual information $I(\mathbf{a}^*; \bar{\mathbf{a}})$. Recall that the process of learning $f_{\mathbf{a}^*}$ implies the Markov chain:

$$\mathbf{a}^* \rightarrow \rho_{1:n} \rightarrow \hat{\mathbf{o}}_{1:n} \rightarrow \bar{\mathbf{a}}, \quad (\text{C17})$$

where for ease of notation, $\rho_{1:n}$ refers to the abbreviation of n resulting states $\rho_{\mathbf{a}^*}(\mathbf{x}^{(1)}), \dots, \rho_{\mathbf{a}^*}(\mathbf{x}^{(n)})$ before taking the measurements, and $\hat{\mathbf{o}}_{1:n}$ refers to the abbreviation of the measured statistic results $\hat{\mathbf{o}}_{\mathbf{a}^*}^{(1)}, \dots, \hat{\mathbf{o}}_{\mathbf{a}^*}^{(n)}$ of n training examples. Then, according to the data processing inequality, the mutual information $I(\mathbf{a}^*; \bar{\mathbf{a}})$ is upper bounded by the mutual information between the target index $\mathbf{a}^*$ and the measured statistical results $\hat{\mathbf{o}}_{1:n}$, i.e.,

$$I(\mathbf{a}^*; \bar{\mathbf{a}}) \leq I(\mathbf{a}^*; \hat{\mathbf{o}}_{1:n}). \quad (\text{C18})$$

Moreover, the mutual information on the right-hand side is upper-bounded by

$$I(\mathbf{a}^*; \hat{\mathbf{o}}_{1:n}) \quad (\text{C19a})$$

$$\leq \sum_{i=1}^n I(\mathbf{a}^*; \hat{\mathbf{o}}_i) \quad (\text{C19b})$$

$$\leq \sum_{i=1}^n I(\mathbf{a}^*; \hat{\mathbf{o}}_i, \mathbf{x}^{(i)}) \quad (\text{C19c})$$

$$= \sum_{i=1}^n I(\mathbf{a}^*; \mathbf{x}^{(i)}) + I(\mathbf{a}^*; \hat{\mathbf{o}}_i | \mathbf{x}^{(i)}) \quad (\text{C19d})$$

$$= \sum_{i=1}^n \mathbb{E}_{\mathbf{x}^{(i)} \sim [-\pi, \pi]^d} I(\mathbf{a}^* | \mathbf{x}^{(i)}; \hat{\mathbf{o}}_i | \mathbf{x}^{(i)}) \quad (\text{C19e})$$

$$= \sum_{i=1}^n \mathbb{E}_{\mathbf{x}^{(i)} \sim [-\pi, \pi]^d} I(\mathbf{a}^*; \hat{\mathbf{o}}_i | \mathbf{x}^{(i)}). \quad (\text{C19f})$$

The first inequality stems from the chain rule and the fact that conditioning reduces entropy, i.e., $I(X; Y_{1:n}) = \sum_{i=1}^n H(Y_i | Y_{1:i-1}) - H(Y_i | X, Y_{1:i-1}) = \sum_{i=1}^n H(Y_i | Y_{1:i-1}) - H(Y_i | X) \leq \sum_{i=1}^n H(Y_i) - H(Y_i | X) = \sum_{i=1}^n I(X; Y_i)$, the second inequality exploits the relation $I(X; Y, Z) = H(X) - H(X | Y, Z) \geq H(X) - H(X | Y) = I(X; Y)$, the first equality employs the chain rule of mutual information with $I(X; Y, Z) = I(X; Y) + I(X; Z | Y)$, the second equality adopts the independence between $\mathbf{a}^*$ and $\mathbf{x}^{(i)}$ with $I(\mathbf{a}^*; \mathbf{x}^{(i)}) = 0$ and the KL divergence reformulation of mutual information, and the last equality comes from the independence between $\mathbf{a}^*$ and $\mathbf{x}^{(i)}$.

The above relation hints that the prerequisite to derive the upper bound of $I(\mathbf{a}^*; \hat{\mathbf{o}}_{1:n})$ is upper bounding $I(\mathbf{a}^*; \hat{\mathbf{o}}_i | \mathbf{x}^{(i)})$. As such, we apply KL divergence formulation to $I(\mathbf{a}^*; \hat{\mathbf{o}}_i | \mathbf{x}^{(i)})$ and obtain

$$I(\mathbf{a}^*; \hat{\mathbf{o}}_i | \mathbf{x}^{(i)}) = D_{\text{KL}}(P_{\mathbf{a}^*, \mathbf{o} | \mathbf{x}^{(i)}} \| P_{\mathbf{a}^*} P_{\mathbf{o} | \mathbf{x}^{(i)}}) \quad (\text{C20a})$$

$$= \sum_{\mathbf{a}^*} \int p(\mathbf{a}^*, \mathbf{o} | \mathbf{x}^{(i)}) \log \frac{p(\mathbf{a}^*, \mathbf{o} | \mathbf{x}^{(i)})}{p(\mathbf{a}^*) p(\mathbf{o} | \mathbf{x}^{(i)})} d\mathbf{o} \quad (\text{C20b})$$

$$= \sum_{\mathbf{a}^*} p(\mathbf{a}^*) \int p(\mathbf{o} | \mathbf{a}^*, \mathbf{x}^{(i)}) \log \frac{p(\mathbf{o} | \mathbf{a}^*, \mathbf{x}^{(i)})}{p(\mathbf{o} | \mathbf{x}^{(i)})} d\mathbf{o} \quad (\text{C20c})$$

$$= \frac{1}{|\mathcal{F}|} \sum_{\mathbf{a}^*} D_{\text{KL}}(P_{\mathbf{o} | \mathbf{a}^*, \mathbf{x}^{(i)}} \| P_{\mathbf{o} | \mathbf{x}^{(i)}}) \quad (\text{C20d})$$

$$\leq \frac{1}{|\mathcal{F}|^2} \sum_{\mathbf{a}^*, \mathbf{a}'} D_{\text{KL}}(P_{\mathbf{o} | \mathbf{a}^*, \mathbf{x}^{(i)}} \| P_{\mathbf{o} | \mathbf{a}', \mathbf{x}^{(i)}}), \quad (\text{C20e})$$

where the third equality uses $p(\mathbf{a}^*, \mathbf{o} | \mathbf{x}^{(i)}) = p(\mathbf{a}^*) p(\mathbf{o} | \mathbf{a}^*, \mathbf{x}^{(i)})$, the fourth equality comes from the fact that $\mathbf{a}^*$ is uniformly sampled with $p(\mathbf{a}^*) = 1/|\mathcal{F}|$, and the inequality employs the property of KL divergence.

Combining the above results, the mutual information is upper bounded by

$$I(\mathbf{a}^*; \bar{\mathbf{a}}) \leq \frac{1}{|\mathcal{F}|^2} \sum_{i=1}^n \mathbb{E}_{\mathbf{x}^{(i)} \sim [-\pi, \pi]^d} \sum_{\mathbf{a}^*, \mathbf{a}'} D_{\text{KL}}(P_{\mathbf{o} | \mathbf{a}^*, \mathbf{x}^{(i)}} \| P_{\mathbf{o} | \mathbf{a}', \mathbf{x}^{(i)}}). \quad (\text{C21})$$

According to our assumption, when the shot number T becomes large, the central limit theorem suggests that $P_{\mathbf{o} | \mathbf{a}^*, \mathbf{x}^{(i)}}$ follows the Gaussian distribution with the mean $\mu_{\mathbf{a}^*}(\mathbf{x}^{(i)})$ and the variance ν^2 . Moreover, assuming that the variance of the varied candidate is the same, we have

$$\mathbb{E}_{\mathbf{x}^{(i)} \sim [-\pi, \pi]^d} D_{\text{KL}}(P_{\mathbf{o} | \mathbf{a}^*, \mathbf{x}^{(i)}} \| P_{\mathbf{o} | \mathbf{a}', \mathbf{x}^{(i)}}) = \mathbb{E}_{\mathbf{x}^{(i)} \sim [-\pi, \pi]^d} \frac{(\mu_{\mathbf{a}^*}(\mathbf{x}^{(i)}) - \mu_{\mathbf{a}'}(\mathbf{x}^{(i)}))^2}{2\nu^2}. \quad (\text{C22})$$

For the nominator in Eq. (C22), the result of Eq. (C9d) gives

$$\mathbb{E}_{\mathbf{x}^{(i)} \sim [-\pi, \pi]^d} (\mu_{\mathbf{a}^*}(\mathbf{x}^{(i)}) - \mu_{\mathbf{a}'}(\mathbf{x}^{(i)}))^2 = \mathbb{E}_{\mathbf{x}^{(i)} \sim [-\pi, \pi]^d} (f_{\mathbf{a}^*}(\mathbf{x}^{(i)}) - f_{\mathbf{a}'}(\mathbf{x}^{(i)}))^2 = 2\epsilon B^2. \quad (\text{C23})$$

For the generic variance ν^2 , it is derived as follows. When the state $\rho_{\mathbf{a}}(\mathbf{x})$ is measured by $O = BX$, the probability of measuring the state associated with the eigenvalue $+1$ and the state associated with the eigenvalue -1 is

$$\mathbb{P}(+1) - \mathbb{P}(-1) = \frac{f_{\mathbf{a}}(\mathbf{x}^{(i)})}{B} \Leftrightarrow \mathbb{P}(+1) = \frac{1 + f_{\mathbf{a}}(\mathbf{x}^{(i)})/B}{2}, \quad \mathbb{P}(-1) = \frac{1 - f_{\mathbf{a}}(\mathbf{x}^{(i)})/B}{2}. \quad (\text{C24})$$

This property indicates that when the shot number is T , the explicit expression of the variance for the input $\mathbf{x}^{(i)}$ is

$$\nu_{\mathbf{a}}^2(\mathbf{x}^{(i)}) = \frac{1}{T} \left[\left(\mathbb{P}(+) * B^2 + \mathbb{P}(-) * (-B)^2 \right) - f_{\mathbf{a}}(\mathbf{x}^{(i)})^2 \right] = \frac{B^2}{T} - \frac{f_{\mathbf{a}}(\mathbf{x}^{(i)})^2}{T}. \quad (\text{C25})$$

Taking expectation over $\mathbf{x}$, we have

$$\nu = \mathbb{E}_{\mathbf{x}^{(i)} \sim [-\pi, \pi]^d} (\nu_{\mathbf{a}, \mathbf{x}^{(i)}}^2) = \frac{B^2}{T} - \frac{\mathbb{E}_{\mathbf{x}^{(i)} \sim [-\pi, \pi]^d} (f_{\mathbf{a}}(\mathbf{x}^{(i)})^2)}{T} = \frac{B^2 - B^2\epsilon}{T}, \quad (\text{C26})$$

where the last equality uses the result of Eq. (C10).

In conjunction with Eqs. (C21), (C22), (C23), and (C26), the mutual information is upper bounded by

$$I(\mathbf{a}^*; \bar{\mathbf{a}}) \leq \frac{n\epsilon T}{1 - \epsilon}. \quad (\text{C27})$$

Note that the mutual information cannot continuously enhance with respect to the increased shot number T . When $T \rightarrow \infty$, the mutual information is upper bounded by

$$\lim_{T \rightarrow \infty} I(\mathbf{a}^*; \hat{\mathbf{o}}_{1:n}) \leq I(\mathbf{a}^*; \rho_{1:n}). \quad (\text{C28})$$

Following the results [59, Lemma 9], the right-hand side term is upper bounded by the number of training examples and the qubit count, i.e.,

$$I(\mathbf{a}^*; \rho_{1:n}) \leq nN \log 2. \quad (\text{C29})$$

Taken together, we have

$$I(\mathbf{a}^*; \bar{\mathbf{a}}) \leq n \cdot \min \left\{ \frac{\epsilon T}{1 - \epsilon}, N \log 2 \right\}. \quad (\text{C30})$$

Reusing the Fano's inequality, we obtain

$$I(\mathbf{a}^*; \bar{\mathbf{a}}) \geq (1 - \mathbb{P}[\mathbf{a}^* \neq \bar{\mathbf{a}}]) \log |\mathcal{F}| - \log 2 \quad (\text{C31a})$$

$$\Rightarrow n \cdot \min \left\{ \frac{\epsilon T}{1 - \epsilon}, N \log 2 \right\} \geq (1 - \mathbb{P}[\mathbf{a}^* \neq \bar{\mathbf{a}}])d - \log 2 \quad (\text{C31b})$$

$$\Rightarrow n \geq \frac{(1 - \mathbb{P}[\mathbf{a}^* \neq \bar{\mathbf{a}}])d - \log 2}{\min \left\{ \frac{\epsilon T}{1 - \epsilon}, N \log 2 \right\}}. \quad (\text{C31c})$$

When $T \leq \frac{(1-\epsilon)}{\epsilon} N \log 2$, the lower bound of the sample complexity can be simplified to

$$n \geq (1 - \epsilon) \frac{(1 - \mathbb{P}[\mathbf{a}^* \neq \bar{\mathbf{a}}])d - \log 2}{\epsilon T}. \quad (\text{C32})$$

□

SM D: Upper bound of the sample complexity when incoherently learning the dynamic with bounded-gates (Proof of Theorem B.2)

In this section, we analyze the upper bound of the sample complexity when a learner can predict the output of quantum circuits with bounded gates within a tolerable error. A restatement of the corresponding theorem is as follows.

Theorem (Restatement of Theorem B.2). *Consider a learner quires an N -qubit quantum circuits and collects a dataset $\mathcal{T} = \{\mathbf{x}^{(i)}, \tilde{\rho}_T(\mathbf{x}^{(i)})\}_{i=1}^n$ containing n training examples to predict unseen states $\rho(\mathbf{x})$ with $\mathbf{x} \in [-\pi, \pi]^d$. Suppose with high probability, there exists a learned model $h(\cdot, \cdot)$ that can achieve*

$$\mathbb{E}_{\mathbf{x} \sim \mathbb{D}_{\mathcal{X}}} |h(\mathbf{x}, O) - \text{Tr}(\rho(\mathbf{x}), O)|^2 \leq \epsilon. \quad (\text{D1})$$

using training data of size

$$n \leq \tilde{\mathcal{O}} \left(\frac{B^2 d + B^2 N G}{\epsilon} \right). \quad (\text{D2})$$

Our proof is rooted in utilizing the packing net and packing number, which are advanced tools broadly used in statistical learning theory [60] and quantum learning theory [61–63], to quantify the complexity of the class of function represented by bounded-gate quantum circuits. For elucidating, in the following, we first introduce some basic concepts and results that will be employed in our proof, followed by presenting the proof of Theorem B.2. Besides, we emphasize that our current focus is solely on the sample complexity, which implies that the computational cost of preparing the training examples may exhibit exponential scaling with the qubit number as indicated in Theorem B.3.

We now elaborate on how to use packing nets to derive the upper bound of the sample complexity of a learner tasked with predicting the incoherent dynamics of bounded-gate quantum circuits. The formal definition of the packing net and packing number is given below.

Definition 1 (Packing net/number). *Let $(\mathcal{U}, \mathfrak{d})$ be a metric space. The subset $\mathcal{V} \subset \mathcal{U}$ is an ϵ -packing net of $\mathcal{U}$ if for any $A, B \in \mathcal{V}$, $\mathfrak{d}(A, B) \geq \epsilon$. The packing number $\mathcal{M}(\mathcal{U}, \epsilon, \mathfrak{d})$ denotes the largest cardinality of an ϵ -packing net of $\mathcal{U}$.*

Intuitively, an ϵ packing number refers to the maximum number of elements that can be ϵ -separated.

We next leverage this definition to prove Theorem B.2. Following notations introduced in the main text, given an observable O , the function space of quantum circuits with G gates is

$$\mathcal{F} = \left\{ f(\mathbf{x}, O) = \text{Tr}(\rho(\mathbf{x})O) \mid \mathbf{x} \in [-\pi, \pi]^d, \text{Arc}(\text{RZ}, \text{CI}) \right\}, \quad (\text{D3})$$

where $\mathbf{x} \in [-\pi, \pi]^d$ is the classical input control, $\text{Arc}(\text{RZ}, \text{CI})$ refers to the set of circuit layouts formed by d RZ gates and $G - d$ Clifford gates, and $\rho(\mathbf{x})$ is the quantum state generated by the bounded gates with $\mathbf{x}$, i.e., $\rho(\mathbf{x}) = U(\mathbf{x})\rho_0 U(\mathbf{x})^\dagger$ and the layout of $U(\mathbf{x})$ follows an element of $\text{Arc}(\text{RZ}, \text{CI})$. The diversity of circuit layouts stems from the constraint that the learner can solely utilize the classical input and the corresponding measurement results to infer the target concept, where prior information regarding the quantum circuit indicates that it comprises G gates, with d RZ gates and $d - G$ CI gates. From the perspective of the learner, the quantum computer can yield many circuit layouts, leading that the function space $\mathcal{F}$ contains in total $|\text{Arc}(\text{RZ}, \text{CI})|$ circuit layouts.

When the quantum computer is specified with an *unknown but fixed* circuit, the corresponding target concept is denoted by $f^*(\mathbf{x}, O) \in \mathcal{F}$. To infer $f^*(\mathbf{x}, O)$, in the training data collection procedure, the learner samples n classical inputs $\{\mathbf{x}^{(i)}\}_{i=1}^n$ from the distribution $\mathbb{D}_{\mathcal{X}}$, sends them into the quantum computer, and obtains the corresponding outcomes $\{\mathbf{o}^{(i)}\}_{i=1}^n$. The outcome $\mathbf{o}^{(i)}$ for $\forall i \in [n]$ is obtained by performing a *single-shot* measurement of O (i.e., $T = 1$) on $\rho(\mathbf{x}^{(i)})$ with $\mathbb{E}[\mathbf{o}^{(i)}] = f^*(\mathbf{x}^{(i)}, O)$. Let the collected dataset be

$$\mathcal{T} = \left\{ \left(\mathbf{x}^{(i)}, \mathbf{o}^{(i)} \right) \right\}_{i=1}^n. \quad (\text{D4})$$

What we are interested in here is the required number of training examples n in $\mathcal{T}$ that allows the learner to produce a prediction model $h_{\mathcal{T}}(\cdot)$ whose prediction $h_{\mathcal{T}}(\mathbf{x})$ is close to $f^*(\mathbf{x})$ on average. More formally, the sample complexity explored here refers to the minimal number of training examples n to ensure that with probability $1 - \delta$, the expected risk satisfies

$$\mathbb{E}_{\mathbf{x} \sim \mathbb{D}_{\mathcal{X}}} |h_{\mathcal{T}}(\mathbf{x}) - f^*(\mathbf{x})|^2 \leq \mathcal{O}(\epsilon). \quad (\text{D5})$$

Huang et al. [62] presented a method that is efficient in sample complexity (but maybe computationally demanding) for solving this task. In particular, according to Definition 1, denote the 4ϵ -packing net of $\mathcal{F}$ as $\mathcal{F}_{4\epsilon}^p$. The learner conducts the empirical risk minimization on the training dataset $\mathcal{T}$ to estimate the optimal hypothesis, i.e.,

$$\hat{h}_{\mathcal{T}} = \arg \min_{f \in \mathcal{F}_{4\epsilon}^p} \frac{1}{n} \sum_{i=1}^n \left| f(\mathbf{x}^{(i)}) - \mathbf{o}^{(i)} \right|^2. \quad (\text{D6})$$

Under this formalism, Ref. [62] proved that the required number of training examples n to achieve an ϵ prediction error, as summarized in the following proposition.

Proposition D.1 (Adapted from Proposition 1, [62]). *Suppose that the observable O satisfies $\sum_i \|O_i\|_{\infty} \leq B$. Let $\hat{h}_{\mathcal{T}}$ be an element of the 4ϵ -packing net of $\mathcal{F}$ with the packing number $\mathcal{M}(\mathcal{F}, 4\epsilon, |\cdot|^2)$ that minimizes the empirical training error in Eq. (D6). Then for $\delta \in (0, 1)$, the size of training data*

$$n \geq \frac{38B^2 \log(4\mathcal{M}(\mathcal{F}, 4\epsilon, |\cdot|^2)/\delta)}{\epsilon} \quad (\text{D7})$$

implies

$$\mathbb{E}_{\mathbf{x} \sim \mathbb{D}_{\mathcal{X}}} \left| \hat{h}_{\mathcal{T}}(\mathbf{x}) - f^*(\mathbf{x}) \right|^2 \leq 12\epsilon \quad \text{with probability at least } 1 - \delta. \quad (\text{D8})$$

Remark. Note that although the original proof of Proposition D.1 only concerns $\|O\| \leq 1$, the pertinent proof can be readily extended to the setting of $\sum_i \|O_i\|_{\infty} \leq B$ concerned in Theorem 1, and therefore we omit it here. Moreover, it only focuses on the case of $T = 1$, but the results still hold for the case of $T > 1$. This is because in the extreme case with $T \rightarrow \infty$, the required training data size is reduced to $n \geq 38B^2 \log(2\mathcal{M}(\mathcal{F}, 4\epsilon, \mathfrak{d})/\delta)/\epsilon$ [62, Equation (C66)]. In this regard, the setting of $T \geq 1$ only trivially influences the sample complexity bound (at most logarithmically). For this reason, we omit the relevant analysis in our proof.

Supported by Proposition D.1, the proof of Theorem B.2 amounts to quantifying the upper bound of the packing number $\mathcal{M}(\mathcal{F}, \epsilon, |\cdot|^2)$. Note that different from quantum neural networks [61, 63, 64] whose hypothesis space is continuous and the packing number depends on the value of ϵ , the function class $\mathcal{F}$ explored here is discrete and the number of elements in this class finite. As a result, we have $\mathcal{M}(\mathcal{F}, \epsilon, |\cdot|^2) \leq |\text{Arc}(\text{RZ}, \text{CI})|$ no matter how ϵ is. In light of this fact, we are now ready to show the proof of Theorem B.2.

Proof of Theorem B.2. According to the above explanation, this proof is composed of two parts. The first part is to quantify the upper bound of the packing number $\mathcal{M}(\mathcal{F}, \epsilon, |\cdot|^2)$, or equivalently the cardinality of $\text{Arc}(\text{RZ}, \text{CI})$. And the second part combines the obtained packing number and Proposition D.1 to acquire the upper bound of the sample complexity to learn $\mathcal{F}$.

We now derive the upper bound of $|\text{Arc}(\text{RZ}, \text{CI})|$. The total number of possible layouts for an N -qubit circuit consisting with d RZ gates and $G - d$ CI gates (i.e., $\text{CI} = \{H, S, \text{CNOT}\}$) is

$$|\text{Arc}(\text{RZ}, \text{CI})| \leq \binom{G}{d} \cdot N^d \cdot 3^{G-d} \cdot \binom{N}{2}^{G-d}, \quad (\text{D9})$$

where the first term $\binom{G}{d}$ computes the number of different arrangements for placing RZ or CI gates at each circuit depth, the second term N^d calculates the total combinations of placing d RZ gates on different qubit wires, the third term 3^{G-d} counts the total combinations of choosing different gates from the CI gate set, and the last term $\binom{N}{2}^{G-d}$ calculates the upper bound for the total combinations of placing the selected CI gates on different qubit wires.

The sample complexity can be efficiently obtained by combining Proposition D.1 and Eq. (D9). That is, the metric entropy of 4ϵ -packing net of $\mathcal{F}$ yields

$$\log(4\mathcal{M}(\mathcal{F}, 4\epsilon, |\cdot|)/\delta) \quad (\text{D10a})$$

$$\leq \log \left(\binom{G}{d} \right) + d \log(N) + (G-d) \log(3) + (G-d) \log \left(\binom{N}{2} \right) + \log \left(\frac{4}{\delta} \right) \quad (\text{D10b})$$

$$\leq d \log \left(\frac{eG}{d} \right) + d \log(N) + (G-d) \log(3) + 2(G-d)N \log \left(\frac{eN}{2} \right) + \log \left(\frac{4}{\delta} \right), \quad (\text{D10c})$$

where the second inequality uses $\binom{a}{b} \leq (ea/b)^b$.

The above result, accompanied by Proposition D.1, suggests the upper bound of the sample complexity to achieve the ϵ -prediction error, i.e.,

$$n = \frac{38B^2 \left(d \log \left(\frac{\epsilon G}{d} \right) + d \log(N) + (G-d) \log(3) + 2(G-d)N \log \left(\frac{\epsilon N}{2} \right) + \log \left(\frac{4}{\delta} \right) \right)}{\epsilon} \quad (\text{D11a})$$

$$\leq \tilde{\mathcal{O}} \left(\frac{B^2 d + B^2 N G}{\epsilon} \right). \quad (\text{D11b})$$

□

SM E: Exponential separation between computational complexity and sample complexity (Proof of Theorem B.3)

We analyze the sample complexity of learning bounded-gate quantum circuits with incoherent measurements and showcase the sample efficiency of this learning task in SM D. However, it is noteworthy that the sample efficiency does not necessarily imply the computational efficiency, as identifying the desired training examples may incur exponential running time. For instance, in the task of pretty-good tomography, the approach outlined in Ref. [65] requires only a linear number of training examples with the qubit count N for a low prediction error, yet the runtime cost scales exponentially with N .

In this section, we prove that learning bounded-gate quantum circuits with incoherent measurements also manifests an exponential separation between sample and computational complexity.

A very recent study [66] explores the computational hardness of learning quantum observables from the measured out data. Here we first introduce some necessary definitions, and then briefly review their key results, followed by elucidating how to generalize such results to our case to complete the proof of Theorem B.3.

BQP complexity class. BQP (short for bounded-error quantum polynomial time) is the class of promise problems that can be solved by polynomial-time quantum computations that may have some small probability of making an error [67]. Let $A = (A_{\text{yes}}, A_{\text{no}})$ be a promise problem and let $a, b : \mathbb{N} \rightarrow [0, 1]$ be functions. Then $A \in \text{BQP}(a, b)$ if and only if there exists a polynomial-time generated family of quantum circuits $\mathcal{Q} = \{\mathcal{Q}_N : N \in \mathbb{N}\}$, where each circuit $\mathcal{Q}_N$ takes N input qubits and produces one output qubit, that satisfies the following properties: (1) if $\mathbf{x} \in A_{\text{yes}}$ then $\Pr(\mathcal{Q} \text{ accepts } \mathbf{x}) \geq a(|\mathbf{x}|)$, and (2) if $\mathbf{x} \in A_{\text{no}}$ then $\Pr(\mathcal{Q} \text{ accepts } \mathbf{x}) \leq b(|\mathbf{x}|)$. The class is defined as $\text{BQP} = \text{BQP}(2/3, 1/3)$.

P/poly complexity class. P/poly, different from BQP, represents the class of decision problems that can be solved by a polynomial-time deterministic algorithm with access to an auxiliary ‘advice’ bitstring [68]. This advice bitstring depends only on the input size and must be identical for all inputs of a given size, regardless of the specific input $\mathbf{x}$. More specifically, a language $L : \{0, 1\}^* \rightarrow \{0, 1\}$ is in P/poly if there exists: (1) a polynomial-time deterministic algorithm $\mathcal{A}$, and (2) a family of ‘advice’ bitstrings $\{\mathbf{a}_N\}_{N \in \mathbb{N}}$, where $\mathbf{a}_N \in \{0, 1\}^{\text{poly}(N)}$, such that for every input $\mathbf{x} \in \{0, 1\}^N$: $\mathcal{A}(\mathbf{x}, \mathbf{a}_N) = L(\mathbf{x})$. Here $\{0, 1\}^*$ denotes the set of all finite-length binary strings.

Note that it is widely believed that BQP is not entirely contained within P/poly. Otherwise, problems like factoring would become solvable by polynomial-size classical circuits.

Learning procedure. Ref. [66] leveraged the relation between BQP and P/poly classes to understand the fundamental limitation of classical learners when learning observables from the measurement data. Denote the concept class as

$$\mathcal{F} = \left\{ f_{\alpha}(\mathbf{x}^{\perp}) = \text{Tr}(\rho_{\text{H}}(\mathbf{x}^{\perp})O(\alpha)) \mid \alpha \in \{-1, 1\}^q \right\}, \quad (\text{E1})$$

where $\rho_{\text{H}}(\mathbf{x}^{\perp}) = U_{\text{H}} |\mathbf{x}^{\perp}\rangle \langle \mathbf{x}^{\perp}| U_{\text{H}}^{\dagger}$ with $\mathbf{x}^{\perp}$ being a bitstring, $U_{\text{H}} = e^{-i\text{H}\tau}$, and $O(\alpha) = \sum_{i=1}^q \alpha_i P_i$ with P_i being the i -th k -local Pauli string.

The goal of the classical learner is to learn a model $h(\mathbf{x}^{\perp})$ which approximates the unknown concept $f_{\alpha}(\mathbf{x}^{\perp})$. The model $h(\mathbf{x}^{\perp})$ is inferred by the training dataset $\mathcal{T} = \{\mathbf{x}^{\perp, (i)}, \hat{y}^{(i)}\}_{i=1}^n$ where $\hat{y}^{(i)}$ refers to the estimated label of $y^{(i)} = \text{Tr}(\rho_{\text{H}}(\mathbf{x}^{\perp, (i)})O(\alpha))$ with an additive error ϵ_2 for $\forall i \in [n]$. The estimation error arises from the finite number of measurements, as the classical learner inputs data $\mathbf{x}$ into the quantum system and relies on a limited number of measurements to obtain the estimated value. An immediate observation is that the procedure of learning by measurement data is identical to the learning paradigm investigated in this work. Once the data collection process is completed, the learner utilizes $\mathcal{T}$ to infer the prediction model $h(\mathbf{x}^{\perp})$ on the classical side.

The efficacy of the classical learner towards the concept class $\mathcal{F}$ is quantified by the following definition.

Definition 2 (Efficient learning condition, Definition 4 in Ref. [66]). A concept class $\mathcal{F}$ in Eq. (E1) is efficiently learnable if there exists a polynomial-time algorithm $\mathcal{A}$ with complexity $\text{poly}(1/\epsilon, 1/\delta, 1/\epsilon_2, N)$, such that for any $\epsilon, \epsilon_2 > 0$, any $0 < \delta < 1$, any target function $f_\alpha \in \mathcal{F}$, and any input distribution $\mathbb{D}$, the following holds:

If $\mathcal{A}$ receives a training set $\mathcal{T} = \{\mathbf{x}^{\perp, (i)}, \hat{y}^{(i)}\}_{i=1}^n$ with $n \sim \text{poly}(1/\epsilon, 1/\delta, 1/\epsilon_2, N)$, it produces a hypothesis $h(\cdot) = \mathcal{A}(\mathcal{T}, \epsilon, \delta, \cdot)$ such that, with probability at least $1 - \delta$, the expected error satisfies $\mathbb{E}_{\mathbf{x}^\perp \sim \mathbb{D}} [|f_\alpha(\mathbf{x}^\perp) - h(\mathbf{x}^\perp)|^2] \leq \epsilon$.

By leveraging the concept of BQP and P/poly classes, Ref. [66] shows the hardness of learning $\mathcal{F}$ from the measurement data.

Lemma E.1 (Adapted from Theorem 2, [66]). For any BQP-complete language, there exists a Hamiltonian H_{hard} to build the concept class

$$\mathcal{F}_{\text{Hard}} = \left\{ f_\alpha(\mathbf{x}^\perp) = \text{Tr}(\rho_{H_{\text{hard}}}(\mathbf{x}^\perp) O(\alpha)) \mid \alpha \in \{-1, 1\}^q \right\}, \quad (\text{E2})$$

where $\rho_H(\mathbf{x}^\perp) = U_{H_{\text{hard}}} |\mathbf{x}^\perp\rangle \langle \mathbf{x}^\perp| U_{H_{\text{hard}}}^\dagger$ with $U_{H_{\text{hard}}} = e^{-iH_{\text{hard}}\tau}$ and $O(\alpha) = \sum_{i=1}^q \alpha_i P_i$ with P_i being the i -th k -local Pauli string. Moreover, no randomized polynomial-time classical algorithm satisfies the learning condition of Def. 2 for this concept class, unless $\text{BQP} \subseteq \text{P/poly}$.

The main idea of the proof for Lemma E.1 is as follows. First, the authors correlate $\mathcal{F}_{\text{Hard}}$ with BQP circuits. In particular, the unitary $U_{H_{\text{hard}}}$ in $\mathcal{F}_{\text{Hard}}$ is specified to be a family of quantum circuit $\{U_{\text{BQP}^N}\}_N$, which decides the BQP-complete language L , one circuit per size; the observable is set as $O(\alpha^*) = Z \otimes \mathbb{I}_2 \otimes \cdots \otimes \mathbb{I}_2$. In this way, the output of quantum circuits can correctly decide every $\mathbf{x}^\perp \in L$, i.e., $f(\mathbf{x}^\perp) > 0$ if $\mathbf{x} \in L$ and $f(\mathbf{x}^\perp) < 0$ if $\mathbf{x} \notin L$. Then, the authors show that if $\mathcal{F}_{\text{Hard}}$ is learnable in the sense of Def. 2, then there must exist a polynomial size classical circuit which evaluates $f_{\alpha^*} = \text{Tr}(\rho_{H_{\text{hard}}}(\mathbf{x}^\perp) O(\alpha^*))$ correctly on every $\mathbf{x}^\perp \in \{0, 1\}^N$, implying $\text{BQP} \subseteq \text{P/poly}$.

We next generalize the results of Lemma E.1 to show the computational hardness of learning the bounded-gate circuit with incoherent dynamics. To achieve this goal, it is sufficient to show the concept class $\mathcal{F}$ in Eq. (3) of the main text can represent the concept class of BQP circuits discussed above.

For BQP circuits, the observable $Z \otimes \mathbb{I}_2 \otimes \cdots \otimes \mathbb{I}_2$ meets the requirement of the observable defined in $\mathcal{F}$, which is formed by Pauli operators with a bounded norm. As for the state $U_{\text{BQP}} |\mathbf{x}^\perp\rangle$ (or equivalently $U_{H_{\text{hard}}} |\mathbf{x}^\perp\rangle$), it can also be expressed by the bounded-gate circuit $\{\text{RZ} + \text{CI}\}$ with the initial state $|0\rangle^{\otimes N}$. Specifically, the bounded-gate circuit is decomposed into two parts, where the first part is used to prepare the state $|\mathbf{x}^\perp\rangle$ and the second part is to prepare the BQP circuit. For the first part, N RZ gates and $\mathcal{O}(N)$ CI gates associated with a proper distribution are sufficient to prepare any input state $|\mathbf{x}^\perp\rangle$ with $\mathbf{x}^\perp \in \{0, 1\}^N$. For the second part, since U_{BQP} contains at most $\mathcal{O}(\text{poly}(N))$ quantum gates and $\{\text{RZ} + \text{CI}\}$ is a universal basis gate set, $\mathcal{O}(\text{poly}(N))$ RZ gates with a proper distribution over the classical inputs and $\mathcal{O}(\text{poly}(N))$ CI gates are sufficient to synthesis U_{BQP} . Taken together, the BQP circuit belongs to $\mathcal{F}$ when the number of $\{\text{RZ} + \text{CI}\}$ gates polynomially scale with N .

The reformulation enables us to generalize the hardness of learning quantum observables from measured data to the broader challenge of learning bounded-gate circuits, which is the focus of our work. That is, according to the results of Lemma E.1, we can conclude that no algorithm can only use the measure-out data to learn the bounded-gate quantum circuit within a polynomial time unless $\text{BQP} \subseteq \text{P/poly}$. This proves Theorem B.3.

SM F: Learnability of the proposed kernel-based ML model (Proof of Theorem 2)

This section provides the proof of Theorem 2, which analyzes how the prediction error of the proposed kernel-based ML model depends on the number of training examples n , the size of the quantum system N , and the dimension of classical inputs d . Recall that in the main text, the proposed state prediction model is

$$\hat{\sigma}_n(\mathbf{x}) = \frac{1}{n} \sum_{i=1}^n \kappa_\Lambda(\mathbf{x}, \mathbf{x}^{(i)}) \tilde{\rho}_T(\mathbf{x}^{(i)}) \text{ with } \kappa_\Lambda(\mathbf{x}, \mathbf{x}^{(i)}) = \sum_{\omega, \|\omega\|_0 \leq \Lambda} 2^{\|\omega\|_0} \Phi_\omega(\mathbf{x}) \Phi_\omega(\mathbf{x}^{(i)}) \in \mathbb{R}. \quad (\text{F1})$$

What we intend to prove is the average discrepancy between $\text{Tr}(\hat{\sigma}_n(\mathbf{x})O)$ and the ground truth $\text{Tr}(\rho(\mathbf{x})O)$ when $\mathbf{x}$ is uniformly and randomly sampled from $[-\pi, \pi]^d$ and the local observable O is sampled from a prior distribution $\mathbb{D}_O$, i.e., $\mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} |\text{Tr}(O\hat{\sigma}_n(\mathbf{x})) - \text{Tr}(O\rho(\mathbf{x}))|^2$.

Before moving to proceed with the further analysis, let us exhibit the formal statement of Theorem 2.

Theorem (Restatement of Theorem 2). *Following notations in the main text, consider a parametrized family of N -qubit states $\mathcal{Q}$ and a sum $O = \sum_{i=1} O_i$ of multiple local observables with $\sum_i \|O_i\|_\infty \leq B$ and the maximum locality of $\{O_i\}$ being K . Suppose $\mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} \|\nabla_{\mathbf{x}} \text{Tr}(\rho(\mathbf{x})O)\|_2^2 \leq C$. Then, let the dataset be $\mathcal{T}_s = \{\mathbf{x}^{(i)} \rightarrow \tilde{\rho}(\mathbf{x}^{(i)})\}_{i=1}^n$ with $\mathbf{x}^{(i)} \sim \text{Unif}[-\pi, \pi]^d$ and $n = |\mathfrak{C}(\Lambda)| \frac{2B^2 9^K}{\epsilon} \log(2 \cdot |\mathfrak{C}(\Lambda)|/\delta)$ with $\mathfrak{C}(\Lambda) = \{\boldsymbol{\omega} | \boldsymbol{\omega} \in \{0, \pm 1\}^d, \text{ s.t. } \|\boldsymbol{\omega}\|_0 \leq \Lambda\}$. When the frequency is truncated to $\Lambda = 4C/\epsilon$, the state prediction model in Eq. (F1) achieves*

$$\mathbb{E}_{\mathcal{T}_s}[\hat{\sigma}_n(\mathbf{x})] = \rho_\Lambda(\mathbf{x}) \quad (\text{F2})$$

and with probability at least $1 - \delta$,

$$\mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} |\text{Tr}(O\hat{\sigma}_n(\mathbf{x})) - \text{Tr}(O\rho(\mathbf{x}))|^2 \leq \epsilon. \quad (\text{F3})$$

To reach Theorem 2, we first use the triangle inequality to decouple the difference between the prediction and ground truth into the truncation error and the estimation error, i.e.,

$$\mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} \left[|\text{Tr}(O\hat{\sigma}_n(\mathbf{x})) - \text{Tr}(O\rho(\mathbf{x}))|^2 \right] \quad (\text{F4a})$$

$$\leq \left(\sqrt{\mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} \left[|\text{Tr}(O\rho_\Lambda(\mathbf{x})) - \text{Tr}(O\rho(\mathbf{x}))|^2 \right]} + \sqrt{\mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} \left[|\text{Tr}(O\hat{\sigma}_n(\mathbf{x})) - \text{Tr}(O\rho_\Lambda(\mathbf{x}))|^2 \right]} \right)^2. \quad (\text{F4b})$$

After decoupling, we then separately derive the upper bound of these two terms, where the relevant results are encapsulated in the following two lemmas whose proofs are given in the subsequent two subsections.

Lemma F.1 (Truncation error of ρ_Λ). *Following notations in Theorem 2, assuming $\mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} \|\nabla_{\mathbf{x}} \text{Tr}(\rho(\mathbf{x})O)\|_2^2 \leq C$, the truncation error induced by removing high-frequency terms of ρ under the trigonometric expansion with $\|\boldsymbol{\omega}\|_0 \leq \Lambda$ is upper bounded by*

$$\mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} |\text{Tr}(O\rho_\Lambda(\mathbf{x})) - \text{Tr}(O\rho(\mathbf{x}))|^2 \leq \frac{C}{\Lambda}. \quad (\text{F5})$$

Lemma F.2 (Estimation error of $\hat{\sigma}_n$). *Following notations in Theorem 2, with probability at least $1 - \delta$, the estimation error induced by finite training examples $\mathcal{T} = \{\tilde{\rho}(\mathbf{x}^{(i)})\}_{i=1}^n$ is upper bounded by*

$$\mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} \left[|\text{Tr}(O\hat{\sigma}_n(\mathbf{x})) - \text{Tr}(O\rho_\Lambda(\mathbf{x}))|^2 \right] \leq |\mathfrak{C}(\Lambda)| \frac{1}{2n} B^2 9^K \log \left(\frac{2 \cdot |\mathfrak{C}(\Lambda)|}{\delta} \right), \quad (\text{F6})$$

where $\mathfrak{C}(\Lambda) = \{\boldsymbol{\omega} | \boldsymbol{\omega} \in \{0, \pm 1\}^d, \text{ s.t. } \|\boldsymbol{\omega}\|_0 \leq \Lambda\}$ refers to the set of truncated frequencies.

We are now ready to present the proof of Theorem 2.

Proof of Theorem 2. The difference between the prediction and ground truth can be obtained by integrating Lemmas F.1 and F.2 into Eq. (F4a). Mathematically, with probability at least $1 - \delta$, we have

$$\mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} \left[|\text{Tr}(O\hat{\sigma}_n(\mathbf{x})) - \text{Tr}(O\rho(\mathbf{x}))|^2 \right] \leq \left(\sqrt{\frac{C}{\Lambda}} + \sqrt{|\mathfrak{C}(\Lambda)| \frac{1}{2n} B^2 9^K \log \left(\frac{2 \cdot |\mathfrak{C}(\Lambda)|}{\delta} \right)} \right)^2. \quad (\text{F7})$$

To ensure the average prediction error is upper bounded by ϵ , it is sufficient to showcase when the inner two terms are upper bounded by $\sqrt{\epsilon}/2$. For the first term, the condition is satisfied when

$$\sqrt{\frac{C}{\Lambda}} \leq \frac{\sqrt{\epsilon}}{2} \Leftrightarrow \Lambda \geq \frac{4C}{\epsilon}. \quad (\text{F8})$$

For the second term, we have

$$\sqrt{|\mathfrak{C}(\Lambda)| \frac{1}{2n} B^2 9^K \log \left(\frac{2 \cdot |\mathfrak{C}(\Lambda)|}{\delta} \right)} \leq \frac{\sqrt{\epsilon}}{2} \Leftrightarrow n \geq |\mathfrak{C}(\Lambda)| \frac{2B^2 9^K}{\epsilon} \log \left(\frac{2 \cdot |\mathfrak{C}(\Lambda)|}{\delta} \right). \quad (\text{F9})$$

Taken together, with probability $1 - \delta$, the prediction error is upper bounded by ϵ when the number of training examples satisfies

$$n \geq |\mathfrak{C}(4C/\epsilon)| \frac{2B^2 9^K}{\epsilon} \log \left(\frac{2 \cdot |\mathfrak{C}(4C/\epsilon)|}{\delta} \right). \quad (\text{F10})$$

□

1. Truncation error of the classical learning model (Proof of Lemma F.1)

Recall that under the trigonometric monomial expansion, the target state ρ with truncation and without truncation takes the form as

$$\rho_\Lambda = \sum_{\omega \in \mathfrak{C}(\Lambda)} \Phi_\omega(\mathbf{x}) \rho_\omega \text{ and } \rho = \sum_{\omega \in \mathfrak{C}(d)} \Phi_\omega(\mathbf{x}) \rho_\omega, \quad (\text{F11})$$

respectively. The purpose of Lemma F.1 is to analyze the upper bound of the discrepancy between $\text{Tr}(\rho_\Lambda O)$ and $\text{Tr}(\rho O)$ induced by the truncation of high-frequency terms.

Proof of Lemma F.1. By adopting the explicit trigonometric monomial expansion of ρ and ρ_Λ in Eq. (F11), we have

$$\mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} |\text{Tr}(O \rho_\Lambda(\mathbf{x}) - \text{Tr}(O \rho(\mathbf{x}))|^2 \quad (\text{F12a})$$

$$= \mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} \left| \sum_{\omega, \|\omega\|_0 > \Lambda} \Phi_\omega(\mathbf{x}) \langle \rho_\omega | O \rangle \right|^2 \quad (\text{F12b})$$

$$= \mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} \sum_{\omega, \|\omega\|_0 > \Lambda} \sum_{\omega', \|\omega'\|_0 > \Lambda} \Phi_\omega(\mathbf{x}) \Phi_{\omega'}(\mathbf{x}) \langle \rho_\omega | O \rangle \langle \rho_{\omega'} | O \rangle \quad (\text{F12c})$$

$$= \sum_{\omega, \|\omega\|_0 > \Lambda} \sum_{\omega', \|\omega'\|_0 > \Lambda} \frac{1}{(2\pi)^d} \int_{[-\pi, \pi]^d} \Phi_\omega(\mathbf{x}) \Phi_{\omega'}(\mathbf{x}) \langle \rho_\omega | O \rangle \langle \rho_{\omega'} | O \rangle d^d x \quad (\text{F12d})$$

$$= \sum_{\omega, \|\omega\|_0 > \Lambda} \sum_{\omega', \|\omega'\|_0 > \Lambda} 2^{-\|\omega\|_0} \delta_{\omega, \omega'} \langle \rho_\omega | O \rangle \langle \rho_{\omega'} | O \rangle \quad (\text{F12e})$$

$$= \sum_{\omega, \|\omega\|_0 > \Lambda} 2^{-\|\omega\|_0} |\langle \rho_\omega | O \rangle|^2 := \sum_{\omega, \|\omega\|_0 > \Lambda} 2^{-\|\omega\|_0} \alpha_\omega^2, \quad (\text{F12f})$$

where the first three equalities follow a direct reformulation, Eq. (F12e) employs the orthogonality of basis functions in the trigonometric expansion, i.e.,

$$\frac{1}{(2\pi)^d} \int_{[-\pi, \pi]^d} \Phi_\omega(\mathbf{x}) \Phi_{\omega'}(\mathbf{x}) d^d x = 2^{-\|\omega\|_0} \delta_{\omega, \omega'}, \quad (\text{F13})$$

and in the last equality we define $\alpha_\omega \equiv \text{Tr}(\rho_\omega O)$ for clarification.

The remainder of the proof uses the assumption of the norm of the gradients of the expectation, i.e., $\|\nabla_{\mathbf{x}} \text{Tr}(\rho(\mathbf{x}) O)\|_2^2 \leq C$, to derive the upper bound of Eq. (F12f). To do so, we first derive the explicit form of the gradient under the trigonometric monomial expansion. That is, the gradient of $\text{Tr}(\rho(\mathbf{x}) O)$ with respect to $\mathbf{x}$ is a d -dimensional vector, i.e.,

$$\nabla_{\mathbf{x}} \text{Tr}(\rho(\mathbf{x}) O) \quad (\text{F14a})$$

$$= \nabla_{\mathbf{x}} \sum_{\omega} \Phi_\omega(\mathbf{x}) \text{Tr}(\rho_\omega O)$$

$$= \sum_{\omega} \nabla_{\mathbf{x}} \Phi_\omega(\mathbf{x}) \alpha_\omega \quad (\text{F14b})$$

$$= \left[\sum_{\omega} \Psi_{\omega_1}(\mathbf{x}_1) \Phi_{\omega_{2:d}}(\mathbf{x}_{2:d}) \alpha_\omega, \dots, \sum_{\omega} \Phi_{\omega_{1:i-1}}(\mathbf{x}_{1:i-1}) \Psi_{\omega_i}(\mathbf{x}_i) \Phi_{\omega_{i+1:d}}(\mathbf{x}_{i+1:d}) \alpha_\omega, \right. \quad (\text{F14c})$$

$$\left. \dots, \sum_{\omega} \Psi_{\omega_{1:d-1}}(\mathbf{x}_{1:d-1}) \Psi_{\omega_d}(\mathbf{x}_d) \alpha_\omega \right]^\top \in \mathbb{R}^d, \quad (\text{F14d})$$

where $\Phi_{\omega_{a:b}}(\mathbf{x}_{a:b}) \equiv \prod_{i=a}^b \Phi_{\omega_i}(\mathbf{x}_i)$, and the derivative with respect to the i -th entry is

$$\Psi_{\omega_i}(\mathbf{x}_i) \equiv \nabla_{\mathbf{x}_i} \Phi_{\omega_i}(\mathbf{x}_i) = \begin{cases} 0 & \text{if } \omega_i = 0 \\ -\sin(\mathbf{x}_i) & \text{if } \omega_i = 1 \\ \cos(\mathbf{x}_i) & \text{if } \omega_i = -1 \end{cases}. \quad (\text{F15})$$

By making use of the above formula, the expectation value of $\|\nabla_{\mathbf{x}} \text{Tr}(\rho(\mathbf{x})O)\|_2^2$ over $\mathbf{x} \sim \text{Unif}[-\pi, \pi]^d$ satisfies

$$\mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} \|\nabla_{\mathbf{x}} \text{Tr}(\rho(\mathbf{x})O)\|_2^2 \quad (\text{F16a})$$

$$= \frac{1}{(2\pi)^d} \int_{[-\pi, \pi]^d} \left[\left(\sum_{\omega} \Psi_{\omega_1}(\mathbf{x}_1) \Phi_{\omega_{2:d}}(\mathbf{x}_{2:d}) \alpha_{\omega} \sum_{\omega'} \Psi_{\omega'_1}(\mathbf{x}_1) \Phi_{\omega'_{2:d}}(\mathbf{x}_{2:d}) \alpha_{\omega'} \right) + \dots \right] \quad (\text{F16b})$$

$$+ \left(\sum_{\omega} \Phi_{\omega_{1:i-1}}(\mathbf{x}_{1:i-1}) \Psi_{\omega_i}(\mathbf{x}_i) \Phi_{\omega_{i+1:d}}(\mathbf{x}_{i+1:d}) \alpha_{\omega} \sum_{\omega'} \Phi_{\omega'_{1:i-1}}(\mathbf{x}_{1:i-1}) \Psi_{\omega'_i}(\mathbf{x}_i) \Phi_{\omega'_{i+1:d}}(\mathbf{x}_{i+1:d}) \alpha_{\omega'} \right) + \dots \quad (\text{F16c})$$

$$+ \left(\sum_{\omega} \Phi_{\omega_{1:d-1}}(\mathbf{x}_{1:d-1}) \Psi_{\omega_d}(\mathbf{x}_d) \alpha_{\omega} \sum_{\omega'} \Phi_{\omega'_{1:d-1}}(\mathbf{x}_{1:d-1}) \Psi_{\omega'_d}(\mathbf{x}_d) \alpha_{\omega'} \right) \Big] d^d x \quad (\text{F16d})$$

$$= \frac{1}{(2\pi)^d} \left[\pi \int_{[-\pi, \pi]^{d-1}} \left(\sum_{\omega_{2:d}} \Phi_{\omega_{2:d}}(\mathbf{x}_{2:d}) \alpha_{1, \omega_{2:d}} \sum_{\omega'_{2:d}} \Phi_{\omega'_{2:d}}(\mathbf{x}_{2:d}) \alpha_{1, \omega'_{2:d}} \right) d^{d-1} x + \dots \right] \quad (\text{F16e})$$

$$+ \pi \int_{[-\pi, \pi]^{d-1}} \left(\sum_{\omega_{2:d}} \Phi_{\omega_{2:d}}(\mathbf{x}_{2:d}) \alpha_{-1, \omega_{2:d}} \sum_{\omega'_{2:d}} \Phi_{\omega'_{2:d}}(\mathbf{x}_{2:d}) \alpha_{-1, \omega'_{2:d}} \right) d^{d-1} x + \dots \quad (\text{F16f})$$

$$+ \pi \int_{[-\pi, \pi]^{d-1}} \left(\sum_{\omega_{1:d-1}} \Phi_{\omega_{1:d-1}}(\mathbf{x}_{1:d-1}) \alpha_{\omega_{1:d-1}, 1} \sum_{\omega'_{1:d-1}} \Phi_{\omega'_{1:d-1}}(\mathbf{x}_{1:d-1}) \alpha_{\omega'_{1:d-1}, 1} \right) d^{d-1} x \quad (\text{F16g})$$

$$+ \pi \int_{[-\pi, \pi]^{d-1}} \left(\sum_{\omega_{1:d-1}} \Phi_{\omega_{1:d-1}}(\mathbf{x}_{1:d-1}) \alpha_{\omega_{1:d-1}, -1} \sum_{\omega'_{1:d-1}} \Phi_{\omega'_{1:d-1}}(\mathbf{x}_{1:d-1}) \alpha_{\omega'_{1:d-1}, -1} \right) d^{d-1} x \Big] \quad (\text{F16h})$$

$$= \left[\sum_{\omega_{2:d}} 2^{-\|[1, \omega_{2:d}]\|_0} |\alpha_{1, \omega_{2:d}}|^2 + \sum_{\omega_{2:d}} 2^{-\|[-1, \omega_{2:d}]\|_0} |\alpha_{-1, \omega_{2:d}}|^2 + \dots \right] \quad (\text{F16i})$$

$$+ \sum_{\omega_{1:d-1}} 2^{-\|[\omega_{1:d-1}, 1]\|_0} |\alpha_{\omega_{1:d-1}, 1}|^2 + \sum_{\omega_{1:d-1}} 2^{-\|[\omega_{1:d-1}, -1]\|_0} |\alpha_{\omega_{1:d-1}, -1}|^2 := \star, \quad (\text{F16j})$$

where Eqs. (F16e)-(F16h) are obtained by taking the expectation value on the gradient term Ψ_{ω_i} for $\forall i \in [d]$, the quantity $\alpha_{\pm 1, \omega_{2:d}}$ refers to $\text{Tr}(\rho_{\pm 1, \omega_{2:d}} O)$ with the frequency value at the 1-st position being ± 1 (the same rule applies to other terms such as $\alpha_{\omega_{1:i}, i+1, \omega_{i+2:d}}$), and Eq. (F16j) is obtained by taking integral over each term in the bracket and the orthogonality of basis functions in $\Phi_{\omega}(\mathbf{x})$ and $\Phi_{\omega'}(\mathbf{x})$ as shown in Eq. (F13).

We now use the reformulated $\mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} \|\nabla_{\mathbf{x}} \text{Tr}(\rho(\mathbf{x})O)\|_2^2$ to derive the upper bound of the truncation error. Namely, Eq. (F12f) can be reformulated as

$$\mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} |\text{Tr}(O\rho_{\Lambda}(\mathbf{x}) - \text{Tr}(O\rho(\mathbf{x}))|^2 \quad (\text{F17a})$$

$$= \sum_{\omega, \|\omega\|_0 > \Lambda} 2^{-\|\omega\|_0} |\alpha_{\omega}|^2 \quad (\text{F17b})$$

$$= \frac{1}{\Lambda} \left(\Lambda \sum_{\omega, \|\omega\|_0 > \Lambda} 2^{-\|\omega\|_0} |\alpha_{\omega}|^2 \right) \quad (\text{F17c})$$

$$\leq \frac{\mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} \|\nabla_{\mathbf{x}} \text{Tr}(\rho(\mathbf{x})O)\|_2^2}{\Lambda}, \quad (\text{F17d})$$

$$\leq \frac{C}{\Lambda}, \quad (\text{F17e})$$

where the last second inequality is supported by the result indicated below and the last inequality is supported by the assumption $\mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} \|\nabla_{\mathbf{x}} \text{Tr}(\rho(\mathbf{x})O)\|_2^2 \leq C$.

The proof of Eq. (F17d). According to Eq. (F16j), to reach Eq. (F17d), it is equivalent to proving that for $\forall \Lambda \in [d]$,

$$\Lambda \sum_{\omega, \|\omega\|_0 > \Lambda} 2^{-\|\omega\|_0} |\alpha_{\omega}|^2 \leq \star. \quad (\text{F18})$$

Note that depending on the number of non-zero entries of the frequency (i.e., $\|\boldsymbol{\omega}\|_0$), the left and right-hand sides in Eq. (F18) can be decomposed into $d - \Lambda$ parts and $d - 1$ parts, respectively. Mathematically, the left hand side of Eq. (F18) yields

$$\Lambda \left(\sum_{\boldsymbol{\omega}, \|\boldsymbol{\omega}\|_0 = \Lambda+1} 2^{-\|\boldsymbol{\omega}\|_0} |\alpha_{\boldsymbol{\omega}}|^2 + \sum_{\boldsymbol{\omega}, \|\boldsymbol{\omega}\|_0 = \Lambda+2} 2^{-\|\boldsymbol{\omega}\|_0} |\alpha_{\boldsymbol{\omega}}|^2 + \dots + \sum_{\boldsymbol{\omega}, \|\boldsymbol{\omega}\|_0 = d} 2^{-\|\boldsymbol{\omega}\|_0} |\alpha_{\boldsymbol{\omega}}|^2 \right). \quad (\text{F19})$$

Besides, the right-hand side of Eq. (F18), i.e., $\star$, can be rewritten as

$$\star = \left[\left(\sum_{\boldsymbol{\omega}_{2:d}, \|\boldsymbol{\omega}_{2:d}\|_0 = 0} 2^{-\|[1, \boldsymbol{\omega}_{2:d}]\|_0} |\alpha_{1, \boldsymbol{\omega}_{2:d}}|^2 + \dots + \sum_{\boldsymbol{\omega}_{2:d}, \|\boldsymbol{\omega}_{2:d}\|_0 = d-1} 2^{-\|[1, \boldsymbol{\omega}_{2:d}]\|_0} |\alpha_{1, \boldsymbol{\omega}_{2:d}}|^2 \right) \right] \quad (\text{F20})$$

$$+ \dots \quad (\text{F21})$$

$$+ \left(\sum_{\boldsymbol{\omega}_{1:d-1}, \|\boldsymbol{\omega}_{1:d-1}\|_0 = 1} 2^{-\|[\boldsymbol{\omega}_{1:d-1}, -1]\|_0} |\alpha_{\boldsymbol{\omega}_{1:d-1}, -1}|^2 + \dots + \sum_{\boldsymbol{\omega}_{1:d-1}, \|\boldsymbol{\omega}_{1:d-1}\|_0 = d-1} 2^{-\|[\boldsymbol{\omega}_{1:d-1}, -1]\|_0} |\alpha_{\boldsymbol{\omega}_{1:d-1}, -1}|^2 \right) \quad (\text{F22})$$

$$= \left(\sum_{\boldsymbol{\omega}_{2:d}, \|\boldsymbol{\omega}_{2:d}\|_0 = 0} 2^{-\|[1, \boldsymbol{\omega}_{2:d}]\|_0} |\alpha_{1, \boldsymbol{\omega}_{2:d}}|^2 + \dots + \sum_{\boldsymbol{\omega}_{1:d-1}, \|\boldsymbol{\omega}_{1:d-1}\|_0 = 0} 2^{-\|[\boldsymbol{\omega}_{1:d-1}, -1]\|_0} |\alpha_{\boldsymbol{\omega}_{1:d-1}, -1}|^2 \right) \quad (\text{F23})$$

$$+ \dots \quad (\text{F24})$$

$$+ \left(\sum_{\boldsymbol{\omega}_{2:d}, \|\boldsymbol{\omega}_{2:d}\|_0 = d-1} 2^{-\|[1, \boldsymbol{\omega}_{2:d}]\|_0} |\alpha_{1, \boldsymbol{\omega}_{2:d}}|^2 + \dots + \sum_{\boldsymbol{\omega}_{1:d-1}, \|\boldsymbol{\omega}_{1:d-1}\|_0 = d-1} 2^{-\|[\boldsymbol{\omega}_{1:d-1}, -1]\|_0} |\alpha_{\boldsymbol{\omega}_{1:d-1}, -1}|^2 \right), \quad (\text{F25})$$

where the second equality is acquired by rearranging, as in every bracket, all frequencies have the same number of non-zero entries.

In conjunction with Eqs. (F19) and (F25), to achieve Eq. (F18), we need to demonstrate that for any $\Lambda < \Lambda' < d$, the following relation is satisfied, i.e.,

$$\Lambda \sum_{\boldsymbol{\omega}, \|\boldsymbol{\omega}\|_0 = \Lambda'} 2^{-\|\boldsymbol{\omega}\|_0} |\alpha_{\boldsymbol{\omega}}|^2 \leq \sum_{\boldsymbol{\omega}_{2:d}, \|\boldsymbol{\omega}_{2:d}\|_0 = \Lambda' - 1} 2^{-\|[1, \boldsymbol{\omega}_{2:d}]\|_0} |\alpha_{1, \boldsymbol{\omega}_{2:d}}|^2 + \dots + \sum_{\boldsymbol{\omega}_{1:d-1}, \|\boldsymbol{\omega}_{1:d-1}\|_0 = \Lambda' - 1} 2^{-\|[\boldsymbol{\omega}_{1:d-1}, -1]\|_0} |\alpha_{\boldsymbol{\omega}_{1:d-1}, -1}|^2. \quad (\text{F26})$$

To achieve this goal, we next prove that the right-hand side in Eq. (F26) equals to

$$\Lambda' \sum_{\boldsymbol{\omega}, \|\boldsymbol{\omega}\|_0 = \Lambda'} 2^{-\|\boldsymbol{\omega}\|_0} |\alpha_{\boldsymbol{\omega}}|^2, \quad \text{with } \Lambda < \Lambda' < d. \quad (\text{F27})$$

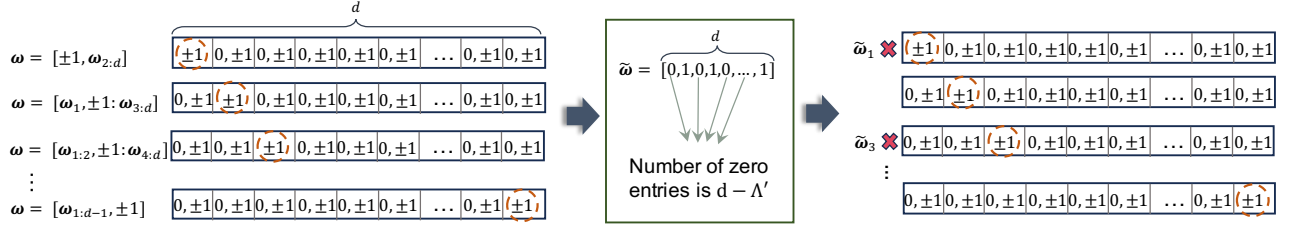
Recall that in Eq. (F27), each frequency $\boldsymbol{\omega} \in \{0, \pm 1\}^d$ with $\|\boldsymbol{\omega}\|_0 = \Lambda'$ appears Λ' times and the total number of frequencies is

$$\Lambda \cdot \left| \left\{ \boldsymbol{\omega} \in \{0, \pm 1\}^d \mid \|\boldsymbol{\omega}\|_0 = \Lambda' \right\} \right| = \Lambda \cdot \left| \binom{d}{d - \Lambda'} \right| \cdot 2^{\Lambda'} = \Lambda' \frac{d!}{(d - \Lambda')! \Lambda'!} 2^{\Lambda'} = \frac{d!}{(d - \Lambda')! (\Lambda' - 1)!} 2^{\Lambda'}. \quad (\text{F28})$$

In addition, the total number of frequencies in the right-hand side of Eq. (F26) is

$$\binom{d-1}{d - \Lambda' - 1} 2^{\Lambda'} + \dots + \binom{d-1}{d - \Lambda' - 1} 2^{\Lambda'} = d \cdot \binom{d-1}{d - \Lambda' - 1} 2^{\Lambda'} = d \frac{(d-1)!}{(d - \Lambda')! (\Lambda' - 1)!} 2^{\Lambda'} = \frac{d!}{(d - \Lambda')! (\Lambda' - 1)!} 2^{\Lambda'}. \quad (\text{F29})$$

Combining with the above two equations, we know that the total number of frequencies in Eq. (F27) and the right-hand side of Eq. (F26) is the same. Accordingly, their equivalence can be reached if we can show that each frequency $\boldsymbol{\omega} \in \{0, \pm 1\}^d$ with $\|\boldsymbol{\omega}\|_0 = \Lambda'$ appears Λ' times in the right-hand side of Eq. (F26). This is indeed the case, as the visual interpretation is shown in Fig. 2. Denote that the specified frequency as $\tilde{\boldsymbol{\omega}}$ with $\|\tilde{\boldsymbol{\omega}}\|_0 = \Lambda'$. In other words, there are $d - \Lambda'$ entries whose value is 0. This property allows us to determine how many times $\tilde{\boldsymbol{\omega}}$ appears on the right side of Eq. (F26). In particular, as shown in the left panel of Fig. 2, the right-hand side of Eq. (F26) can be divided into d groups, depending on the location of ± 1 . Note that among these d groups, every entry with the value 0 in $\tilde{\boldsymbol{\omega}}$ precludes one group, and only $d - (d - \Lambda') = \Lambda'$ feasible groups are preserved, as shown in the right panel of Fig. 2. Moreover, for each feasible group, $\tilde{\boldsymbol{\omega}}$ can only appear once, because of the orthogonality of different



Supplementary Fig. 2: **A visual interpretation about the appearance times of $\tilde{\omega}$ in the right-hand side of Eq. (F26).**

frequencies. Consequently, we obtain that each frequency $\tilde{\omega} \in \{0, \pm 1\}^d$ with $\|\tilde{\omega}\|_0 = \Lambda'$ appears Λ' times, indicating that the right-hand side in Eq. (F26) equals to Eq. (F27).

In other words, we achieve

$$\star = \sum_{\Lambda'=1}^d \Lambda' \sum_{\omega, \|\omega\|_0 = \Lambda'} 2^{-\|\omega\|_0} |\alpha_\omega|^2 \quad (\text{F30})$$

$$\Rightarrow \star = \sum_{\omega, \|\omega\|_0 > \Lambda} \|\omega\|_0 2^{-\|\omega\|_0} |\alpha_\omega|^2 + \sum_{\omega, \Lambda \geq \|\omega\|_0 \geq 1} \|\omega\|_0 2^{-\|\omega\|_0} |\alpha_\omega|^2 \quad (\text{F31})$$

$$\Rightarrow \star \geq \sum_{\omega, \|\omega\|_0 > \Lambda} \|\omega\|_0 2^{-\|\omega\|_0} |\alpha_\omega|^2 \quad (\text{F32})$$

$$\Rightarrow \star \geq \Lambda \sum_{\omega, \|\omega\|_0 > \Lambda} 2^{-\|\omega\|_0} |\alpha_\omega|^2. \quad (\text{F33})$$

Taken together, the truncation error can be upper bounded by the averaged gradient norm of the specified circuit. $\square$

2. Estimation error of the classical learning model (Proof of Lemma F.2)

The core of the proof is to show that the state prediction model $\hat{\sigma}_n(\mathbf{x})$ in Eq. (F1) is equal to the trigonometric expansion of the truncated target quantum state $\rho_\Lambda(\mathbf{x})$ if we take the expectation over the training data, which includes the randomness from the sampled inputs $\mathbf{x}^{(i)}$ and classical shadow. With this relation in mind, we can quantify the estimation error of the proposed state prediction model $\hat{\sigma}_n(\mathbf{x})$ when calculating the expectation value of an unseen state under the specified observable O , i.e., $\mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} [|\text{Tr}(O\hat{\sigma}_n(\mathbf{x})) - \text{Tr}(O\rho_\Lambda(\mathbf{x}))|^2]$.

Proof of Lemma F.2. We first prove the equivalence between the expectation of the classical representations and the target quantum state, i.e., $\mathbb{E}_{\mathcal{T}}[\hat{\sigma}_n(\mathbf{x})] = \rho_\Lambda(\mathbf{x})$. Following the explicit form of $\hat{\sigma}_n(\mathbf{x})$ in Eq. (F1), we obtain

$$\mathbb{E}_{\mathcal{T}}[\hat{\sigma}_n(\mathbf{x})] = \frac{1}{n} \sum_{i=1}^n \mathbb{E}_{\mathbf{x}^{(i)} \sim [-\pi, \pi]^d} \left[\kappa_\Lambda(\mathbf{x}, \mathbf{x}^{(i)}) \right] \mathbb{E}_{s_1^{(\mathbf{x}^{(i)})}, \dots, s_N^{(\mathbf{x}^{(i)})}} \tilde{\rho}_1(\mathbf{x}^{(i)}) \quad (\text{F34a})$$

$$= \mathbb{E}_{\mathbf{x}^{(1)} \sim [-\pi, \pi]^d} \left[\kappa_\Lambda(\mathbf{x}, \mathbf{x}^{(1)}) \right] \rho(\mathbf{x}^{(1)}) \quad (\text{F34b})$$

$$= \mathbb{E}_{\mathbf{x}^{(1)} \sim [-\pi, \pi]^d} \sum_{\omega, \|\omega\|_0 \leq \Lambda} 2^{\|\omega\|_0} \Phi_\omega(\mathbf{x}) \Phi_\omega(\mathbf{x}^{(1)}) \rho(\mathbf{x}^{(1)}) \quad (\text{F34c})$$

$$= \sum_{\omega, \|\omega\|_0 \leq \Lambda} \Phi_\omega(\mathbf{x}) 2^{\|\omega\|_0} \mathbb{E}_{\mathbf{x}^{(1)} \sim [-\pi, \pi]^d} \Phi_\omega(\mathbf{x}^{(1)}) \rho(\mathbf{x}^{(1)}) \quad (\text{F34d})$$

$$= \sum_{\omega, \|\omega\|_0 \leq \Lambda} \Phi_\omega(\mathbf{x}) 2^{\|\omega\|_0} \frac{1}{(2\pi)^d} \int_{[-\pi, \pi]^d} \Phi_\omega(\mathbf{x}^{(1)}) \rho(\mathbf{x}^{(1)}) d^d x^{(1)} \quad (\text{F34e})$$

$$= \sum_{\omega, \|\omega\|_0 \leq \Lambda} \Phi_\omega(\mathbf{x}) 2^{\|\omega\|_0} \frac{1}{(2\pi)^d} \int_{[-\pi, \pi]^d} \Phi_\omega(\mathbf{x}^{(1)}) \sum_{\omega'} \Phi_{\omega'}(\mathbf{x}) \rho_\omega d^d x^{(1)} \quad (\text{F34f})$$

$$= \sum_{\omega, \|\omega\|_0 \leq \Lambda} \Phi_{\omega}(\mathbf{x}) 2^{\|\omega\|_0} \sum_{\omega'} \frac{1}{(2\pi)^d} \int_{[-\pi, \pi]^d} \Phi_{\omega}(\mathbf{x}^{(1)}) \Phi_{\omega'}(\mathbf{x}^{(1)}) \rho_{\omega} d^m x^{(1)} \quad (\text{F34g})$$

$$= \sum_{\omega, \|\omega\|_0 \leq \Lambda} \Phi_{\omega}(\mathbf{x}) \rho_{\omega} \quad (\text{F34h})$$

$$= \rho_{\Lambda}(\mathbf{x}), \quad (\text{F34i})$$

where $s_j^{(\mathbf{x}^{(i)})}$ denotes the randomized measurement outcome for the j -th qubit for the state $\rho(\mathbf{x}^{(i)})$ with $s_j^{(\mathbf{x}^{(i)})} \in \{|0\rangle, |1\rangle, |\pm\rangle, |\pm i\rangle\}$, Eq. (F34b) uses the fact that each $\mathbf{x}^{(i)}$ is sampled independently and uniformly from $[-\pi, \pi]^d$, Eq. (F34c) employs the explicit formula of the truncated trigonometric monomial kernel κ_{Λ} , Eq. (F34f) adopts trigonometric expansion of the quantum state, i.e., $\rho(\mathbf{x}^{(i)}) = \sum_{\omega} \Phi_{\omega}(\mathbf{x}^{(i)}) \rho_{\omega}$, and Eq. (F34h) comes from the orthogonality of basis functions in Eq. (F13).

To complete the proof, we next move to analyze the estimation error $\mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} [|\text{Tr}(O\hat{\sigma}_n(\mathbf{x})) - \text{Tr}(O\rho_{\Lambda}(\mathbf{x}))|^2]$. Define

$$\tilde{A}_{\omega} = \frac{1}{n} \sum_{i=1}^n 2^{\|\omega\|_0} \Phi_{\omega}(\mathbf{x}^{(i)}) \text{Tr}(\tilde{\rho}_1(\mathbf{x}^{(i)})O) - \text{Tr}(\rho_{\omega}O). \quad (\text{F35})$$

By making use of the trigonometric expansion of ρ_{Λ} and the explicit formalism of $\hat{\sigma}_n(\mathbf{x})$, the estimation error can be reformulated as

$$\mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} [|\text{Tr}(O\hat{\sigma}_n(\mathbf{x})) - \text{Tr}(O\rho_{\Lambda}(\mathbf{x}))|^2] \quad (\text{F36a})$$

$$= \mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} \left[\left| \frac{1}{n} \sum_{i=1}^n \kappa_{\Lambda}(\mathbf{x}, \mathbf{x}^{(i)}) \text{Tr}(\tilde{\rho}_1(\mathbf{x}^{(i)})O) - \sum_{\omega, \|\omega\| \leq \Lambda} \Phi_{\omega}(\mathbf{x}) \text{Tr}(\rho_{\omega}O) \right|^2 \right] \quad (\text{F36b})$$

$$= \mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} \left[\left| \sum_{\omega, \|\omega\| \leq \Lambda} \Phi_{\omega}(\mathbf{x}) \left(\frac{1}{n} \sum_{i=1}^n 2^{\|\omega\|_0} \Phi_{\omega}(\mathbf{x}^{(i)}) \text{Tr}(\tilde{\rho}_1(\mathbf{x}^{(i)})O) - \text{Tr}(\rho_{\omega}O) \right) \right|^2 \right] \quad (\text{F36c})$$

$$= \mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} \left[\sum_{\omega, \|\omega\| \leq \Lambda} \sum_{\omega', \|\omega'\| \leq \Lambda} \Phi_{\omega}(\mathbf{x}) \Phi_{\omega'}(\mathbf{x}) \tilde{A}_{\omega} \tilde{A}_{\omega'} \right] \quad (\text{F36d})$$

$$= \sum_{\omega, \|\omega\| \leq \Lambda} \sum_{\omega', \|\omega'\| \leq \Lambda} \frac{1}{(2\pi)^d} \int_{[-\pi, \pi]^d} \Phi_{\omega}(\mathbf{x}) \Phi_{\omega'}(\mathbf{x}) d^d x \tilde{A}_{\omega} \tilde{A}_{\omega'} \quad (\text{F36e})$$

$$= \sum_{\omega, \|\omega\| \leq \Lambda} 2^{-\|\omega\|_0} \tilde{A}_{\omega}^2 \quad (\text{F36f})$$

$$= \sum_{\omega, \|\omega\| \leq \Lambda} 2^{-\|\omega\|_0} \left| \frac{1}{n} \sum_{i=1}^n 2^{\|\omega\|_0} \Phi_{\omega}(\mathbf{x}^{(i)}) \text{Tr}(\tilde{\rho}_1(\mathbf{x}^{(i)})O) - \text{Tr}(\rho_{\omega}O) \right|^2 \quad (\text{F36g})$$

$$\equiv \sum_{\omega, \|\omega\| \leq \Lambda} |\tilde{D}_{\omega}(\mathcal{T})|^2, \quad (\text{F36h})$$

where Eq. (F36d) adopts the explicit form of the trigonometric monomial kernel κ_{Λ} , Eq. (F36d) employs the definition $\tilde{A}_{\omega}$ in Eq. (F35), and Eq. (F36f) comes from the evaluation of the orthogonality of basis functions in Eq. (F13).

Such reformulation suggests that the derivation of the upper bound of $\mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} [|\text{Tr}(O\hat{\sigma}_n(\mathbf{x})) - \text{Tr}(O\rho(\mathbf{x}))|^2]$ is reduced to deriving the upper bound of $\tilde{D}_{\omega}(\mathcal{T})$ for $\forall \omega \in \mathfrak{C}(\Lambda)$. To do so, we rewrite the term $\text{Tr}(\rho_{\omega}O)$ in $\tilde{D}_{\omega}(\mathcal{T})$ as the trigonometric monomial expansion of $\rho(\mathbf{x})$, i.e.,

$$\text{Tr}(\rho_{\omega}O) = 2^{\|\omega\|_0} \frac{1}{(2\pi)^d} \int_{[-\pi, \pi]^d} \Phi_{\omega}(\mathbf{x}) \text{Tr}(\rho(\mathbf{x})O) d^d x \quad (\text{F37a})$$

$$= 2^{\|\omega\|_0} \mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} \Phi_{\omega}(\mathbf{x}) \text{Tr}(\rho(\mathbf{x})O) \quad (\text{F37b})$$

$$= 2^{\|\omega\|_0} \mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} \Phi_{\omega}(\mathbf{x}) \mathbb{E}_{s_1^{\mathbf{x}}, \dots, s_N^{\mathbf{x}}} \text{Tr}(\tilde{\rho}_1(\mathbf{x})O). \quad (\text{F37c})$$

Accordingly, the quantity $|\tilde{D}_\omega(\mathcal{T})|^2$ yields

$$|\tilde{D}_\omega(\mathcal{T})|^2 = \left| \frac{1}{n} \sum_{i=1}^n \Phi_\omega(\mathbf{x}^{(i)}) \text{Tr}(\tilde{\rho}_1(\mathbf{x}^{(i)})O) - \mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} \Phi_\omega(\mathbf{x}) \mathbb{E}_{s_1^\pi, \dots, s_N^\pi} \text{Tr}(\tilde{\rho}_1(\mathbf{x})O) \right|^2. \quad (\text{F38})$$

This formalism hints that we can use Hoeffding's inequality to bound $\tilde{D}_\omega(\mathcal{T})$. Recall that the requirement of applying Hoeffding's inequality is ensuring the expectation value is bounded. In our case, we have

$$|\Phi_\omega(\mathbf{x}) \text{Tr}(\tilde{\rho}_1(\mathbf{x})O)| \leq |\text{Tr}(\tilde{\rho}_1(\mathbf{x})O)| \quad (\text{F39a})$$

$$\leq \|O\|_\infty \|\tilde{\rho}_1(\mathbf{x})\|_1 \quad (\text{F39b})$$

$$= 3^K B, \quad (\text{F39c})$$

where the first inequality uses the Cauchy-Schwarz inequality and $|\Phi_\omega(\mathbf{x})| \leq 1$, the second inequality adopts the von Neumann's trace inequality with Hölder's inequality, and the last inequality exploits $\|\tilde{\rho}_1(\mathbf{x})\|_1 \leq 3^K$ [19, Eq. (F37)] and the condition $\|O\|_\infty \leq B$.

The bounded expectation term enables us to use Hoeffding's inequality to attain the following result, i.e.,

$$\Pr \left[\tilde{D}_\omega(\mathcal{T})^2 \geq \tau^2 \right] = \Pr \left[\tilde{D}_\omega(\mathcal{T}) \geq \tau \right] \leq 2 \exp \left(-\frac{2n\tau^2}{B^2 9^K} \right). \quad (\text{F40})$$

Denote the set of truncated frequencies as $\mathfrak{C}(\Lambda) = \{\omega | \omega \in \{0, \pm 1\}^d, \text{ s.t. } \|\omega\|_0 \leq \Lambda\}$. This leads to

$$\Pr \left[\sum_{\omega \in \mathfrak{C}(\Lambda)} |D_\omega(\mathcal{T})|^2 \geq |\mathfrak{C}(\Lambda)| \tau^2 \right] \leq \sum_{\omega \in \mathfrak{C}(\Lambda)} \Pr \left[|D_\omega(\mathcal{T})|^2 \geq \tau^2 \right] \leq |\mathfrak{C}(\Lambda)| \cdot 2 \exp \left(-\frac{2n\tau^2}{B^2 9^K} \right). \quad (\text{F41})$$

Let the right-hand side be δ . We have

$$\tau = \sqrt{\frac{1}{2n} B^2 9^K \log \left(\frac{2 \cdot |\mathfrak{C}(\Lambda)|}{\delta} \right)}. \quad (\text{F42})$$

This concludes the proof as with probability at least $1 - \delta$, the mean-square error between the prediction and the ground truth taken over the randomness of the sampled inputs and classical shadow is upper bounded by

$$\mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} \left[\left| \text{Tr}(O\sigma_n^{(1)}(\mathbf{x})) - \text{Tr}(O\rho(\mathbf{x})) \right|^2 \right] \leq |\mathfrak{C}(\Lambda)| \frac{1}{2n} B^2 9^K \log \left(\frac{2 \cdot |\mathfrak{C}(\Lambda)|}{\delta} \right). \quad (\text{F43})$$

□

SM G: Computational time for training and prediction

Here we analyze the computational cost of the proposed ML model. For clarity, here we separately analyze the computational time of our proposal required in the training and inference.

Training time. Recall that the training procedure of our model amounts to loading the collected training dataset $\mathcal{T}_s = \{\mathbf{x}^{(i)}, \tilde{\rho}_T(\mathbf{x}^{(i)})\}_{i=1}^n$ to the classical memory. According to the explanation of classical shadow in SM A 1, the required computation cost to store and load the N -qubit state $\tilde{\rho}_T(\mathbf{x}^{(i)})$ with T snapshots is $\mathcal{O}(NT)$. Simultaneously, the computation cost to store and load the classical input $\mathbf{x}^{(i)}$ is $\mathcal{O}(d)$. Combining these facts with the result of Theorem 2 such that the required total number of training examples n of our model, the computation cost to load the dataset $\mathcal{T}_s$ is

$$\mathcal{O}(nNT) = \mathcal{O} \left(NT |\mathfrak{C}(\Lambda)| \frac{B^2 9^K}{\epsilon} \log \left(\frac{2 \cdot |\mathfrak{C}(\Lambda)|}{\delta} \right) \right) = \tilde{\mathcal{O}} \left(\frac{B^2 9^K NT |\mathfrak{C}(4C/\epsilon)|}{\epsilon} \right). \quad (\text{G1})$$

As discussed in the main text, when C is bounded (i.e., $C \sim \mathcal{O}(1/\text{poly}(N))$ or $C \sim \mathcal{O}(1/\exp(N))$) or d is a small constant in many practical scenarios, the cardinality $|\mathfrak{C}(4C/\epsilon)|$ polynomially scales with N and d , and thus training our model is computationally efficient.

Inference (prediction) time. Suppose that the observable O is constituted by multiple local observables with a bounded norm, i.e., $O = \sum_{i=1}^q O_i$ and $\sum_l \|O_i\| \leq B$, and the maximum locality of $\{O_i\}$ is K . Following the definition of our model in Eq. (F1), given a new input $\mathbf{x}$ and the observable O , the prediction yields

$$h_s \equiv \text{Tr}(O\hat{\sigma}_n(\mathbf{x})) = \frac{1}{n} \sum_{i=1}^n \kappa_\Lambda(\mathbf{x}, \mathbf{x}^{(i)}) \text{Tr}(O\tilde{\rho}_T(\mathbf{x}^{(i)})). \quad (\text{G2})$$

In this regard, the evaluation involves summing over the assessment of each training example $(\mathbf{x}^{(i)}, \tilde{\rho}_T(\mathbf{x}^{(i)}))$ for $\forall i \in [n]$, and the evaluation of each training example can be further decomposed into two components. That is, the first component is classically computing the shadow estimation $\text{Tr}(O\tilde{\rho}_T(\mathbf{x}^{(i)}))$; the second component is calculating the kernel function $\kappa_\Lambda(\mathbf{x}, \mathbf{x}^{(i)})$ for $\forall i \in [n]$. According to the runtime complexity of shadow estimation elucidated in SM A 1, when $K \sim \mathcal{O}(1)$, the computation of each $\text{Tr}(O_i\tilde{\rho}_T(\mathbf{x}^{(i)}))$ can be completed in $\mathcal{O}(T)$ time after storing the classical shadow in the classical memory. In this regard, the computation cost to complete the first part is

$$\mathcal{O}(Tq). \quad (\text{G3})$$

Moreover, based on the explicit definition of the kernel $\kappa_\Lambda(\cdot, \cdot)$, the computation cost of evaluating $\kappa_\Lambda(\mathbf{x}, \mathbf{x}^{(i)})$ is

$$\mathcal{O}(|\mathfrak{C}(4C/\epsilon)|). \quad (\text{G4})$$

In conjunction with the computation cost of each example and the total number of training examples n , the required predicting time of our proposal is

$$\mathcal{O}(n(Tq + |\mathfrak{C}(4C/\epsilon)|)) \leq \tilde{\mathcal{O}}\left(\frac{TqB^{2gK}|\mathfrak{C}(4C/\epsilon)|^2}{\epsilon}\right). \quad (\text{G5})$$

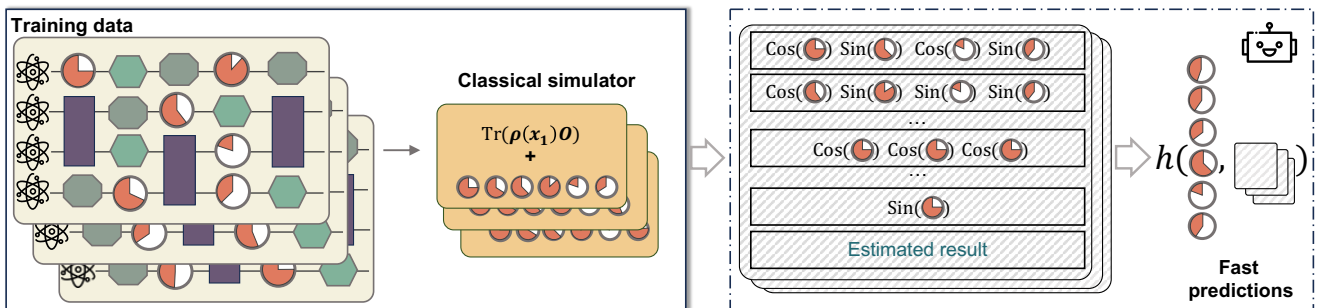
In conclusion, when C is bounded, the whole procedure of our proposal (encompassing the training and predicting) is both computational and memory efficient, which is upper bounded by

$$\tilde{\mathcal{O}}\left(\frac{TNqB^{2gK}|\mathfrak{C}(4C/\epsilon)|^2}{\epsilon}\right). \quad (\text{G6})$$

SM H: Classical prediction model with the full expansion (Proof of Corollary 1)

This section comprises two parts. In SM H 1, we elaborate on the implementation details of the protocol introduced in Corollary 1. Then, in SM H 2, we present the proof details of Corollary 1.

1. Implementation of the classical prediction model



Supplementary Fig. 3: **Purely classical learning model with the full expansion.** The pipeline is similar to the one presented in Fig. 1 of the main text. The only difference is the way of collecting training data, where the circuit layout is known by the learner and the expectation value $\text{Tr}(\rho(bx)O)$ should be efficiently calculated by the employed classical simulators.

The implementation of the protocol introduced in Corollary 1 is visualized in Fig. 3. The only difference with the one introduced in the main text is the way of collecting training data. Specifically, when the tunable quantum circuit

architecture formed by RZ + CI is known to the learner, the classical simulators (e.g., tensor network simulators, LOWESA, and near Clifford circuits simulators) are employed to collect the training data

$$\mathcal{T}_c = \left\{ \mathbf{x}^{(i)} \rightarrow \text{Tr}(\rho(\mathbf{x}^{(i)})O) \right\}_{i=1}^n. \quad (\text{H1})$$

Once the training dataset is prepared, the learner uses it to form the kernel-based prediction model, i.e., given a new input $\mathbf{x}$ and an observable O , it takes the form as

$$h_c(\mathbf{x}, O) = \frac{1}{n} \sum_{i=1}^n \kappa(\mathbf{x}, \mathbf{x}^{(i)}) \text{Tr}(\rho(\mathbf{x}^{(i)})O) \text{ with } \kappa(\mathbf{x}, \mathbf{x}^{(i)}) = \sum_{\omega, \|\omega\|_0 \leq d} 2^{\|\omega\|_0} \Phi_\omega(\mathbf{x}) \Phi_\omega(\mathbf{x}^{(i)}) \in \mathbb{R}. \quad (\text{H2})$$

Note that this protocol is relatively more restrictive than the one introduced in the main text, as the dimension d is required to be sufficiently low with the full expansion and the expectation value under the observable O should be efficiently calculated by the exploited classical simulators.

2. Proof of Corollary 1

The formal statement of Corollary 1 is as follows.

Corollary (Restatement of Corollary 1). *Following notations in the main text, consider a parametrized family of N -qubit states $\mathcal{Q}$ and $\|O\|_\infty \leq B$. Then, let the dataset be $\mathcal{T}_c = \{\mathbf{x}^{(i)} \rightarrow \text{Tr}(\rho(\mathbf{x}^{(i)})O)\}_{i=1}^n$ with $\mathbf{x}^{(i)} \sim \text{Unif}[-\pi, \pi]^d$ and $n = 3^d B^2 \log\left(\frac{2 \cdot 3^d}{\delta}\right) (2\epsilon)^{-1}$. Then with probability at least $1 - \delta$, the average prediction error of the prediction model in Eq. (H2) satisfies*

$$\mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} |h_c(\mathbf{x}, O) - \text{Tr}(O\rho(\mathbf{x}))|^2 \leq \epsilon. \quad (\text{H3})$$

Proof of Corollary 1. Since the full expansion is adopted, the truncation error is eliminated. We only need to quantify the estimation error induced by the finite training examples. The discrepancy can be reformulated as

$$\mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} \left[|h_c(\mathbf{x}, O) - \text{Tr}(O\rho(\mathbf{x}))|^2 \right] \quad (\text{H4a})$$

$$= \mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} \left[\left| \sum_{\omega} \Phi_{\omega}(\mathbf{x}) \left(\frac{1}{n} \sum_{i=1}^n 2^{\|\omega\|_0} \Phi_{\omega}(\mathbf{x}^{(i)}) \text{Tr}(\rho(\mathbf{x}^{(i)})O) - \text{Tr}(\rho_{\omega}O) \right) \right|^2 \right] \quad (\text{H4b})$$

$$= \mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} \left[\sum_{\omega} \sum_{\omega'} \Phi_{\omega}(\mathbf{x}) \Phi_{\omega'}(\mathbf{x}) A_{\omega} A_{\omega'} \right] \quad (\text{H4c})$$

$$= \sum_{\omega} \sum_{\omega'} \frac{1}{(2\pi)^d} \int_{[-\pi, \pi]^d} \Phi_{\omega}(\mathbf{x}) \Phi_{\omega'}(\mathbf{x}) d^d x A_{\omega} A_{\omega'} \quad (\text{H4d})$$

$$= \sum_{\omega} 2^{-\|\omega\|_0} \Phi_{\omega}(\mathbf{x}) A_{\omega}^2 \quad (\text{H4e})$$

$$= \sum_{\omega} 2^{-\|\omega\|_0} \left| \frac{1}{n} \sum_{i=1}^n 2^{\|\omega\|_0} \Phi_{\omega}(\mathbf{x}^{(i)}) \text{Tr}(\rho(\mathbf{x}^{(i)})O) - \text{Tr}(\rho_{\omega}O) \right|^2 \quad (\text{H4f})$$

$$\equiv \sum_{\omega} |D_{\omega}(\mathcal{T})|^2, \quad (\text{H4g})$$

where Eq. (H4f) comes from the evaluation of the orthogonality of basis functions in Eq. (F13). Recall the analysis in the proof of Theorem 1, the quantity $|D_{\omega}(\mathcal{T})|^2$ yields

$$|D_{\omega}(\mathcal{T})|^2 = \left| \frac{1}{n} \sum_{i=1}^n \Phi_{\omega}(\mathbf{x}^{(i)}) \text{Tr}(\rho(\mathbf{x}^{(i)})O) - \mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} \Phi_{\omega}(\mathbf{x}) \text{Tr}(\rho(\mathbf{x})O) \right|^2. \quad (\text{H5})$$

This formalism hints that we can use Hoeffding's inequality to bound $D_{\omega}(\mathcal{T})$. In this case, we have

$$|\Phi_{\omega}(\mathbf{x}) \text{Tr}(\rho(\mathbf{x})O)| \leq |\text{Tr}(\rho(\mathbf{x})O)| \leq \|O\|_{\infty} \|\rho(\mathbf{x})\|_1 = B, \quad (\text{H6})$$

where the first inequality uses the Cauchy–Schwarz inequality and $|\Phi_{\omega}(\mathbf{x})| \leq 1$, the second inequality adopts von Neumann’s trace inequality with Hölder’s inequality, and the last inequality exploits $\|\rho(\mathbf{x})\|_1 = 1$ and $\|O\|_{\infty} \leq B$.

The bounded expectation term enables us to use Hoeffding’s inequality to attain the following result, i.e.,

$$\Pr[D_{\omega}(\mathcal{T})^2 \geq \tau^2] = \Pr[D_{\omega}(\mathcal{T}) \geq \tau] \leq 2 \exp\left(-\frac{2n\tau^2}{B^2}\right). \quad (\text{H7})$$

This leads to

$$\Pr\left[\sum_{\omega \in \mathfrak{C}(d)} |D_{\omega}(\mathcal{T})|^2 \geq |\mathfrak{C}(d)|\tau^2\right] \leq \sum_{\omega \in \mathfrak{C}(d)} \Pr[|D_{\omega}(\mathcal{T})|^2 \geq \tau^2] \leq 3^d \cdot 2 \exp\left(-\frac{2n\tau^2}{B^2}\right). \quad (\text{H8})$$

Let the right-hand side be δ . We have

$$\tau = \sqrt{\frac{1}{2n} B^2 \log\left(\frac{2 \cdot 3^d}{\delta}\right)}. \quad (\text{H9})$$

This concludes the proof as with probability at least $1 - \delta$, the mean-square error between the prediction and the ground truth taken over the randomness of sampled inputs and classical shadow is upper bounded by

$$\mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} \left[\left| \text{Tr}(O \hat{\sigma}_n^{(c)}(\mathbf{x})) - \text{Tr}(O \rho(\mathbf{x})) \right|^2 \right] \leq 3^d \frac{1}{2n} B^2 \log\left(\frac{2 \cdot 3^d}{\delta}\right). \quad (\text{H10})$$

Let the right-hand side be equal to the tolerant error ϵ . We have

$$n = \frac{3^d B^2 \log\left(\frac{2 \cdot 3^d}{\delta}\right)}{2\epsilon}. \quad (\text{H11})$$

□

SM I: Learning bounded-gate quantum circuit with CI gates and parameterized multi-qubit gates

In the main text, we primarily elaborate on the application of the proposed ML model in predicting bounded-gate quantum circuits consisting of RZ and CI gates. However, our proposal is adaptable and can be readily extended to predict bounded-gate quantum circuits composed of alternative basis gate sets. To exemplify this flexibility, in this section, we illustrate how our proposal and the associated theoretical findings (Theorem 2) can be effectively expanded to encompass a wider context, specifically bounded-gate circuits incorporating CI gates alongside parameterized multi-qubit gates generated by arbitrary Pauli strings.

For ease of notations, given an N -qubit quantum circuit, we define the j -th parameterized gate generated by the Pauli string $P_{\mathbf{a}_j} \in \{\mathbb{I}, X, Y, Z\}^N$ with $\mathbf{a}_j \in \{0, 1, 2, 3\}^N$ as

$$\text{RP}(\mathbf{x}_j) = \exp\left(-i \frac{\mathbf{x}_j}{2} P\right) \equiv \cos\left(\frac{\mathbf{x}_j}{2}\right) \mathbb{I}_{2^N} + i \sin\left(\frac{\mathbf{x}_j}{2}\right) P. \quad (\text{I1})$$

Then the bounded-gate quantum circuit takes the form as

$$U(\mathbf{x}) = \prod_{j=1}^d (\text{RP}_{\mathbf{a}_j}(\mathbf{x}_j) u_e), \quad (\text{I2})$$

where $u_e \in \text{CI}$ refers to an arbitrary Clifford gate or their combinations. In what follows, we demonstrate how the proposed ML model effectively predicts the incoherent dynamics of bounded-gate circuits in Eq. (I1), with the resulting prediction error aligning with the findings outlined in Theorem 2.

Recall the implementation of the proposed ML model and the proof of Theorem 2 presented in SM F. In brief, if the pre-measured state under the Pauli-basis expansion can be expressed as the form

$$\rho(\mathbf{x}) = U(\mathbf{x}) (|0\rangle \langle 0|)^{\otimes N} U(\mathbf{x})^\dagger = \sum_{\omega} \Phi_{\omega}(\mathbf{x}) \langle 0| \mathfrak{U}_{\omega}^\dagger \equiv \sum_{\omega} \Phi_{\omega}(\mathbf{x}) \rho_{\omega}, \quad (\text{I3})$$

where the trigonometric monomial basis is $\Phi_{\omega}(\mathbf{x}) = \prod_{i=1}^d \begin{cases} 1 & \text{if } \omega_i = 0 \\ \cos(\mathbf{x}_i) & \text{if } \omega_i = 1 \\ \sin(\mathbf{x}_i) & \text{if } \omega_i = -1 \end{cases}$, then we can form the state prediction model in Eq. (F1), i.e.,

$$\hat{\sigma}_n(\mathbf{x}) = \frac{1}{n} \sum_{i=1}^n \kappa(\mathbf{x}, \mathbf{x}^{(i)}) \tilde{\rho}_T(\mathbf{x}^{(i)}), \quad \text{with } \kappa(\mathbf{x}, \mathbf{x}^{(i)}) = \sum_{\mathbf{k} \in \mathbb{Z}^d, \|\mathbf{k}\|_2 \leq \Lambda} \cos(\pi \mathbf{k} \cdot (\mathbf{x} - \mathbf{x}^{(i)})) \in \mathbb{R}, \quad (\text{I4})$$

and the analysis in Theorem 2 applies. In this respect, the key aspect of extending our proposal from $\{\text{RZ} + \text{CI}\}$ gate set to $\{\text{RP}_{\mathbf{a}_j} + \text{CI}\}$ gate set involves demonstrating that each parameterized gate $\text{RP}_{\mathbf{a}_j}$ with $j \in [d]$ can be represented in the form specified by Eq. (A10), i.e.,

$$\mathfrak{R}_{\text{P}_{\mathbf{a}_j}}(\mathbf{x}_j) = D'_0 + \cos(\mathbf{x}_j) D'_1 + \sin(\mathbf{x}_j) D'_2, \quad (\text{I5})$$

where D'_0 , D'_1 , and D'_2 are three constant matrices with the size $2^N \times 2^N$.

We next prove that the form in Eq. (I5) is satisfied for any Pauli string $P_{\mathbf{a}_j}$. For clarity, in the following analysis, we denote $P_{\mathbf{a}_j}$ and $\text{RP}_{\mathbf{a}_j}(\mathbf{x}_j)$ as P_j and $\text{RP}_j(x)$, respectively. According to the definition of PTM, we have

$$[\mathfrak{R}_{\text{P}_{\mathbf{a}_j}}(\mathbf{x}_j)]_{ik} = \text{Tr}(\text{RP}_j(x) P_i \text{RP}_j(-x) P_k). \quad (\text{I6})$$

Since P_i , P_j , and P_k are Pauli strings, any two of them must commute or anti-commute to each other. By exploiting this property, we next separately analyze the value of $[\mathfrak{R}_{\text{P}_{\mathbf{a}_j}}(\mathbf{x}_j)]_{ik}$ for $\forall i, k \in [4^d]$.

Case I: $[P_i, P_j] = 0$ or $[P_k, P_j] = 0$. In this case, Eq. (I6) can be simplified to

$$[\mathfrak{R}_{\text{P}_{\mathbf{a}_j}}(\mathbf{x}_j)]_{ik} = \text{Tr}(P_i P_k) = \delta_{ij}. \quad (\text{I7})$$

In other words, for the indices $i, k \in [4^d]$ whose corresponding Pauli strings commute with P_j , the corresponding PTM entry is a constant, which is zero if $i \neq k$ and 1 if $i = k$. According to the formula in Eq. (I5), the relevant entries, whose indices satisfy $[P_i, P_j] = 0$ or $[P_k, P_j] = 0$, belong to D'_0 .

Case II: $\{P_i, P_j\} = 0$ or $\{P_k, P_j\} = 0$. In this case, Eq. (I6) can be simplified to

$$[\mathfrak{R}_{\text{P}_{\mathbf{a}_j}}(\mathbf{x}_j)]_{ik} = \text{Tr}(\text{RP}_j(2x) P_i P_k) = \cos(x) \text{Tr}(P_i P_k) + \imath \sin(x) \text{Tr}(P_j P_i P_k) = \cos(x) \delta_{ij} + \imath \sin(x) \text{Tr}(P_j P_i P_k), \quad (\text{I8})$$

where the second equality employs $\text{RP}_j(2x) = \cos(x) \mathbb{I}_{2^N} + \imath \sin(x) P_j$. In other words, for the indices $i, k \in [4^d]$ whose corresponding Pauli strings anticommute with P_j , (i) when $i = k$, the corresponding PTM entry only contributes to D'_1 in Eq. (I5); (ii) when $i \neq k$ and $P_j P_i P_k = \mathbb{I}_{2^N}$, the corresponding PTM entry only contributes to D'_2 in Eq. (I5); (iii) else, the corresponding PTM entry is zero, which only contributes to D'_0 .

Taken together, for any pair indices (i, k) , the corresponding entry only contributes to one of D'_0 , D'_1 , and D'_2 , indicating the relation in Eq. (I5). As a result, our proposal and Theorem 2 applies to the basis gate set $\{\text{RP}_{\mathbf{a}_j}, \text{CI}\}$.

SM J: Connection and differences with Ref. [1] and Pauli path simulators

In this section, we detail how our work relates to and differs from Ref. [1] and Pauli path simulators. Specifically, in Subsection J1, we discuss the complementary relationship between our work and Ref. [1], highlighting how the combination of these studies enriches quantum learning theory and deepens our understanding of the learnability of bounded-gate quantum circuits. In SM J2, we clarify the distinctions between the proposed learning model and Pauli path simulators, addressing differences at both the technical and application levels.

1. Complementation with Ref. [1]

For comprehensive, we summarize the theoretical results achieved by Ref. [1] and our work in Table 2. Particularly, Ref. [1] investigates the computational complexities of quantum learners across three learning tasks:

- Task (i) learning bounded-gate unitaries;
- Task (ii) learning bounded-gate quantum states;

- Task (iii) learning $\mathbb{R}$ -valued classical functions with 1-Lipschitz continuity.

Different from Ref. [1], we explore the computational complexities of classical learners for a new task:

- Task (iv) predicting the expectation values of bounded-gate quantum circuit for a given observable.

The achieved theoretical results under different learning tasks (i.e., Tasks (i)-(iv)) and different learning paradigms (i.e., quantum learners versus classical learners) are complementary, providing the following two crucial insights in the context of the learnability of bounded-gate quantum circuits.

Supplementary Table 2: Summarization of theoretical results achieved by Ref. [1] and our work. The terms ‘Q-learner’ and ‘C-learner’ refer to the quantum and classical learning paradigms, respectively. Besides, the terms ‘Samp’ and ‘Comp’ refer to the abbreviations of ‘sample complexity’ and ‘computational complexity’, respectively. C-function refers to the classical function with ν being the number of variables in the classical function (Task (iii)). The notations N and ϵ denote the number of qubits and the prediction error. In Ref. [1], the explored bounded-gate circuit is realized by $\hat{G}$ two-qubit gates, while in our work the explored bounded-gate circuit is composed of d RZ gates and $G - d$ Clifford gates. The notations B and T refer to the norm of the observable and the snapshots when collecting each training example, respectively. The markers $\checkmark$, $\times$, $?$ separately represent ‘yes’, ‘no’, and ‘unknown’.

	Learn unitary (Task (i))	Learn state (Task (ii))	Learn C-function (Task (iii))	Learn observable (Task (iv))
Q-learner	Theorems 4&6, Ref. [1] Samp: $\begin{cases} \tilde{O}(\hat{G} \min\{\frac{1}{\epsilon^2}, \frac{\sqrt{2N}}{\epsilon}\}) \\ \Omega(\frac{\hat{G}}{\epsilon}) \end{cases}$ Comp: $e^{\Omega(\min\{\hat{G}, N\})}$ Algorithm: $\checkmark$	Theorems 1&2, Ref. [1] Samp: $\tilde{\Theta}(\frac{\hat{G}}{\epsilon^2})$ Comp: $e^{\Omega(\min\{\hat{G}, N\})}$ Algorithm: $\checkmark$	Theorem 7, Ref [1] Samp: $\Omega(\frac{1}{\epsilon^\nu})$ Comp: $\tilde{\Omega}(\frac{1}{\epsilon^\nu})$ Algorithm: $\times$	$?$
C-learner	$?$	$?$	$\times$	Theorem 1, this work Samp: $\begin{cases} \tilde{O}(\frac{B^2 d + B^2 N G}{\epsilon^2}) \\ \tilde{\Omega}(\frac{(1-\epsilon)d}{\epsilon T}) \end{cases}$ Comp: $O(\exp(N, d))$ Algorithm: $\checkmark$ (Theorem 2)

Before moving on to discuss the two insights, let us emphasize the necessity of exploring both classical and quantum learners. Recall that Ref. [1] focuses on the quantum learner, which may implement the learning model on quantum computers (e.g., quantum neural networks) and require quantum resources during both the training and inference stages. In contrast, our work centers on the classical learner, which implements the learning model on classical computers, requiring quantum resources only during the data collection stage. This means that both the training and inference stages for classical learners are conducted entirely on classical hardware. A comprehensive analysis about the computational complexities of both quantum and classical learners can guide various practical applications and indicate the utility of quantum computers. On the one side, given the foreseeable scarcity of quantum resources, if classical learners can achieve comparable performance to quantum learners for a specific task, they offer undeniable benefits in terms of resource efficiency. On the other side, if quantum learners demonstrate provable computational advantages over classical learners for a given task, this would establish the practical utility of quantum computing for that particular application. For example, Ref. [66] provides a compelling example of how computational capabilities differ between classical and quantum learners, demonstrating that quantum learners can achieve exponential runtime speedups in predicting the properties of ground states.

a. Insight I: Computational complexity analysis

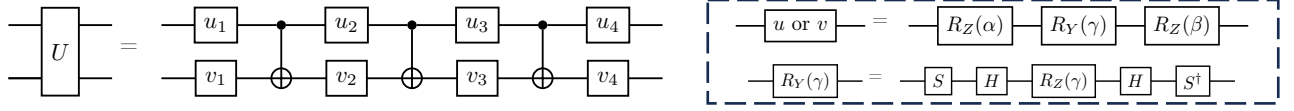
Based on the achieved computational complexities presented in Table 2, we draw the following key observations in the context of learning bounded-gate quantum circuits.

Observation I. Let us revisit the achieved sample complexity in Tasks (i), (ii), and (iv) summarized in Table 2. That is, despite the decreasing levels of information gain—from learning an unknown unitary, to an unknown state, and finally to unknown expectation values, we observe a consistent scaling behavior for both classical and quantum learners.

In particular, Theorems 1 and 4 in Ref. [1] demonstrate that when learning a circuit or a pure state composed of $\hat{G}$ two-qubit gates, $\Theta(\hat{G})$ training examples are sufficient and necessary for a quantum learner to ensure a bounded prediction error. Similarly, Theorem 1 in the main text establishes that when predicting the expectation values of

a bounded-gate circuit composed of d RZ gates and $G - d$ Clifford gates, $\Theta(d)$ training examples are sufficient and necessary for a classical learner to achieve a bounded prediction error.

We would like to note that although Ref. [1] and our work use different descriptions of bounded-gate circuits, the results achieved in our study can be directly extended to their framework. This is because our description of bounded-gate circuits can be efficiently transformed into their setting with only a constant overhead. More precisely, a bounded-gate quantum circuit composed of $\hat{G}$ two-qubit gates can be implemented using at most $d = 24\hat{G}$ RZ gates and $G - d = 35\hat{G}$ Clifford gates to implement this circuit. An intuition is provided in Fig. 4. Ref. [69] has proved that three CNOT gates, associated with eight single-qubit gates, are necessary and sufficient in order to implement an arbitrary unitary transformation of two qubits. Meanwhile, any single qubit gate can be realized by three RZ gates and four single-qubit Clifford gates [70]. Taken together, $3 \times 8 = 24$ RZ gates and $4 \times 8 + 3 = 35$ Clifford gates can implement an arbitrary unitary transformation of two qubits. Since there are $\hat{G}$ two-qubit gates, $d = 24\hat{G}$ RZ gates and $G - d = 35\hat{G}$ Clifford gates are sufficient to implement this circuit.



Supplementary Fig. 4: **Decomposition of an arbitrary two-qubit gate into RZ and Clifford gates.** The dashed box highlights the implementation of the single-qubit gates u_* and v_* with $*$ $\in \{1, 2, 3, 4\}$ by RZ gates and single-qubit Clifford gates.

According to the result of Theorem 1 in our work, when a classical learner is applied to predict the expectation value of a bounded-gate quantum circuit composed of $\hat{G}$ two-qubit gates, $\tilde{O}((B^2\hat{G} + B^2N\hat{G})/\epsilon)$ and $\tilde{\Omega}((1 - \epsilon)\hat{G}/(\epsilon T))$ training examples are sufficient and necessary to achieve a bounded prediction error. This result is consistent with the findings for Tasks (i) and (ii) achieved in Ref. [1], both of which display a linear-in- $\hat{G}$ sample complexity.

The linear scaling behavior with respect to the number of tunable gates across different learners (i.e., quantum versus classical) and tasks (i.e., Tasks (i), (ii), and (iv)) suggests a consistent linear-in- d (or linear-in- $\hat{G}$) sample complexity for classical learners in Tasks (i)&(ii) and for quantum learners in Task (iv). This conjecture is supported by two key facts. First, covering and packing nets, as well as (quantum) hypothesis selection, are robust tools for analyzing the sample complexity of learning bounded-gate quantum circuits, often providing tight bounds [19, 59, 61, 63, 71]. Second, the cardinality of these nets is intrinsically dependent on the number of gates (d , G , or $\hat{G}$), which leads to similar results across tasks, despite variations in the intended information gain.

Observation II. We next turn to rethink the achieved runtime complexities in Tasks (i), (ii), and (iv) summarized in Table 2. An immediate observation is that, for all tasks, there are no efficient *quantum learners* for learning bounded-gate unitaries or quantum states [1, Theorems 2&6]. In addition, there are no efficient *classical learners* capable of accurately predicting the mean-value space of arbitrary bounded-gate quantum circuits (Theorem 1 in the main text). These results are complementary and offer valuable insights into the unresolved issues outlined in Table 2.

First, the inherent limitations of quantum learners in learning bounded-gate unitaries or states strongly suggest that there are no efficient classical learners capable of accomplishing Tasks (i) and (ii). This is because from an information-theoretic perspective, classical learners cannot extract much more information than quantum learners. Consequently, classical learners are unlikely to outperform quantum learners in these tasks.

Second, while neither quantum nor classical learners can efficiently learn arbitrary bounded-gate quantum circuits, it is natural to ask whether specific conditions on the explored bounded-gate circuits could lead to the computational separations, i.e., quantum learners can provide efficient solutions, whereas classical learners would still face computational hardness. Answering this question can enhance our understanding of the learnability of bounded-gate quantum circuits and help identify the practical utility of quantum learners.

b. Insight II: Learning algorithm design

The computational hardness of learning bounded-gate quantum circuits in Tasks (i), (ii) and (iv) highlights the need to develop advanced learning algorithms with provable computational efficiency under practical and realistic conditions. Both Ref. [1] and our work make initial strides in this direction. Below, we summarize the advancements of each method and outline potential directions for future research.

For Tasks (i) and (ii), Ref. [1] proposes efficient learning algorithms for relatively restrictive scenarios where the number of two-qubit gates $\hat{G}$ scales logarithmically with the number of qubits N , i.e., $\hat{G} \sim \mathcal{O}(\log(N))$. Conceptually, at the inference stage, given any new quantum circuit composed of $\mathcal{O}(\log(N))$ two-qubit gates, the proposed algorithm

first identifies the non-trivial $\mathcal{O}(\log(N))$ qubits on which two-qubit gates are applied and then employs Pauli-based tomography on these qubits. The achieved result naturally raises two future research directions. The first direction is to explore whether there exist efficient learning algorithms beyond the logarithmic scaling scenario. The second research direction is to explore how to design effective classical learners that achieve performance comparable to quantum learners in Tasks (i) and (ii).

For Task (iv), we devise an efficient classical-shadow predictor based on the kernel methods. While a direct comparison may not be entirely fair, it is worth highlighting that the conditions required for provable efficiency in our approach are much less restrictive than those in Ref. [1]. Specifically, our protocol can efficiently learn a broad class of quantum circuits composed of a polynomial number of gates, as opposed to being limited to logarithmic scenarios. The achieved results, combined with the unsolved issues in Table 2, lead to two interesting questions to be further explored. The first one is to explore alternative efficient classical learning algorithms. The second one is to analyze the computational separation between quantum and classical learning algorithms for Task (iv).

In summary, the complementary roles of Ref. [1] and our work expand the landscape of learning algorithms for bounded-gate quantum circuits, paving the way for deeper exploration of their utility and limitations in quantum computing.

2. Connections and differences with Pauli path simulators

The primary distinction between our work and existing classical simulators, especially for Pauli path simulators, lies in their respective purposes. Specifically, our work focuses on developing a shadow state predictor to predict the mean values of a bounded-gate quantum circuit for new rotation angles and observables. In contrast, existing classical algorithms aim to efficiently simulate the mean values of a bounded-gate quantum circuit for varying rotation angles while keeping the observables fixed. This fundamental difference—prediction versus estimation—necessitates distinct theoretical analyses (i.e., the way of using Pauli transfer matrix (PTM) and the truncation method) and underscores unique benefits and advancements our protocol offered in many practically relevant scenarios.

Different roles of PTM. PTM is a widely used tool in quantum computing, with applications ranging from gate set tomography [72] to quantum learning theory [73] and the analysis of quantum circuit dynamics [7, 8, 74, 75]. In the context of simulating the mean values of bounded-gate quantum circuits, PTM is typically combined with Pauli path simulations to compute the contribution of each path [7, 8]. As explained in SM A2, given an N -qubit quantum circuit $U(\mathbf{x})$ is composed of d RZ gates and $G - d$ Clifford gates, the expectation value of the given observable O under the PTM representation yields

$$\text{Tr}(\rho(\mathbf{x})O) = \text{Tr}(U(\mathbf{x})(|0\rangle\langle 0|)^{\otimes N}U(\mathbf{x})^\dagger O) = \sum_{\omega} \Phi_{\omega}(\mathbf{x}) \text{Tr}(\rho_{\omega}O),$$

where the notation $\Phi_{\omega}(\mathbf{x})$ with $\omega \in \{0, \pm 1\}^d$ refers to the trigonometric monomial basis defined in Eq. (A12) and $\text{Tr}(\rho_{\omega}O)$ refers to the expectation value of the *purely-Clifford circuit* for the path indexed by ω . For Pauli path simulators, the expectation value for a portion of $\{\omega\}$ should be explicitly computed on the classical hardware.

Unlike Pauli path simulators, our proposal adopts a fundamentally different approach to leveraging PTM. Instead of computing nontrivial $\text{Tr}(\rho_{\omega}O)$ like Pauli path simulators, we utilize PTM to derive the explicit formula of the bounded-gate quantum state with respect to the input $\mathbf{x}$, i.e.,

$$\rho(\mathbf{x}) = U(\mathbf{x})(|0\rangle\langle 0|)^{\otimes N}U(\mathbf{x})^\dagger = \sum_{\omega} \Phi_{\omega}(\mathbf{x})\rho_{\omega},$$

where $\Phi_{\omega}(\mathbf{x})$ and ρ_{ω} follow the same definitions in Eq. (A13). This explicit form inspires us to design the shadow state predictor $\hat{\sigma}_n(\mathbf{x})$ in Eq. (F1). Moreover, in Appendix F2 (proof of Lemma F.2), we prove that given any new input $\mathbf{x}' \in [-\pi, \pi]^d$, the expectation value of the shadow state predictor $\hat{\sigma}_n(\mathbf{x}')$ matches the truncated target quantum state $\rho_{\Lambda}(\mathbf{x}')$, i.e., $\mathbb{E}[\hat{\sigma}_n(\mathbf{x}')] = \rho_{\Lambda}(\mathbf{x}') \equiv \sum_{\omega, \|\omega\|_0 \leq \Lambda} \Phi_{\omega}(\mathbf{x}')\rho_{\omega}$.

The above result sets our approach apart from the use of PTM in Pauli path simulators. Specifically, our work leverages PTM to derive the explicit formula of the concept class with respect to the input $\mathbf{x}$ and subsequently design the shadow state prediction model $\hat{\sigma}_n(\mathbf{x})$, whereas Pauli path simulators such as LOWESA utilize PTM to compute values for each nontrivial path.

Truncation strategy. The low-weight truncation scheme, as with PTM, is also a widely utilized strategy in quantum computing. However, the approach to truncation and the elements being truncated differ substantially across applications, leading to distinct theoretical results and heuristic performances. For Pauli path simulators, low-weight truncation is applied to the frequencies ω or the weights of Pauli operators to achieve computational efficiency while

maintaining a low estimation error. In contrast, in our work, low-weight truncation is applied to the frequencies to ensure computational efficiency while minimizing the prediction error. This fundamental distinction in purpose underscores that prior results on Pauli path simulators offer no direct implications for learning protocols, necessitating an entirely new theoretical framework.

Recall that Theorem 1 in the main text establishes that no efficient learning model can predict all bounded-gate quantum circuits. Consequently, the most critical challenge lies in identifying specific classes of bounded-gate quantum circuits that can be efficiently learned—a question that classical simulators do not need to address. To tackle this, we establish a connection between the gradient norm of bounded-gate circuits and the truncation error as indicated in Lemma F.1. This relation warrants that the proposed learning model can achieve efficient predictions with low prediction error. This novel result, which is absent in Pauli path simulators, highlights the uniqueness of our approach and opens new avenues for developing efficient learning models tailored to predicting the mean values of bounded-gate quantum circuits.

Unique benefits of our learning model in two scenarios. As explained previously, our method focuses on the shadow state prediction, while LOWESA centers on Heisenberg simulations. These differences define the unique strengths of our model in the following two scenarios.

- (i) Circuits with many Clifford gates. Classical Pauli path simulators, such as LOWESA, must explicitly simulate Clifford gates, which, although theoretically efficient ($\mathcal{O}(N^2)$ complexity with N being the qubit count), can be computationally expensive in terms of wall-clock time. In contrast, our learning model is insensitive to the number of Clifford gates, as these gates do not affect the learning process. We validate this advantage through numerical simulations in SM K 5, demonstrating superior performance in this regime.
- (ii) Circuits with unknown system noise. Classical simulators, including both tensor network simulators and Pauli path simulators, rely on assumptions about noise strength when simulating the dynamics of noisy quantum circuits. However, these assumptions generally do not align with the actual noise present in the system. In contrast, a learning-based approach (such as the extension of our model) can automatically capture and adapt to inherent system noise by leveraging the dataset collected from the target quantum system, offering a more flexible and accurate alternative.

SM K: More numerical simulations

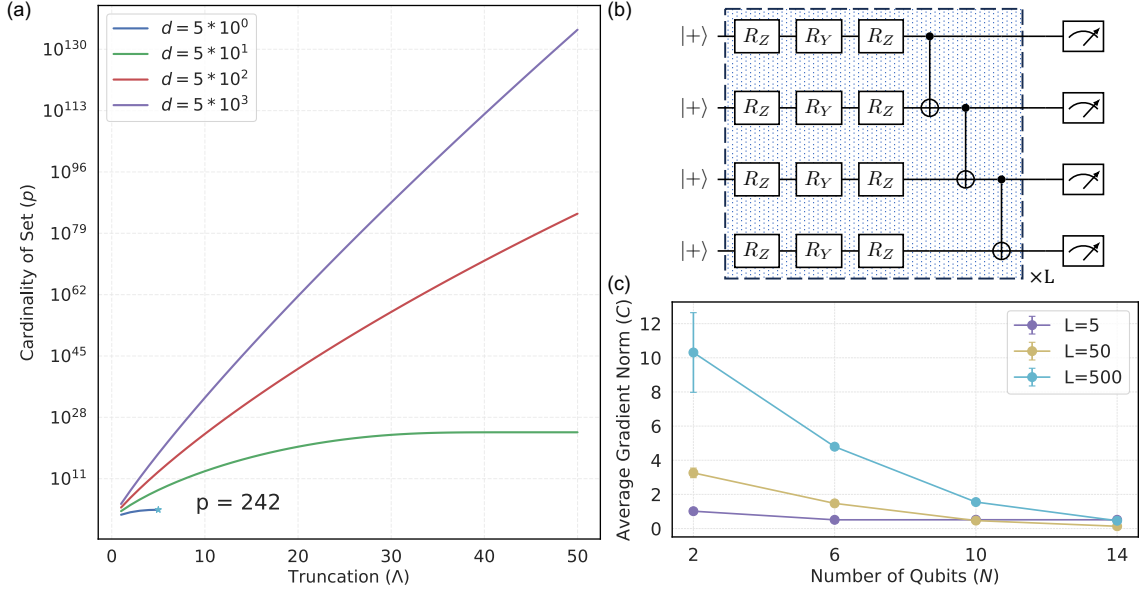
In this section, we demonstrate more simulation details omitted in the main text. Specifically, in SM K 1, we provide heuristic evidence about the required truncation value Λ in practice. In SM K 2, we provide more simulation details about predicting properties of rotational N -qubit GHZ states. Then, in SM K 3, we illustrate more simulation results about the global Hamiltonian simulation task. Next, in SM K 4, we demonstrate how to use the proposed ML model to enhance variational quantum algorithms, including variational quantum Eigen-solver and quantum neural networks. Last, in SM K 5, we numerically explore the performance of the proposed ML model over and LOWESA when processing quantum circuits with many Clifford gates.

1. Heuristic evidence about the truncation value Λ

In this subsection, we conduct a systematic analysis to heuristically understand the behavior of Λ , or equivalently $|\mathfrak{C}(4C/\epsilon)|$. The results provide insights into the conditions under which the proposed ML model can operate efficiently. The scaling behavior of $\mathfrak{C}(\Lambda)$. We begin by examining how the cardinality of the truncated frequency set $\mathfrak{C}(\Lambda)$ defined in Theorem 2 of the main text scales with the input dimensions d (i.e., the number of RZ gates) and the truncation thresholds Λ . Mathematically, we have

$$p \equiv |\mathfrak{C}(\Lambda)| = \sum_{k=0}^{\Lambda} \binom{d}{k} 2^k. \quad (\text{K1})$$

The scaling behavior is visualized in Fig. 5(a), where p exhibits two distinct phases: in Phase I, for small truncation values Λ , p scales exponentially with both Λ and the input dimension d ; in Phase II, as Λ increases further, the growth of p begins to saturate and eventually converges to a fixed value. Notably, such a scaling imposes a substantial computational burden for large d and Λ . As shown in the plot, when both d and Λ are large, the computational overhead becomes unaffordable, e.g., when $d = 50$ and $\Lambda = 10$, p is above 10^{12} .



Supplementary Fig. 5: **The problem-dependent behavior of the truncation threshold Λ .** (a) SCALING BEHAVIOR OF p . The x-axis and y-axis represent the truncation thresholds and the cardinality of the truncation set $p \equiv |\mathcal{C}(\Lambda)|$, respectively. The symbol ‘ $\star$ ’ marks that the cardinality of the truncation set is $p = 242$ when $d = 5$ and $\Lambda = 5$. (b) QUANTUM CIRCUIT FOR AVERAGE GRADIENT NORM COMPUTATION. The panel illustrates the quantum circuits used for computing the average gradient norm C when the number of qubits is $N = 4$. The dashed box highlights the implementation of hardware-efficient ansatz and the notation ‘ L ’ denotes the number of layers. (c) SCALING BEHAVIOR OF C . The simulation results of the average gradient norm C with varied qubit count N and layer number L . The vertical bar refers to the variance of the gradient norm.

The relation between Λ and the gradient norm C . The above analysis indicates that the practical efficiency of the proposed ML model lies in the small- Λ regime, especially for the large d . According to the relation between Λ and the gradient norm C in Theorem 2, this is equivalent to requesting a small value about the ratio $4C/\epsilon$. In this regard, the problem reduces to determining how the average gradient norm C scales in practice.

We perform the following numerical simulations to investigate the scaling behavior of the average gradient norm with $C = \mathbb{E}_{\mathbf{x} \sim [-\pi, \pi]^d} \|\nabla_{\mathbf{x}} \text{Tr}(\rho(\mathbf{x})O)\|_2^2$. Specifically, the class of states $\{\rho(\mathbf{x})\}$ is prepared by applying an N -qubit hardware-efficient ansatz $U(\mathbf{x})$ to an N -qubit product state $|+\rangle^{\otimes N}$ with $|+\rangle = (|0\rangle + |1\rangle)/\sqrt{2}$, i.e.,

$$\{\rho(\mathbf{x}) = U(\mathbf{x})(|+\rangle \langle +|)^{\otimes N} U(\mathbf{x})^\dagger | \mathbf{x} \in [-\pi, \pi]^d\}$$

and

$$U(\mathbf{x}) = \prod_{l=1}^L \left(\bigotimes_{j=1}^N \text{RZ}(\mathbf{x}_{j*(l-1)}) \text{RY}(\mathbf{x}_{j*(l-1)+1}) \text{RZ}(\mathbf{x}_{j*(l-1)+2}) \right) u_e, \quad (\text{K2})$$

where L denotes the number of layers, u_e refers to the entanglement layer consisting of N CNOT gates applied sequentially to adjacent qubits, and the dimension of the input yields

$$d = 3 * N * L.$$

The observable interacted with $\rho(\mathbf{x})$ is fixed to be $O = (X_1 + Y_1 + Z_1)/3$, where X_1 (or Y_1, Z_1) refers to apply the Pauli operator X (or Y, Z) to the first qubit. An illustration of the employed quantum circuit is depicted in Fig. 5(b).

The hyper-parameter settings are as follows. The number of qubits N varies from 2 to 14, and for each specified qubit count, the number of layers L is varied from 5 to 500. For each configuration, we uniformly and randomly sample the input 500 times to estimate the average gradient norm C and its variance.

The scaling behavior of the average gradient norm is illustrated in Fig. 5(c). An important observation is that, for all layer numbers L , the average gradient norm C converges to a small value with the negligible variance as the qubit count N increases. When $N = 14$, the value of C with $L=50$ is around 0.13 ± 10^{-6} . Although the concrete value of C depends on the employed ansatz $U(\mathbf{x})$ and the qubit count N , the obtained simulation results support a reasonable conjecture that $C \ll 1$ as the number of qubits increases.

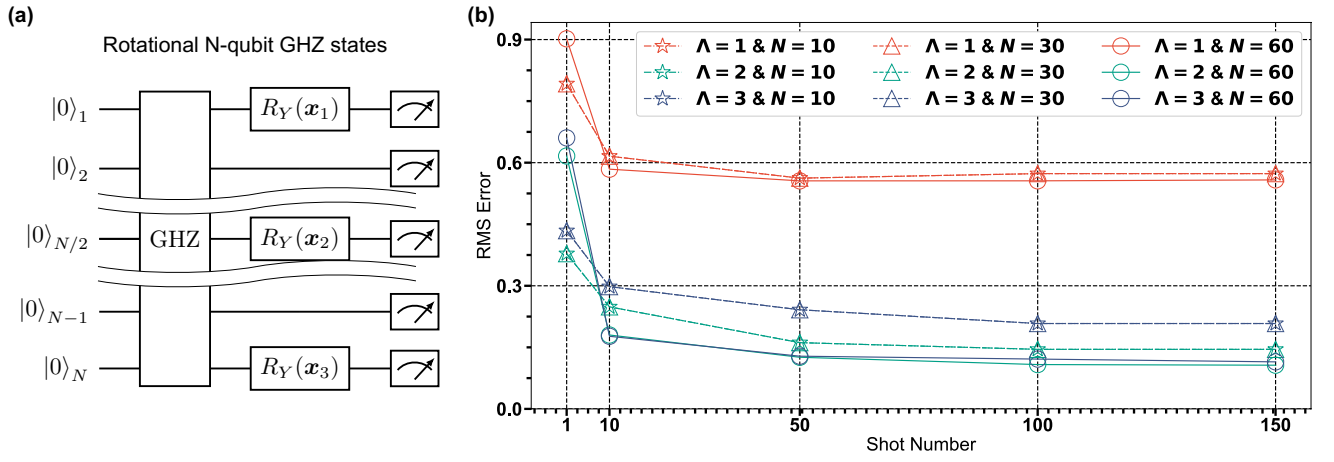
The above findings validate the potential of the proposed ML model to predict interested properties of large-qubit circuits. According to the relation $\Lambda = 4C/\epsilon$ and the scaling behavior of Λ exhibited in Fig. 5, we conclude that the primary factor influencing the applicability of the proposed ML model is the error threshold ϵ . That is, when the given task requires high-precision predictions such that $\Lambda = 4C/\epsilon$ yields a very large constant, the proposed ML model will encounter computational issues; otherwise, the proposed ML model can be efficient.

2. Numerical simulations of N -qubit rotational GHZ states

Dataset construction of N -qubit rotational GHZ states. The mathematical form of the N -qubit rotational GHZ states is

$$|\text{GHZ}(\mathbf{x})\rangle = (R_{Y_1}(\mathbf{x}_1) \otimes R_{Y_{N/2}}(\mathbf{x}_2) \otimes R_{Y_N}(\mathbf{x}_3)) \frac{|0 \cdots 0\rangle + |1 \cdots 1\rangle}{\sqrt{2}}. \quad (\text{K3})$$

The circuit implementation of this class of quantum states is visualized in Fig. 6(a).



Supplementary Fig. 6: **More simulation results of rotational N -qubit GHZ states.** (a) CIRCUIT IMPLEMENTATION OF THE ROTATIONAL N -QUBIT GHZ STATES. The three R_Y gates apply to the first qubit, the $(N/2)$ -th qubit, and the last qubit. (b) PREDICTION ERROR. The root mean squared (RMS) error of the trained ML model with varied truncation Λ , the shot number T , and the qubit count N .

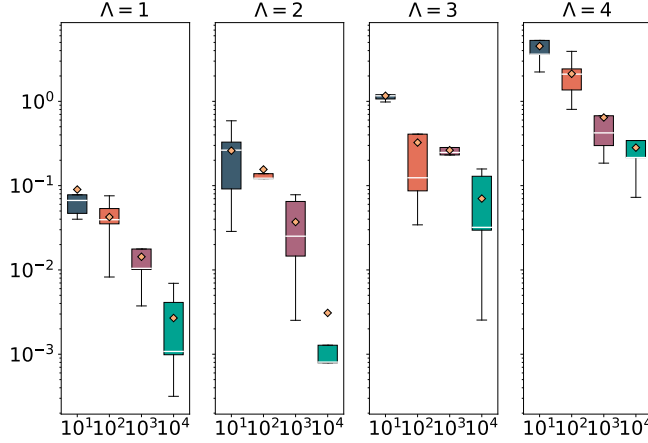
In both subtasks presented in the main text, the classical shadow of each example $(\mathbf{x}^{(i)}, \tilde{\rho}_T(\mathbf{x}^{(i)}))$ in the training dataset $\mathcal{T}$ are acquired by PastaQ library [76]. In the first subtask of two-point correlation estimation, the exact value is obtained by matrix product operators (MPO) provided by PastaQ [76]. In the second subtask of predicting expectation values on $Z_1 \otimes Z_N$, the accurate results yield

$$\begin{aligned} & \langle \text{GHZ}(\mathbf{x}) | Z_1 \otimes Z_N | \text{GHZ}(\mathbf{x}) \rangle \\ &= -\sin(\mathbf{x}_1) \sin(\mathbf{x}_2) \cos(\mathbf{x}_3) - \cos(\mathbf{x}_1) \sin(\mathbf{x}_2) \sin(\mathbf{x}_3) + \sin(\mathbf{x}_1) \cos(\mathbf{x}_2) \sin(\mathbf{x}_3) + \cos(\mathbf{x}_1) \cos(\mathbf{x}_2) \cos(\mathbf{x}_3). \end{aligned} \quad (\text{K4})$$

Hyper-parameter settings. The random seed to collect training examples and test examples is set as 1234 and 123, respectively. The hyper-parameter of MPO used to calculate the exact values of the two-point correlation of rotational GHZ states is as follows. The cutoff value is set as 10^{-8} and the max dimension is set as 50.

Cardinality of the frequency set $\mathcal{C}(\Lambda)$. In the main text, we adopt three settings of the maximum frequency length, i.e., $\Lambda = 1, 2, 3$, to evaluate the performance of the proposed ML model. The corresponding cardinality of the frequency set is $|\mathcal{C}(\Lambda = 1)| = 7$, $|\mathcal{C}(\Lambda = 2)| = 19$, and $|\mathcal{C}(\Lambda = 3)| = 27$, respectively. The similar performance between $\Lambda = 2$ and $\Lambda = 3$ (full expansion) in Fig. 2 indicates that truncating the high-frequency terms does not apparently affect the capability of the proposed ML model.

Prediction error versus varied number of qubits N . We append more simulation results about how the prediction error of the proposed ML model depends on the shot number T , the truncation Λ , and the qubit counts N . To be specific, we fix the number of training examples to be $n = 500$ and collect these training examples under different qubit count N , where the maximum shot number is set as $T = 150$. Fig. 6(b) visualizes the root mean



Supplementary Fig. 7: **Prediction error on the state evolved by 60-qubit global Hamiltonians with $d = 30$ with different sizes of the training dataset.** The meaning of notations in the box plot is as follows. The label ‘ $\Lambda = a$ ’ stands for that the truncation value is set as a . The x -axis refers to the varied number of training examples n , ranging from $n = 10$ to $n = 10^4$. The y -axis refers to the achieved RMS prediction error. All results are obtained using the same random seeds.

squared (RMS) prediction error under different settings on 10 test examples. The achieved results indicate that the performance of the proposed ML model is dominated by the truncation number Λ (i.e., the dimension of classical inputs d) and not sensitive to the number of qubits, which echoes our theoretical analysis. More precisely, when $\Lambda \geq 2$ and $T \geq 100$, the prediction error attains a very low value for both $N = 10, 30, 60$.

3. More details of synthetic global Hamiltonian simulation

Here we conduct a much more difficult task compared to the one shown in the main text. In particular, we apply the proposed ML model to predict the magnetization with $\langle \bar{Z} \rangle$ when $d = 30$. The number of training examples and the snapshot for each example are set to $n = 50000$ and $T = 50$, respectively. In this scenario, full expansion becomes computationally infeasible, where the cardinality of the frequency set is 3^{30} . The corresponding cardinality of the frequency set is $|\mathfrak{C}(\Lambda = 1)| = 61$, $|\mathfrak{C}(\Lambda = 2)| = 1801$, $|\mathfrak{C}(\Lambda = 3)| = 472761$, and respectively. When evaluating the statistical performance of the proposed ML model with $n < 50000$, we sample a subset from the whole dataset using different random seeds. The random seeds for all relevant simulations are set as 12345, 22222, 33333, 44444, and 55555, respectively.

RMS prediction error. Fig. 7 demonstrates the RMS prediction error of our proposal with the varied number of training examples, i.e., $n \in \{10, 10^2, 10^3, 10^4\}$. An immediate observation is that with the increased n , the averaged prediction error continuously decreases for all settings of Λ . In addition, the achieved results hint that the proposed model with a larger truncation value Λ may require more training examples to surpass the one with a smaller truncation value. For instance, the proposed ML model with $\Lambda = 2$ attains a better performance compared to $\Lambda = 1$ when $n = 10^4$. It is expected that the model with $\Lambda = 3, 4$ can attain better performance by further increasing n .

For completeness, we also append the corresponding standard error in Table 3. Particularly, for all settings with $\Lambda \in \{1, 2, 3, 4\}$, the averaged RMS prediction error and the corresponding standard error continuously decrease with the increased number of training examples.

Supplementary Table 3: **Simulation results of predicting the magnetization of states evolved by the 60-qubit global Hamiltonian.**

	$n = 100$	$n = 250$	$n = 500$	$n = 750$	$n = 1000$
$\Lambda = 1$	0.0424 ± 0.0222	0.0210 ± 0.0126	0.0130 ± 0.0109	0.0152 ± 0.0117	0.0143 ± 0.0088
$\Lambda = 2$	0.1556 ± 0.0959	0.1862 ± 0.0933	0.0729 ± 0.0472	0.0729 ± 0.0472	0.037 ± 0.0293
$\Lambda = 3$	0.3244 ± 0.3473	0.3736 ± 0.3249	0.4012 ± 0.2055	0.3502 ± 0.1357	0.2631 ± 0.1003
$\Lambda = 4$	2.1227 ± 1.061	0.9519 ± 0.7148	0.9678 ± 0.6590	0.7926 ± 0.4887	0.6432 ± 0.5222

The role of shot number T . Here we conduct numerical simulations to explore how the shot number T influences

the prediction error. Specifically, we fix the number of training examples as $n = 500$ and the truncation value as $\Lambda = 1$, but vary the shot number from $T = 50$ to $T = 500$. Each setting is repeated five times to collect the statistical results, where the randomness stems from distilling different training datasets from 600 training examples. The simulation results are summarized in Table. 4, where the averaged prediction error does not decrease with the increased number of measurements. This phenomenon echoes our theoretical analysis, suggesting that once the shot number exceeds a certain threshold, it does not heavily affect the performance of the proposed model.

Supplementary Table 4: **Simulation results of predicting the magnetization of states evolved by the 60-qubit global Hamiltonian with varied shot numbers.**

	$T = 50$	$T = 500$
$n = 500 \text{ \& } \Lambda = 1$	0.0130 ± 0.0109	0.0251 ± 0.0038

4. More simulation results of enhanced variational quantum algorithms by pretraining

As shown in SM A 3, a major application of the proposed ML model is enhancing variational quantum algorithms by substantially reducing the quantum resource demands. In this subsection, we provide a comprehensive explanation of the corresponding algorithmic implementation and then proceed to numerical simulations that demonstrate the efficacy of our approach.

Algorithmic implementation. Recall that contemporary quantum devices encounter limitations such as connectivity, gate fidelities, and coherence time. To overcome these constraints, experimentalists typically adopt tailored ansatzes constructed from the native gate set from the specified quantum device to implement various variational quantum algorithms [77, 78]. Essentially, these ansatzes maintain the consistent gate layouts while adjusting parameters for various computational tasks, as delineated by the framework formulated in Eq. (3) of the main text. This intrinsic relation warrants the use of the proposed ML model to enhance plenty of variational quantum algorithms associated with device-specific or task-specific ansatzes.

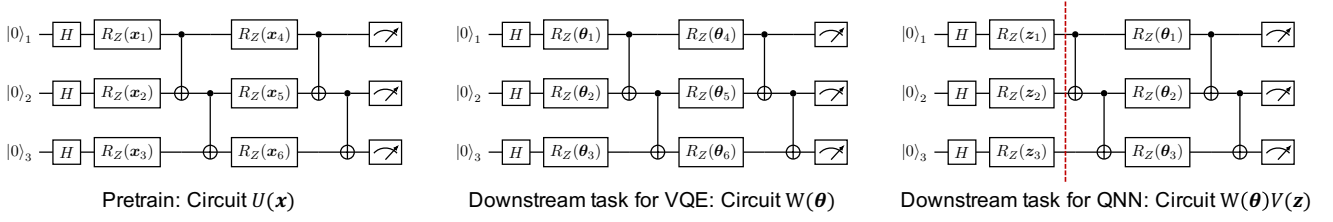
The learning framework is summarized in Alg. 1. For a given quantum device and a specified ansatz, the learner first constructs the training dataset following the outlined procedure in the main text. Once the dataset is prepared, the proposed ML model serves as a surrogate to optimize variational quantum algorithms in which the optimization correspond to minimize the output value of the surrogate, obviating the need for direct implementation on the specified quantum device. Since the optimization process (i.e., the third step) is conducted entirely on classical processors, it significantly reduces the overhead associated with accessing sparse quantum devices in the contemporary era.

Algorithm 1: Optimizing variational quantum algorithms in an offline manner

1. (Dataset construction) Randomly generate classical inputs $\mathbf{x} \sim [-\pi, \pi]^d$, feed it to the specified N -qubit quantum device to obtain the pre-measured state $U(\mathbf{x})|0\rangle^{\otimes N}$, and apply Pauli-based classical shadow with T shots to constitute a single training example $(\mathbf{x}, \tilde{\rho}_T(\mathbf{x}))$;
 2. (Dataset construction) Repeat the above procedure n times to construct the training dataset $\mathcal{T}_s$;
 3. (Downstream task optimization) Formalize the prediction model h_s following Eq. (6) and use it to optimize various downstream tasks without access to the quantum processor.
-

In what follows, we first briefly explain how our approach supports two crucial classes of variational quantum algorithms: the variational quantum Eigen-solvers (VQEs) and quantum neural networks (QNNs). Then, in SM K 4 a, we present numerical simulations demonstrating the application of the proposed pretraining strategy to enhance a 3-qubit ansatz for both ground-state energy estimation and binary classification tasks. To demonstrate the potential of the proposed pretraining strategy at the use-case level, we employ it to enhance VQEs when processing a class of transversefield Ising models (TFIMs) up to 50 qubits in SM K 4 b. In addition, we conduct numerical simulations to illustrate how the proposed pretraining strategy can advance VQEs when handling a class of H_5 molecules in SM K 4 c. Note that the learning scheme presented below can be readily extended to wide applications covered by variational quantum algorithms.

Remark. The focus on TFIMs and H_5 molecules stems from the fact that they are widely studied in the literature, garnering attention in quantum many-body physics and quantum chemistry, spanning algorithmic developments to experimental investigations. In particular, TFIM serves as a prototypical model for understanding quantum phase transitions, non-equilibrium dynamics, and critical phenomena in condensed matter systems [79]. Its simplicity,



Supplementary Fig. 8: **A toy model of optimizing variational quantum algorithms in an offline manner.** Left panel: In the pre-training state, the learner collects training data from a 3-qubit device with a hardware-efficient ansatz (HEA) $U(\mathbf{x})$, which contains $d = 6$ tunable parameters. Then the learner can use the collected data to train an ML model introduced in the main text to enhance various downstream tasks. Middle panel: The optimization of variational quantum Eigen-solvers (VQEs) associated with the ansatz $W(\boldsymbol{\theta})$ can be accomplished by the proposed ML model without access to the quantum resources. Right panel: The optimization of quantum neural networks (QNNs) associated with the encoding circuit $V(\mathbf{z})$ and the trainable circuit $W(\boldsymbol{\theta})$ can be accomplished by the proposed ML model without the access to the quantum resources.

coupled with its rich physics, makes it a cornerstone for testing quantum algorithms [80, 81] and benchmarking quantum simulators [82, 83]. In addition, the hydrogen molecule (e.g., H_2 and H_5) represents a small yet nontrivial system that captures essential features of electron correlation and quantum entanglement in molecular systems [84]. It is widely used to benchmark quantum computational methods in solving quantum chemistry problems with near-term quantum devices [85, 86].

Ground state energy estimation by VQE. Here we first briefly recap the mechanism of VQE when applied to estimate the ground state energy of a specified Hamiltonian. Given an N -qubit Hamiltonian $\mathfrak{H} \in \mathbb{C}^{2^N \times 2^N}$, the ground state energy estimation aims to find its minimum eigenvalue, i.e.,

$$\mathfrak{E}^* = \min_{|\psi\rangle \in \mathbb{C}^{2^N}} \langle \psi | \mathfrak{H} | \psi \rangle. \quad (\text{K5})$$

To estimate the ground state energy $\mathfrak{E}^*$, VQE adopts an ansatz $W(\boldsymbol{\theta})$ to prepare a variational quantum state $|\psi(\boldsymbol{\theta})\rangle = W(\boldsymbol{\theta})|\psi_0\rangle^{\otimes N}$ with a fixed input state $|\psi_0\rangle$. The trainable parameters $\boldsymbol{\theta}$ are optimized by minimizing the loss function $\mathcal{L}(\boldsymbol{\theta}, \mathfrak{H}) = \text{Tr}(|\psi_0\rangle\langle\psi_0|W(\boldsymbol{\theta})^\dagger \mathfrak{H} W(\boldsymbol{\theta}))$. The optimization of VQE follows an iterative manner, i.e., the classical optimizer continuously leverages the output of the quantum circuits to update $\boldsymbol{\theta}$ and the update rule is

$$\boldsymbol{\theta}^{(t+1)} = \boldsymbol{\theta}^{(t)} - \eta \frac{\partial \mathcal{L}(\boldsymbol{\theta}^{(t)}, \mathfrak{H})}{\partial \boldsymbol{\theta}}, \quad (\text{K6})$$

where η refers to the learning rate. The first-order gradient in the equation can be obtained by parameter shift rule [87]. Mathematically, the derivative with respect to the k -th parameter $\mathbf{x}_k$ for $\forall k \in [d]$ is

$$\frac{\partial \mathcal{L}(\boldsymbol{\theta}^{(t)}, \mathfrak{H})}{\partial \boldsymbol{\theta}_k} = \frac{1}{2 \sin \alpha} \left[\text{Tr}(|\psi_0\rangle\langle\psi_0|W(\boldsymbol{\theta}^{(t,+)})^\dagger \mathfrak{H} W(\boldsymbol{\theta}^{(t,+)}) - \text{Tr}(|\psi_0\rangle\langle\psi_0|W(\boldsymbol{\theta}^{(t,-)})^\dagger \mathfrak{H} W(\boldsymbol{\theta}^{(t,-)}) \right], \quad (\text{K7})$$

where $\boldsymbol{\theta}^{(t,\pm)} = \boldsymbol{\theta}^{(t)} \pm \alpha \mathbf{e}_k$, $\mathbf{e}_k$ is the unit vector along the $\boldsymbol{\theta}_k$ axis, and α can be any real number but the multiple of π because of the diverging denominator.

Supported by the proposed ML model, the optimization of $\boldsymbol{\theta}$ can be entirely carried out on classical processors. More precisely, define $W(\boldsymbol{\theta}) = U(\mathbf{x})$ (see Fig. 8 for the visual interpretation), the two terms of the derivatives in Eq. (K7) can be predicted by h_s in Alg. 1 and the optimized parameters can be obtained in an offline manner. For convenience, in the subsequent context, we dub the VQEs optimized by the proposed ML model $h(\mathbf{x})$ as *Offline-VQE*. An alternative scenario involves utilizing the optimized parameters of Offline-VQE as effective initial parameters for running VQEs on real quantum devices. In either case, the offline optimization enabled by the proposed ML model dramatically reduces the demand for quantum resources, a critical advantage given the scarcity of available quantum processors in the current landscape.

Binary classification by QNN. Denote the input space as $\mathcal{Z}$, the binary label (class) space as $\mathcal{Y} = \{0, 1\}$, and the training dataset as $\mathcal{D} = \{(\mathbf{z}^{(i)}, y^{(i)})\}_{i=1}^n$ with samples drawn i.i.d. from an unknown probability distribution $\mathbb{D}$ on $\mathcal{Z} \times \mathcal{Y}$. The purpose of a binary classification algorithm is using $\mathcal{D}$ to infer a hypothesis (a.k.a., a binary classifier) $g_{\mathcal{D}} : \mathcal{Z} \rightarrow \mathbb{R}$ from the hypothesis space to separate training examples from binary classes. This is equivalent to identifying an optimal hypothesis minimizing the expected risk $R(g) = \mathbb{E}_{(\mathbf{z}, y) \sim \mathbb{D}}[\ell(g(\mathbf{z}), y)]$, where $\ell(\cdot, \cdot)$ is the per-sample loss and for clarity we specify it as the square error [88]. Unfortunately, the inaccessible distribution $\mathbb{D}$ forbids

us to assess the expected risk directly. In practice, the binary classification algorithm alternatively learns an empirical classifier $\hat{h}$ as the global minimizer of the (regularized) loss function

$$\mathcal{L}(g, \mathcal{D}) = \frac{1}{n} \sum_{i=1}^n \frac{1}{2} \left(g(\mathbf{z}^{(i)}) - y^{(i)} \right)^2 + \mathfrak{E}(g), \quad (\text{K8})$$

where $\mathfrak{E}(g)$ is an optional regularizer. Given an unseen example $\mathbf{z}'$, its predicted label is 0 if $g(\mathbf{z}') < 0.5$; otherwise, its predicted label is 1.

When QNN is employed to implement the binary classifier, the hypothesis g is realized by variational quantum circuits followed by a predefined measurement operator Π_0 [71, 89]. The mathematical expression of the hypothesis space for the binary quantum classifier (BiQC) is

$$\mathcal{G} = \left\{ g(\mathbf{z}, \boldsymbol{\theta}) = \text{Tr} \left(V(\mathbf{z}) (|0\rangle\langle 0|)^{\otimes N} V(\mathbf{z})^\dagger W(\boldsymbol{\theta})^\dagger \Pi_0 W(\boldsymbol{\theta}) \right) \mid \boldsymbol{\theta} \in \Theta \right\}, \quad (\text{K9})$$

where $W(\boldsymbol{\theta})$ is the trainable circuit and $V(\mathbf{z})$ is another variational circuit that encodes the training example $\mathbf{z}^{(i)}$ into the quantum state $\rho(\mathbf{z}^{(i)})$. As with VQE, the optimization of trainable parameters $\boldsymbol{\theta}$ can be completed by the gradient descent methods (e.g., stochastic gradient descent or batch gradient descent) and the derivatives can be acquired by the parameter shift rule given in Eq. (K7).

Supported by the proposed ML model, the optimization of $\boldsymbol{\theta}$ for BiQC can be entirely carried out on classical processors. According to Alg. 1, the classical input $\mathbf{x}$ should be divided into parts, where the first part refers to the training example $\mathbf{z}$ and the second part refers to the trainable parameters $\boldsymbol{\theta}$ in BiQCs. In addition, the circuit architecture of $U(\mathbf{x})$ amounts to the combination of $W(\boldsymbol{\theta})$ and $V(\mathbf{z})$ (see Fig. 8 for the visual interpretation). In this way, for an arbitrary input $\mathbf{z}$ and trainable parameters $\boldsymbol{\theta}$, the derivatives $\partial \mathcal{L} / \partial \theta_k$, which is formed by $g(\mathbf{z}, \boldsymbol{\theta}^{(t)})$ and $g(\mathbf{z}, \boldsymbol{\theta}^{(t, \pm)})$, can be estimated by the proposed ML model without accessing quantum resources. Due to this offline property, we call BiQCs optimized by the proposed ML model as *Offline-BiQCs*.

Remark. The approach of leveraging the proposed ML model to propel BiQCs can be readily extended to more complex scenarios. Specifically, the proposed ML model can enhance the optimization of multi-class quantum classifiers [71] and quantum regression models [90], significantly reducing the quantum resource overhead. Moreover, it can also improve QNNs with diverse architectures, such as data re-uploading strategy [91] and quantum convolutional neural networks [92, 93].

a. Pretraining a 3-qubit HEA for both VQE and QNN

Recall that contemporary quantum devices encounter limitations such as connectivity, gate fidelities, and coherence time. To overcome these constraints, experimentalists typically adopt hardware-efficient ansatzes (HEAs) constructed from the native gate set from the specified quantum device to implement various variational quantum algorithms [77, 78]. Essentially, these HEAs are tailored to individual quantum devices, maintaining consistent gate layouts while adjusting parameters for various computational tasks, as delineated by the framework formulated in Eq. (3) of the main text. This intrinsic relation warrants the use of the proposed ML model to enhance plenty of variational quantum algorithms associated with device-specific HEAs.

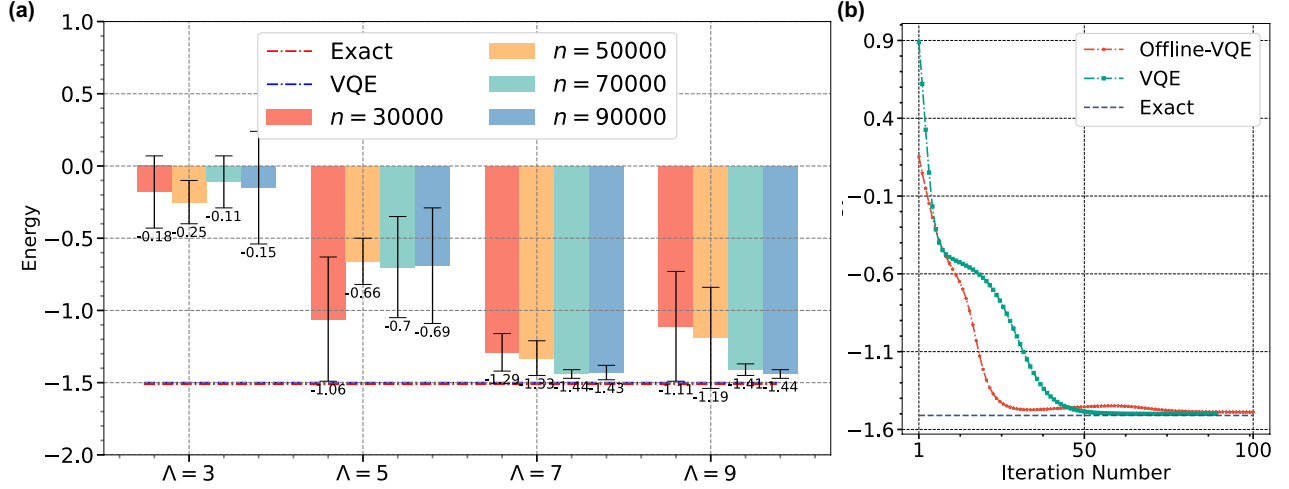
To exhibit the effectiveness of our proposal for enhancing VQEs and QNNs, we employ it to pretrain a 3-qubit hardware-efficient ansatzes (HEAs) [77, 78], followed by estimating the ground state of the Transverse-field Ising model (TFIM) and classifying a synthetic binary dataset, respectively. The explicit form of the adopted HEA is

$$U(\mathbf{x}) = \prod_{l=1}^3 \left(\text{CNOT}_{2,3} \text{CNOT}_{1,2} \bigotimes_{i=1}^3 \text{RY}(\mathbf{x}_{i+3(l-1)}) \right), \quad (\text{K10})$$

where $\text{CNOT}_{a,b}$ denotes applying CNOT gate to a -th and b -th qubits. This ansatz contains in total $d = 9$ classical parameters.

Pre-training. At the stage of dataset construction, we collect in total 90000 training examples and the shot number of each training example is set as $T = 1000$. The random seed to generate such a dataset is set as 123. Once the data collection is completed, we form the kernel-based learning model to accomplish the following two downstream tasks on the classical side.

Downstream task I: ground state energy estimation of TFIM. The mathematical form of the exploited 3-qubit one-dimensional (1D) TFIM is $H_{\text{TFIM}} = -0.1(Z_1 Z_2 + Z_2 Z_3) + 0.5(X_1 + X_2 + X_3)$. The ground state energy of H_{TFIM} is $\mathfrak{E}_{\text{TFIM}}^* = -1.51\text{Ha}$.



Supplementary Fig. 9: **Simulation results for ground state energy estimation of TFIM.** (a) THE ESTIMATED ENERGY OF OFFLINE-VQE UNDER DIFFERENT HYPER-PARAMETER SETTINGS. The label $n = a$ refers that the number of training data is a . The label ‘Exact’ denotes the accurate ground state energy of the employed TFIM. The blue dotted line highlights the estimated result of traditional VQE with infinite shots. The vertical bar reflects the standard error of Offline-VQE in each setting. (b) THE OPTIMIZATION PROCESS OF TOP-1 OFFLINE-VQE AND CONVENTIONAL VQE.

In numerical simulations, we adopt different hyper-parameter settings to evaluate the performance of Offline-VQE. In particular, we vary the number of training examples as $n \in \{30000, 50000, 70000, 90000\}$ and set the truncation frequency as $\Lambda \in \{3, 5, 7, 9\}$. The maximum iteration is set as 200. The initial parameters are uniformly sampled from the range $[-\pi, \pi]^9$. Each setting is repeated 5 times to obtain the statistical results. For comprehensive, we employ the conventional VQE with the same ansatz, the same initial parameters, the same optimizer, and infinite shots as the benchmark.

The achieved results are depicted in Fig. 9. Fig. 9(a) demonstrates the performance of Offline-VQE under various settings. A key observation is that with the increased frequency truncation and number of training examples, the estimated ground state energy of Offline-VQE converges to the exact result. Namely, when $\Lambda \geq 7$ and $n \geq 70000$, the estimation error is less than 0.1Ha. Besides, Fig. 9(b) compares the top-1 Offline-VQE (i.e., $\Lambda = 7$ and $n = 90000$) with conventional VQE. The achieved results exhibit a similar convergence rate during the optimization, where both of them converge to the near-optimal value after 50 iterations. These results validate the potential of Offline-VQE in advancing conventional VQEs.

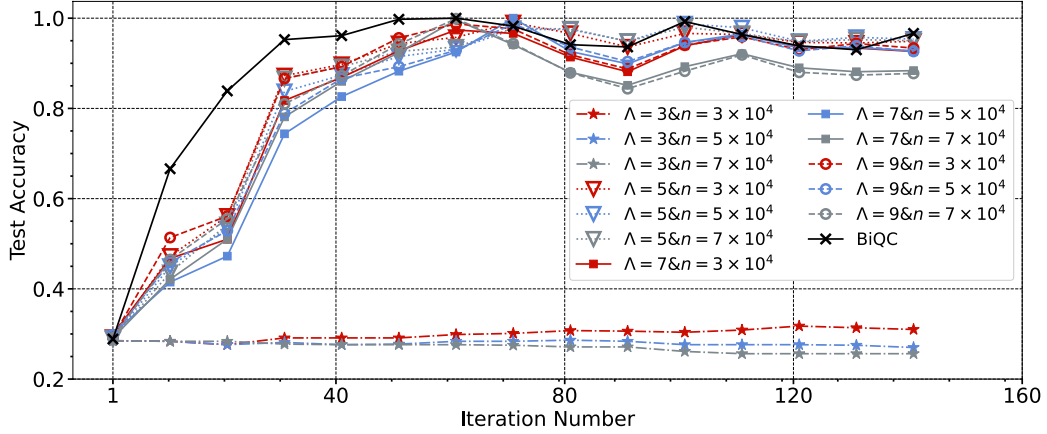
Downstream task II: binary synthetic data classification. The construction rule of the binary synthetic dataset follows the study [78]. That is, we randomly and uniformly sample classical data $\mathbf{z}$ in the range $[-\pi, \pi]^3$ and then embed them into a 3-qubit quantum circuit $V(\mathbf{z}) = \text{CNOT}_{2,3} \text{CNOT}_{1,2} \bigotimes_{j=1}^3 \text{RY}(\mathbf{z}_j)$. Then, we evolve the quantum state $V(\mathbf{z})|0\rangle^{\otimes 3}$ by the unitary $W(\boldsymbol{\theta}^*) = \prod_{l=1}^2 \left(\text{CNOT}_{2,3} \text{CNOT}_{1,2} \bigotimes_{j=1}^3 \text{RY}(\boldsymbol{\theta}_{j+3(l-1)}^*) \right)$ with the fixed parameters $\boldsymbol{\theta}^*$ and measuring the evolved state under the observable $X \otimes \mathbb{I}_4$. For the input $\mathbf{z}^{(i)}$, its label is defined as

$$y^{(i)} = \text{sign} \left(\text{Tr} \left(W(\boldsymbol{\theta}^*) V(\mathbf{z}^{(i)}) (|0\rangle\langle 0|)^{\otimes 3} V(\mathbf{z}^{(i)})^\dagger W(\boldsymbol{\theta}^*)^\dagger (X \otimes \mathbb{I}_4) \right) \right), \quad (\text{K11})$$

where $\text{sign}(\cdot)$ denotes the sign function. We collect 500 positive samples and 500 negative samples and split them into the training set and the test set with a ratio of 0.2.

In numerical simulations, we adopt different hyper-parameter settings to evaluate the performance of Offline-BiQC. The varied settings are analogous to those employed in the offline-VQE, where the number of training examples is set as $n \in \{30000, 50000, 70000\}$ and the truncation frequency is set as $\Lambda \in \{3, 5, 7, 9\}$. The initial parameters are uniformly sampled from the range $[-\pi, \pi]^6$. The learning rate and the maximum iteration are set as 0.5 and 140, respectively. For comprehensive, we employ the conventional BiQC with the same encoding and training circuits, the same initial parameters, the same optimizer, and infinite shots as the benchmark.

The simulation results are shown in Fig. 10. When $\Lambda = 3$, the test accuracy of Offline-BiQC is lower than 40% no matter how the number of training examples n is. These results reflect the crucial role of the truncation value in warranting the performance of Offline-BiQC. In addition, Offline-BiQC attains a superior performance with an increased truncation value. Namely, when $\Lambda \geq 5$ and $n \geq 30000$, its performance is comparable with conventional



Supplementary Fig. 10: **Simulation results of Offline-BiQC.** The figure depicts the test accuracy of Offline-BiQC under varied settings. The label ‘ $\Lambda = a \& n = b$ ’ denotes that the truncation value is a and the number of training examples used to form the classical representations is b . The label ‘BiQC’ refers to the conventional BiQC optimized in an online manner.

BiQC, where both of them attain a test accuracy above 85%. These results validate the capability of Offline-BiQC to advance the study of quantum neural networks by greatly reducing the demand for quantum resources. Another phenomenon is that Offline-BiQC and BiQC encounter the oscillated test accuracy after 80 iterations. This is mainly caused by the employed large learning rate. Adopting a smaller learning rate, associated with the learning rate decay strategy, can mitigate this issue.

b. Pretraining Hamiltonian-variational ansatz towards a class of Transverse-field Ising model

Following the definition of 3-qubit TFIM in SM K 4 a, the mathematical expression of a family of N -qubit (one-dimensional) TFIMs is

$$\mathcal{H}_{\text{TFIM}} = \left\{ \mathcal{H}_{\text{TFIM}}(J; h) = \sum_{i=1}^{N-1} J Z_i Z_{i+1} + h \sum_{i=1}^N X_i \mid J, h \in \mathbb{R} \right\}, \quad (\text{K12})$$

where J and h refer to the coupling strength and the external field applied in a transverse direction, respectively. In the following, we evaluate the performance of the proposed ML model in pretraining a Hamiltonian-variant ansatz to minimize the ground state energies in $\mathcal{H}_{\text{TFIM}}$.

Benchmark methods. To attain a comprehensive understanding about the capabilities of the proposed ML model, we compare its performance with three other types of classical learning surrogates in pretraining VQEs. Additionally, we use conventional VQEs as a benchmark to evaluate the discrepancy between the optimized results and the exact ground-state energy. Below, we outline the fundamental mechanisms of each approach.

Method I: Conventional VQE. The implementation of VQE mainly follows the rules introduced in Eq. (K6). The only difference is that we set the number of measurements to infinity (i.e., $T \rightarrow \infty$) and compute the gradients using the *automatic differentiation method*, i.e., the update of trainable parameters is completed by the Adam optimizer [94]. The infinite measurements can eliminate the influence of estimation errors, allowing us to isolate and examine how the employed ansatz $U(\mathbf{x})$ affects the performance, i.e., the discrepancy between the optimized results of VQE and the exact ground-state energy.

Method II: The proposed ML model. The first surrogate applied to the pre-training VQE is the proposed ML model, i.e.,

$$h_s(\mathbf{x}, \mathbf{H}) = \frac{1}{n} \sum_{i=1}^n \kappa_{\Lambda}(\mathbf{x}, \mathbf{x}^{(i)}) g(\mathbf{x}^{(i)}, \mathbf{H}),$$

where $\kappa_{\Lambda}(\mathbf{x}, \mathbf{x}^{(i)})$ is the truncated trigonometric monomial kernel and $g(\mathbf{x}^{(i)}, \mathbf{H}) = \text{Tr}(\tilde{\rho}_T(\mathbf{x}^{(i)}) \mathbf{H})$ refers to the shadow estimation of $\text{Tr}(\rho(\mathbf{x}^{(i)}) \mathbf{H})$ with T snapshots for each training example in $\mathcal{T}_s$ (Step 2 in Alg. 1).

Given access to the proposed ML model $h_s(\mathbf{x}, \mathbf{H})$, the initialized parameters of $U(\mathbf{x})$ take the form as

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} h_s(\mathbf{x}, \mathbf{H}). \quad (\text{K13})$$

To find the parameters minimizing the surrogate loss $h_s(\mathbf{x}, \mathbf{H})$, we employ Adam optimizer, an advanced gradient-descent-based method.

Method III: kernel ridge regression. We identify that when the employed ansatz $U(\mathbf{x})$ consists of d RZ gates and an arbitrary number of Clifford gates, the estimated energy in VQE can be reformulated into an inner-product form under the PTM representations, i.e., $\text{Tr}(HU(\mathbf{x})\rho_0U(\mathbf{x})^\dagger) = \langle \mathbf{w}^*, \vec{\Phi}(\mathbf{x}) \rangle$, where $\vec{\Phi}(\mathbf{x}) = \{\Phi_{\omega}(\mathbf{x})\}_{\omega}$ is the 3^d -dimensional vector with $\omega \in \{0, \pm 1\}^d$ and $\Phi_{\omega}(\mathbf{x})$ being the trigonometric monomial basis, and the optimal parameters $\mathbf{w}^* = \{2^{\|\omega\|_0} \text{Tr}(\rho_{\omega} \mathbf{H})\}_{\omega}$ is also a 3^d -dimensional vector with ρ_{ω} being parameter-independent matrix.

This inner-product reformulation enables us to utilize kernel ridge regression (KRR) [95] to construct a learning surrogate. Specifically, given the training dataset $\{(\mathbf{x}^{(i)}, y^{(i)})\}_{i=1}^n$ with $y^{(i)} = \text{Tr}(HU(\mathbf{x}^{(i)})\rho_0U(\mathbf{x}^{(i)})^\dagger)$ in the ideal case or $y^{(i)} = \text{Tr}(\tilde{\rho}_T(\mathbf{x}^{(i)})\mathbf{H})$ in the shadow case for $\forall i \in [n]$, we can use a KRR model $h_{\text{KRR}}(\mathbf{x}; \mathbf{w}) = \langle \mathbf{w}, \vec{\Phi}(\mathbf{x}) \rangle$ to minimize the empirical risk, i.e., $\min_{\mathbf{w}} \frac{1}{2} \sum_{i=1}^n (h_{\text{KRR}}(\mathbf{x}^{(i)}; \mathbf{w}) - y^{(i)})^2 + \frac{\alpha}{2} \|\mathbf{w}\|_2$, where α is a hyper-parameter. Under the *dual representation* [95], the closed-form solution of the above minimization problem is

$$h_{\text{KRR}}(\mathbf{x}; \hat{\mathbf{w}}) = \mathbf{k}(\mathbf{x}) \cdot (K + \alpha \mathbb{I}_n)^{-1} \mathbf{y}, \quad (\text{K14})$$

where $\hat{\mathbf{w}}$ refers to the optimized parameters, $\mathbf{y} = [y^{(1)}, \dots, y^{(n)}]$ is the label vector, $K \in \mathbb{R}^{n \times n}$ refers to the kernel matrix with

$$K_{ij} \equiv k(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) = \langle \vec{\Phi}(\mathbf{x}^{(i)}), \vec{\Phi}(\mathbf{x}^{(j)}) \rangle = \prod_{m=1}^d (1 + 2 \cos(\mathbf{x}_m^{(i)} - \mathbf{x}_m^{(j)})), \quad \forall i, j \in [n], \quad (\text{K15})$$

and $\mathbf{k}(\mathbf{x})$ is an n -dimensional vector whose i -th entry is $\mathbf{k}(\mathbf{x})_i = \langle \vec{\Phi}(\mathbf{x}), \vec{\Phi}(\mathbf{x}^{(i)}) \rangle = \prod_{m=1}^d (1 + 2 \cos(\mathbf{x}_m - \mathbf{x}_m^{(i)}))$. Given access to the optimized learning model $h_{\text{KRR}}(\mathbf{x}; \hat{\mathbf{w}})$ in Eq. (K14), the pre-training VQE amounts to solving the following minimization problem,

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} h_{\text{KRR}}(\mathbf{x}, \hat{\mathbf{w}}). \quad (\text{K16})$$

The optimization is completed by the Adam optimizer. After that, the optimized result serves as the initialized parameters of $U(\mathbf{x})$ for the corresponding VQE task.

Method IV: Random Fourier features. The computational efficiency of KRR highly depends on the number of training examples n . When the number of training examples n becomes large, the space and time cost of KRR scales with $\mathcal{O}(n^2)$ and $\mathcal{O}(n^3)$, respectively. To address the computational bottleneck, an alternative is employing random Fourier features (RFF) techniques to learn the given dataset [96]. More specifically, when the target kernel is *shift-invariant*, i.e., $k(\mathbf{x}, \mathbf{x}') = k(\mathbf{x} - \mathbf{x}')$, the KRR model $h_{\text{KRR}}(\mathbf{x}; \mathbf{w})$ can be estimated by RFF-based regression model, i.e.,

$$h_{\text{RFF}}(\mathbf{x}, \mathbf{w}) = \langle \mathbf{w}, \vec{\Psi}(\mathbf{x}) \rangle, \quad (\text{K17})$$

where $\vec{\Psi}(\mathbf{x}) = [\cos(\frac{\gamma}{d'} \langle \boldsymbol{\theta}_1, \mathbf{x} + b_1 \rangle), \sin(\frac{\gamma}{d'} \langle \boldsymbol{\theta}_1, \mathbf{x} + b_1 \rangle), \dots, \cos(\frac{\gamma}{d'} \langle \boldsymbol{\theta}_{d'}, \mathbf{x} + b_{d'} \rangle), \sin(\frac{\gamma}{d'} \langle \boldsymbol{\theta}_{d'}, \mathbf{x} + b_{d'} \rangle)]^\top$ is the RFF with d' dimensions, γ and d' are tunable hyper-parameters, $\boldsymbol{\theta}_i$ for $\forall i \in [d']$ are d' -dimensional vectors sampled from a multivariate standard normal distribution, and b_i for $\forall i \in [d']$ is scalar sampled from the uniform distribution ranging from $-\pi$ to π .

Following Eq. (K15), an immediate observation is that the kernel under consideration is *shift-invariant*, justifying the use of an RFF-based regression model as a surrogate. Specifically, given the training dataset $\{(\mathbf{x}^{(i)}, y^{(i)})\}_{i=1}^n$, we train the RFF-based regression model by minimizing the empirical risk

$$\min_{\mathbf{w}} \frac{1}{2} \sum_{i=1}^n (h_{\text{RFF}}(\mathbf{x}^{(i)}; \mathbf{w}) - y^{(i)})^2 + \frac{\alpha}{2} \|\mathbf{w}\|_2, \quad (\text{K18})$$

where α is a hyper-parameter. Denote the optimized parameters as $\hat{\mathbf{w}}$. As before, we use Adam optimizer to locate the solution that minimizes the surrogate loss, i.e.,

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} h_{\text{RFF}}(\mathbf{x}, \hat{\mathbf{w}}). \quad (\text{K19})$$

The obtained estimate serves as the initialization parameters of $U(\mathbf{x})$ for the corresponding VQEs.

Method V: Multi-layer Perceptron. We last exploit the performance of deep learning protocols for VQE's parameter initialization. For this purpose, we adopt multi-layer perceptron (MLP) as the deep learning protocol [97], whose mathematical expression is

$$h_{\text{MLP}}(\mathbf{x}; \mathbf{W}) = W_L \text{ReLU}(\dots \text{ReLU}(W_2 \text{ReLU}(W_1 \mathbf{x}))), \quad (\text{K20})$$

where $\mathbf{W} = \{W_i\}_{i=1}^L$ refers to L trainable weight matrices and the size of W_i for $\forall i \in [L]$ is adjustable. Given the training dataset $\mathcal{T}_s = \{(\mathbf{x}^{(i)}, y^{(i)})\}_{i=1}^n$, we train MLP by minimizing the empirical risk

$$\min_{\mathbf{W}} \frac{1}{2} \sum_{i=1}^n \left(h_{\text{MLP}}(\mathbf{x}^{(i)}; \mathbf{w}) - y^{(i)} \right)^2 + \frac{\alpha}{2} \sum_i \|W_i\|_F, \quad (\text{K21})$$

where α is a hyper-parameter. Given access to the optimized MLP $h_{\text{MLP}}(\mathbf{x}, \hat{\mathbf{W}})$, the initialization parameters for $U(\mathbf{x})$ are acquired via solving

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} h_{\text{MLP}}(\mathbf{x}, \hat{\mathbf{W}}). \quad (\text{K22})$$

For a fair comparison, the optimization of MLP is also completed by the Adam optimizer.

Note that while the employed MLP lacks provable guarantees, benchmarking its performance is crucial given the growing success of deep learning in quantum system learning tasks.

Metrics. For comprehensive, we employ two metrics to evaluate the performance of different benchmarking methods.

① *Prediction error.* Our first metric concerns the ability of the trained learning models to predict unseen examples. Denote the test dataset as $\{(\mathbf{x}^{(i)}, y^{(i)})\}_{i=1}^{n_{\text{test}}}$ such that the test examples are sampled from the same distribution with the training examples. The prediction error refers to the mean square error on the test dataset, i.e.,

$$R_{\text{predict}} = \frac{1}{n_{\text{test}}} \sum_{i=1}^{n_{\text{test}}} \left(h_*(\mathbf{x}) - y^{(i)} \right)^2, \quad (\text{K23})$$

where $h_* \in \{h_s, h_{\text{KRR}}, h_{\text{RFF}}, h_{\text{MLP}}\}$.

② *Absolute energy difference.* The second metric compares the difference between the estimated energy $\hat{E}_{\text{VQE}}$ returned by VQE and the exact minimum ground state energy E_{Exact} of the given Hamiltonian, i.e.,

$$R_{\text{abs}} = \left| \hat{E}_{\text{VQE}} - E_{\text{Exact}} \right|. \quad (\text{K24})$$

The estimated energy is defined by $\hat{E}_{\text{VQE}} = \text{Tr}(\mathbf{H}U(\hat{\mathbf{x}})\rho_0 U(\hat{\mathbf{x}})^\dagger)$. Given different parameters $\hat{\mathbf{x}}$ returned by varied learning surrogates with $h_* \in \{h_s, h_{\text{KRR}}, h_{\text{RFF}}, h_{\text{MLP}}\}$, we can evaluate how these surrogates advance the parameter initialization by comparing R_{abs} .

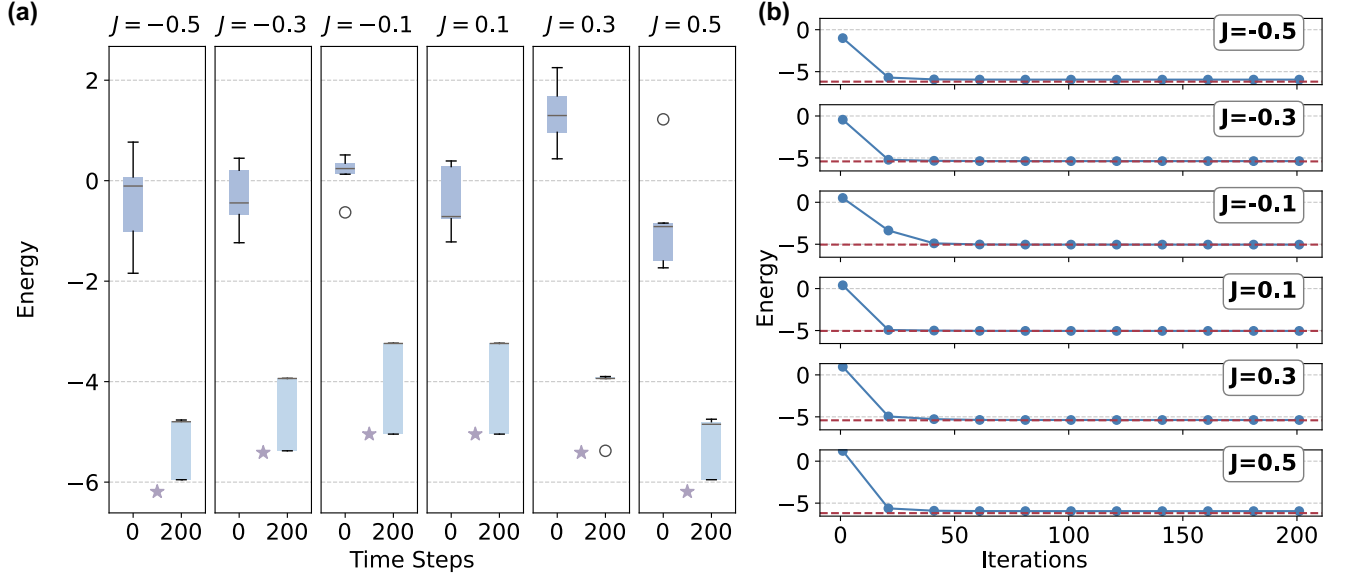
Problem setup. The detailed settings of TFIMs are as follows. We investigate six TFIMs in $\mathcal{H}_{\text{TFIM}}$ in Eq. (K12). In particular, these Hamiltonians are constructed by fixing $N = 10$, $h = -0.5$, and varying the coupling strength with $J \in \{-0.5, -0.3, -0.1, 0.1, 0.3, 0.5\}$.

Moreover, Hamiltonian-informed ansatz is adopted to estimate their ground state energies, i.e.,

$$U(\mathbf{x}) = \prod_{i=1}^{N-1} \exp \left(-\imath \frac{\mathbf{x}_i}{2} (Z_i \otimes Z_{i+1}) \right) \prod_{i=N}^{2N-1} \left(-\imath \frac{\mathbf{x}_i}{2} (X_i) \right) \mathbf{H}^{\otimes N}, \quad (\text{K25})$$

where $\imath$ refers to the imaginary unit and $\mathbf{H}$ refers to the Hadamard gate. As illustrated in Fig. 11, optimal parameters for this ansatz exist to minimize the six Hamiltonians under investigation. In particular, for each setting, we perform five repetitions with different initial parameters to collect statistical results, using a learning rate of 0.8 for the Adam optimizer. Fig. 11(a) illustrates the statistical performance of VQEs. That is, for each configuration, the exact ground-state energy can be accurately estimated after 200 iterations with *well-initialized parameters*. For clarity, we visualize the training dynamics of the VQE for the top-performing setting in Fig. 11(b). An immediate observation is that across all settings, the absolute error between the estimated energy and the exact value remains small, which are 0.24, 0.04, 5.06×10^{-4} , 4.92×10^{-4} , 0.04, and 0.24 for $J = -0.5, -0.3, -0.1, 0.1, 0.3, 0.5$, respectively.

The process of pretraining VQEs by classical learning surrogates follows the steps outlined in Alg. 1. Specifically, in Steps 1&2, we construct the training dataset $\mathcal{T}_s$ by randomly sampling $\mathbf{x} \in [-\pi, \pi]^{19}$ from a uniform distribution and



Supplementary Fig. 11: **Performance of conventional VQEs on the family of TFIMs with $N = 10$.** (a) STATISTICAL PERFORMANCE OF CONVENTIONAL VQES. The box plots show the distribution of the estimated energies at the first iteration and after 200 iterations for each setting, with J values ranging from -0.5 to 0.5 . Each box represents the interquartile range (IQR), with the horizontal line inside indicating the median energy. The whiskers extend to the most extreme data points within 1.5 times the IQR, and outliers beyond this range are shown as individual circles. The purple stars represent the exact ground-state energies. (b) TRAINING DYNAMICS OF THE BEST VQES. The plots show the convergence of the estimated energies toward the exact ground-state energies (dashed red lines) over 200 iterations for the top-performing instance in each setting.

inputting these values into the ansatz $U(\mathbf{x})$ in Eq. (K25) to generate classical shadows (when $N = 10$, the dimension of the input $\mathbf{x}$ is $d = 19$). The snapshot for each quantum state $\rho(\mathbf{x})$ is fixed to be $T = 100$ and the number of trainable examples is set $n = 7500$.

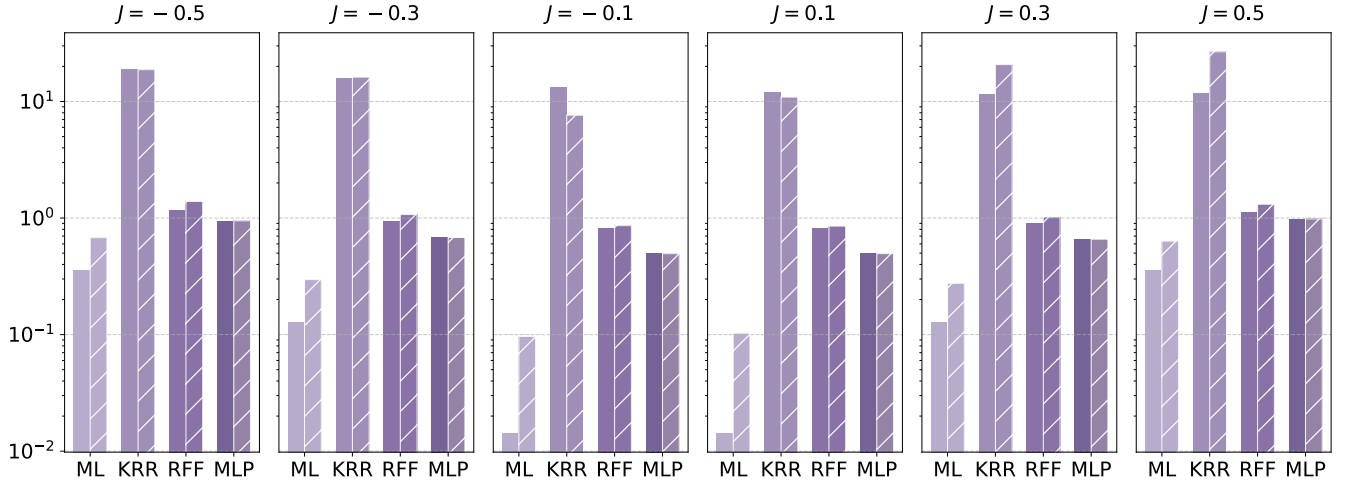
Once the dataset $\mathcal{T}_s$ is collected, we move to Step 3 of Alg. 1. That is, the dataset $\mathcal{T}_s$ is used to pretrain VQEs by the four classical surrogates to estimate the ground-state energies of the six explored TFIMs. The hyper-parameter settings of each learning surrogate are as follows.

- **Proposed ML model.** For all configurations, we fix the truncation value as $\Lambda = 2$ and the number of training examples as $n = 7500$ to obtain the corresponding learning surrogate $h_s(\mathbf{x}; \mathbf{H})$ following Eq. (K13). In the procedure of minimizing the surrogate loss as defined in Eq. (K13), we set the learning rate of Adam optimizer as 0.01 and the total number of iterations as 500.
- **Kernel regression model.** For all configurations, we set the number of training examples as $n = 100$ to avoid the memory issue. The learning models h_{KRR} are trained following Eq. (K14). Once the trained models are available under different J values, we follow Eq. (K16) to minimize their corresponding surrogate losses. As with the proposed ML model, we set the learning rate of Adam optimizer as 0.01 and the total number of iterations as 500.
- **Random Fourier Feature.** For all configurations, we set the dimension of random Fourier features as $d' = 800$ in Eq. (K17), to match the feature dimension of the proposed learning models $h_s(\mathbf{x}; \mathbf{H})$. We follow Eq. (K18) to h_{RFF} under different J values. The number of training examples is set as $n = 7500$ and the hyper-parameter is fixed to be $\alpha = 1$. In the procedure of minimizing the surrogate loss as defined in Eq. (K19), we follow the same routine with $h_s(\mathbf{x}; \mathbf{H})$, where the learning rate of Adam optimizer is 0.01 and the total number of iterations is 500.
- **MLP.** For all configurations, the exploited $h_{\text{MLP}}(\mathbf{x}, \mathbf{W})$ in Eq. (K20) refers to a 2-hidden-layer MLP, where the dimension of the input layer, the first hidden layer, the second hidden layer, and the output layer is 19, 32, 8, and 1, respectively. The total number of trainable parameters of MLP is 872, comparable to those used in the proposed learning model and KRR. Moreover, each hidden layer is followed by a ReLU activation function. We use $n = 7500$ training examples to optimize MLPs under varied J values are as follows. The learning rate of the Adam optimizer is set as 0.001. The batch size is set as 200. The hyper-parameter α is set as 0.0001. Once

$h_{\text{MLP}}(\mathbf{x}, \hat{\mathbf{W}})$ is accessible, we follow the surrogate loss in Eq. (K22) to pretrain VQEs. As with the proposed ML model, we set the learning rate of Adam optimizer as 0.01 and the total number of iterations as 500.

Remark. To assess the impact of noisy labels caused by the finite snapshots of classical shadows in each training example, we also benchmark the performance of the employed learning surrogate under the assumption of an infinite number of shots for each configuration, i.e., $T \rightarrow \infty$, $\hat{\rho}_T(\mathbf{x}) = \rho(\mathbf{x})$, and $y^{(i)} = \text{Tr}(\rho(\mathbf{x}^{(i)})\mathbf{H})$. We repeat each configuration described above five times to collect statistical results.

Numerical results. The predictor error of the employed four learning surrogates (i.e., the first metric R_{predict} in Eq. (K23)) is demonstrated in Fig. 12, which is evaluated by 500 test examples. For all subplots, the proposed ML models outperform the rest three learning surrogates under both ideal (highlighted by the solid bar) and noisy (highlighted by the shadowed bar) scenarios. Particularly, the prediction error of the proposed ML model under the ideal (noisy) setting achieves 0.36 (0.68), 0.13 (0.30), 0.01 (0.10), 0.01 (0.10), 0.13 (0.28), and 0.36 (0.63) when $J = -0.5, -0.3, -0.1, 0.1, 0.3, 0.5$, respectively. Another observation is that while the noisy labels caused by finite snapshots (i.e., $T = 100$) can slightly degrade prediction performance of the proposed ML model, the impact is not substantial, with a maximum degradation of no more than 0.32 across all settings.



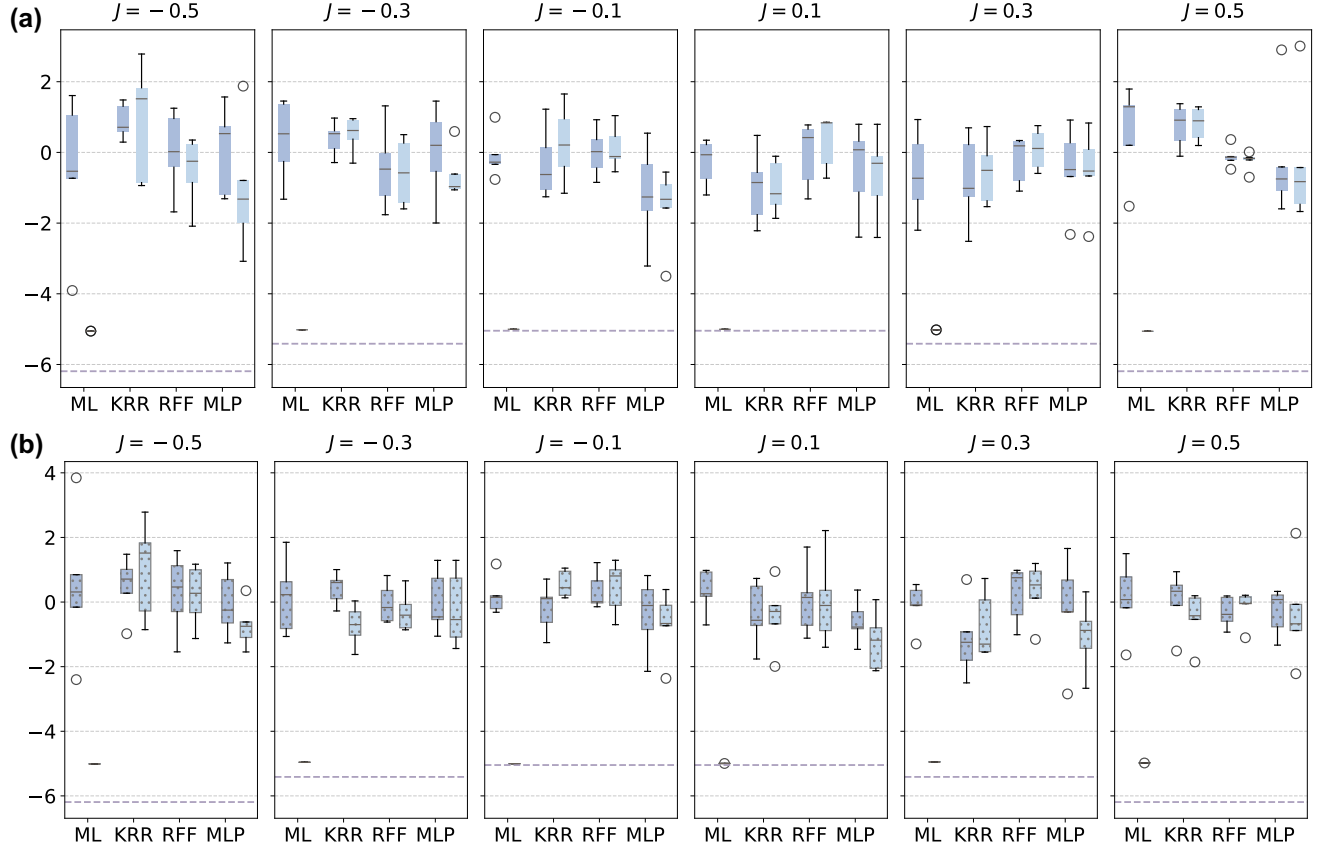
Supplementary Fig. 12: **Prediction error of four learning surrogates for the family of TFIMs with $N = 10$.** Each subplot corresponds to the simulation results for a specific value of J in the TFIM. Solid bars represent the prediction error under ideal conditions ($T \rightarrow \infty$ in the training dataset $\mathcal{T}_s$), while shadowed bars with a “/” pattern denote the noisy case ($T = 100$). The four surrogates are labeled as ML (the proposed ML model), KRR (Kernel ridge regression), RFF (random Fourier features), and MLP (Multi-Layer Perceptron), respectively.

Next, we use the optimized learning surrogates to pretrain VQEs for the six explored TFIMs. The estimated energies, both with and without pretraining by the four learning surrogates, are illustrated in Fig. 13. An immediate observation is that compared to other learning surrogates, the estimated energy obtained from the proposed ML model is greatly closer to the exact result in both ideal and noisy settings. Notably, unlike the proposed ML model, the performance of VQEs does not improve after pretraining with other learning surrogates and, in some cases, may even degrade. These contrastive outcomes further validate the effectiveness of our proposed learning protocol.

Supplementary Table 5: **Comparison between conventional VQEs and the proposed ML model on a family of TFIMs with $N = 10$.** The table presents the estimated ground state energies for various coupling strengths J , as obtained from the conventional VQE and the proposed ML model under both ideal and noisy conditions (labeled by ‘Pretrain (ideal)’ and ‘Pretrain (shadow)’, respectively). The notation ‘ $a \pm b$ ’ indicates that the mean value is a and the variance is b .

	$J = -0.5$	$J = -0.3$	$J = -0.1$	$J = 0.1$	$J = 0.3$	$J = 0.5$
VQE	-5.252 ± 0.327	-4.512 ± 0.498	-3.960 ± 0.784	-3.961 ± 0.784	-4.213 ± 0.339	-5.26 ± 0.32
Pretrain (ideal)	-5.056 ± 0.000	-5.020 ± 0.000	-5.002 ± 0.000	-5.002 ± 0.000	-5.021 ± 0.000	-5.057 ± 0.000
Pretrain (shadow)	-5.012 ± 0.000	-4.959 ± 0.000	-5.002 ± 0.000	-4.996 ± 0.000	-4.951 ± 0.000	-4.981 ± 0.000

Another key observation from the simulation results is that the performance of the proposed ML model remains



Supplementary Fig. 13: **Pretraining performance of different learning surrogates on the family of TFIMs with $N = 10$.** The upper panel (a) and the lower panel (b) separately exhibit the simulation results under the ideal and noisy settings ($T \rightarrow \infty$ and $T = 100$). Each subplot corresponds to a specific J value in the TFIM, and the box plots depict the estimated ground-state energy distributions for four learning surrogates. For each surrogate, the estimated energies without pre-training and after pre-training are visualized by the deep and light blue colors, respectively. Dashed horizontal lines represent the exact ground-state energies.

comparable between the ideal and noisy settings, as illustrated in Figs. 13(a) and (b), respectively. This indicates that the noisy labels introduced by the finite snapshots of classical shadows only subtly effect the performance of the proposed ML model. Furthermore, the results suggest that a constant number of snapshots (e.g., $T = 100$) is sufficient to achieve strong performance, aligning with the findings of Theorem 2 in the main text.

Finally, we compare the performance of the conventional VQE (using infinite measurements) with the proposed ML model. The statistical results are summarized in Table 5. The first key observation is that the proposed ML model, in both ideal and noisy settings, generally achieves energy estimates that are comparable to or better than those of the conventional VQE, particularly at coupling strengths $J \in \{-0.3, -0.1, 0.1, 0.3\}$. The second key observation is the consistency and stability of the proposed ML model's performance across both settings, especially at intermediate coupling strengths, where the VQE exhibits significant variance and large estimation errors. These findings confirm the potential of the proposed ML-based approach to enhance conventional VQE methods, particularly in practical scenarios with constrained quantum resources.

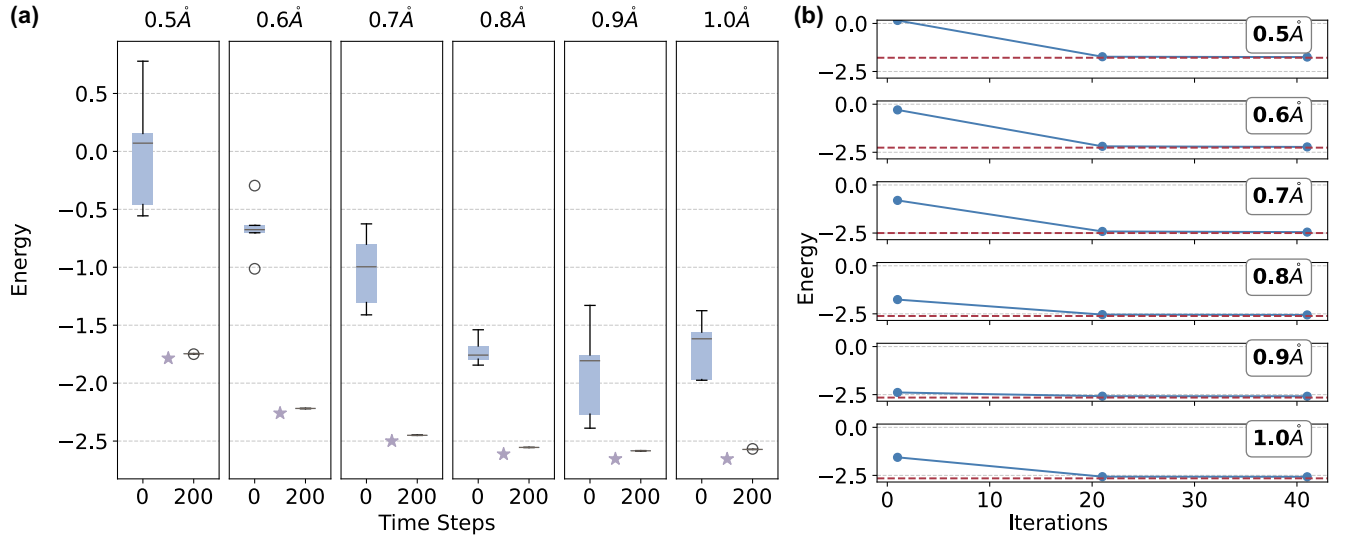
Additional information about pretraining VQE on a 50-qubit TFIM. In the main text, we apply the proposed ML model to pretrain VQE on a 50-qubit TFIM. The explored Hamiltonian, ansatz, optimizers, and metrics are identical to those introduced earlier, except that the qubit size N is increased from 10 to 50. Recall the main plot in Fig. 3(b) of the main text, which illustrates the optimization dynamics of the proposed ML model during the minimization of the surrogate loss in Eq. (K13). The estimated energies after pretraining are comparable and closely approximate the optimal results, which are -24.95 , -24.90 , and -25.99 for the ideal dataset, shadow dataset, and the exact value, respectively. Furthermore, for both the noisy and ideal cases, the variance diminishes as the number of iterations increases. These results highlight the effectiveness of the proposed ML model in large-scale scenarios, showcasing its potential to facilitate a wide range of practical use-case tasks.

c. Pretraining Hamiltonian-variational ansatz towards a class of H_5 molecules

Here we apply the proposed method to pretrain Hamiltonian-variational ansatz towards a class of H_5 molecules. The detailed settings of H_5 molecules are as follows. We center on six H_5 molecules in $\mathcal{H}_{H_5}$ with varying bond lengths, i.e., $R \in \{0.5\text{\AA}, 0.6\text{\AA}, 0.7\text{\AA}, 0.8\text{\AA}, 0.9\text{\AA}, 1.0\text{\AA}\}$. Moreover, a simplified ansatz inspired by the unitary coupled-cluster ansatz is exploited to estimate their ground state energies, i.e.,

(K26)

The simplifications are twofold. First, we remove all quantum gates interacting with the last three qubit wires. Second, we decompose the double-excitation gates into RZ and Clifford gates, leveraging their universal properties. Consequently, by initializing the state as $\rho_0 = |1111100000\rangle\langle 1111100000|$, the corresponding VQE in Eq. (K6) achieves good performance in estimating the ground-state energies of the six explored $H_{H_5}(R)$ molecules.

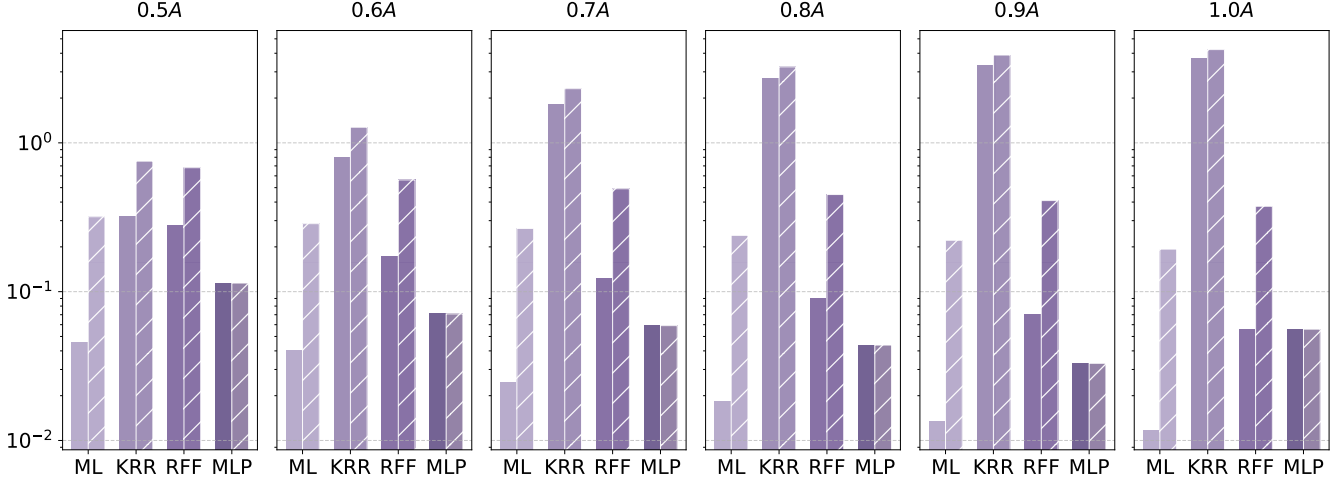


Supplementary Fig. 14: **Performance of conventional VQEs on the family of H_5 molecules with $N = 10$.** (a) STATISTICAL PERFORMANCE OF CONVENTIONAL VQES. The box plots show the distribution of the estimated energies at the first iteration and after 200 iterations for each setting, with the bond length R ranging from $0.5\text{\AA}$ to $1.0\text{\AA}$. All components of the box follow the same explanations in Fig. 11. (b) TRAINING DYNAMICS OF THE BEST VQES. The plots show the convergence of the estimated energies toward the exact ground-state energies (dashed red lines) over 200 iterations for the top-performing instance in each setting.

Benchmark methods and metrics. The benchmark models and metrics are exactly identical to those in SM K 4 b.

Problem setup. The simulation results in Fig. 14 validate that optimal parameters for the ansatz in Eq. (K26) exist to minimize the six Hamiltonians under investigation. In particular, for each setting, we perform five repetitions with different initial parameters to collect statistical results, using a learning rate of 0.8 for the Adam optimizer. Fig. 14(a) illustrates the statistical performance of VQEs. That is, for each configuration, the exact ground-state energy can be accurately estimated after 200 iterations, especially for the small bond lengths. For clarity, we visualize the training dynamics of the VQE for the top-performing setting in Fig. 14(b). Specially, across all settings, the absolute error between the estimated energy and the exact value remains small, which are 0.04, 0.04, 0.05, 0.06, 0.07, and 0.08 for $R = 0.5\text{\AA}, 0.6\text{\AA}, 0.7\text{\AA}, 0.8\text{\AA}, 0.9\text{\AA}, 1.0\text{\AA}$, respectively.

Similar to the task of TFIMs, the process of pretraining VQEs by classical learning surrogates follows the steps outlined in Alg. 1. In Steps 1&2, we construct the training dataset $\mathcal{T}_s$ by randomly sampling $\mathbf{x} \in [-\pi, \pi]^{14}$ from a uniform distribution and inputting these values into the ansatz $U(\mathbf{x})$ in Eq. (K26) to generate classical shadows. The snapshot for each quantum state $\rho(\mathbf{x})$ is fixed to be $T = 100$ and the number of trainable examples is set $n = 7500$. Once the dataset $\mathcal{T}_s$ is collected, we move to Step 3 of Alg. 1, where the four classical surrogates employ $\mathcal{T}_s$ to pretrain VQEs. The hyper-parameter settings for each learning surrogate are identical to those used in the TFIM tasks.



Supplementary Fig. 15: **Prediction error of four learning surrogates for the family of H_5 molecules with $N = 10$.** Each subplot corresponds to the simulation results for a specific bond length R . The solid bars, shaded bars, and x-axis labels follow the same representations as those in Fig. 12.

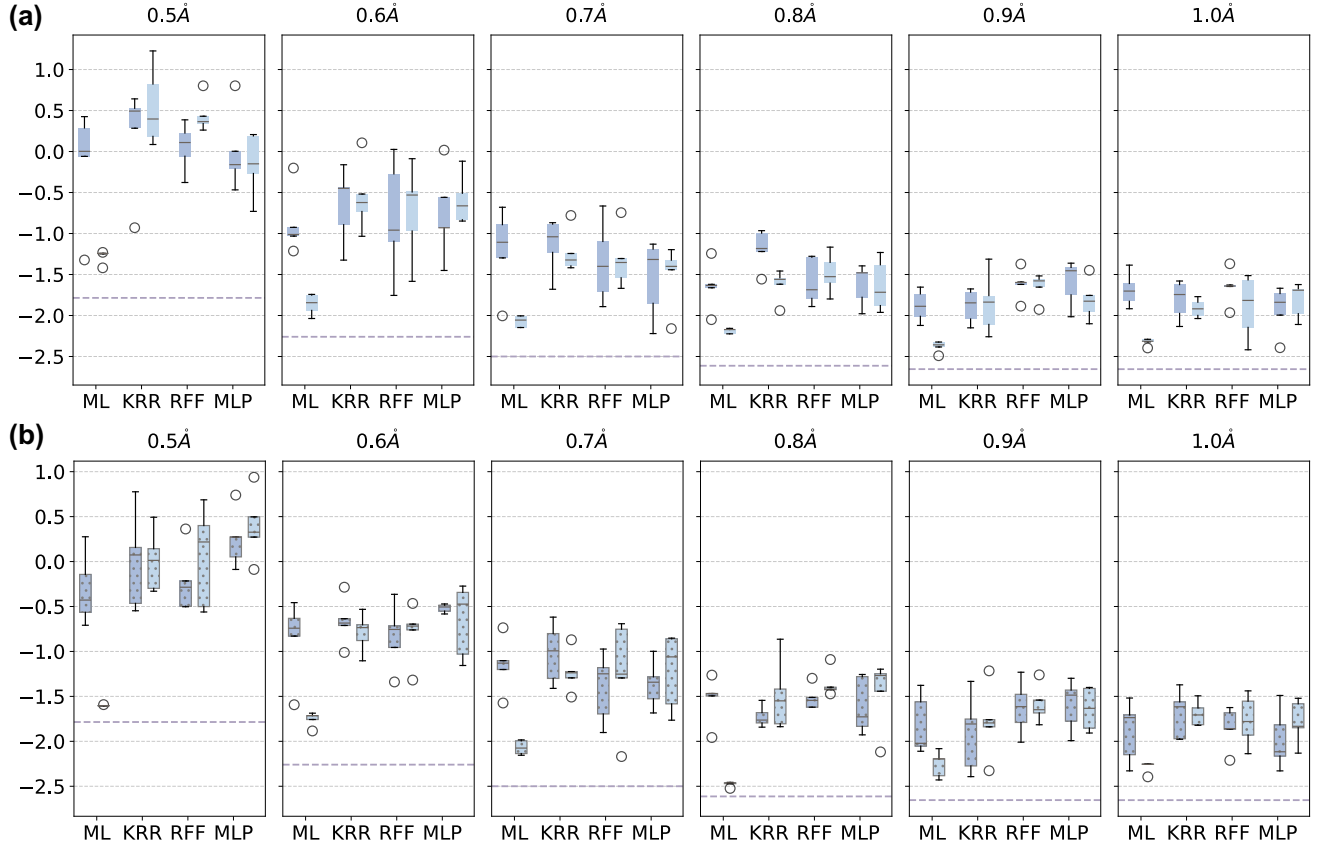
Numerical results. The predictor error R_{predict} of the employed four learning surrogates is shown in Fig. 15, which is evaluated by 500 test examples. For all explored bond lengths, the proposed ML model outperforms the other three learning surrogates under the ideal setting. Additionally, under the noisy setting, the proposed ML model performs slightly worse than MLP but remains superior to the other two learning surrogates. More specifically, the prediction error of the proposed ML model under the ideal (noisy) setting achieves 0.046 (0.319), 0.040 (0.286), 0.025 (0.266), 0.018 (0.239), 0.013 (0.222), and 0.012 (0.193) when $R = 0.5\text{\AA}, 0.6\text{\AA}, 0.7\text{\AA}, 0.8\text{\AA}, 0.9\text{\AA}, 1.0\text{\AA}$, respectively. The increased discrepancy between the ideal and shadow results, compared to the TFIM task, arises from the presence of numerous global Pauli terms in the H_5 molecules, which substantially amplify the Pauli-based shadow estimation error.

We proceed to use the optimized four learning surrogates to pretrain VQEs for the six explored H_5 molecules. The estimated energies, both with and without pretraining by the four learning surrogates, are illustrated in Fig. 16. The first key observation is that, under both ideal and noisy conditions, the parameters returned by the proposed ML model substantially improve estimation accuracy after pretraining. In contrast, other learning surrogates fail to reduce the estimation error through pretraining and, in some cases, even lead to degraded performance. Furthermore, while the shadowed dataset increases the test prediction error of the proposed ML model discussed in Fig. 15, it notably improves the estimation of the ground-state energy compared to both the ideal case and MLP, particularly when the bond length is $R = 0.8\text{\AA}$. These observations align with the results of Theorem 2 in the main text.

Supplementary Table 6: **Comparison between conventional VQEs and the proposed ML model on a family of H_5 with $N = 10$.** The table presents the estimated ground state energies for various coupling strengths J , as obtained from the conventional VQE and the proposed ML model under both ideal and noisy conditions (labeled by ‘Pretrain (ideal)’ and ‘Pretrain (shadow)’, respectively). The notation ‘ $a \pm b$ ’ indicates that the mean value is a and the variance is b .

	$R = 0.5\text{\AA}$	$R = 0.6\text{\AA}$	$R = 0.7\text{\AA}$	$R = 0.8\text{\AA}$	$R = 0.9\text{\AA}$	$R = 1.0\text{\AA}$
VQE	-1.747 ± 0.000	-2.219 ± 0.000	-2.45 ± 0.000	-2.555 ± 0.000	-2.584 ± 0.000	-2.571 ± 0.000
Pretrain (ideal)	-1.276 ± 0.005	-1.862 ± 0.013	-2.072 ± 0.004	-2.189 ± 0.001	-2.376 ± 0.004	-2.322 ± 0.002
Pretrain (shadow)	-1.601 ± 0.000	-1.752 ± 0.005	-2.069 ± 0.001	-2.475 ± 0.001	-2.257 ± 0.017	-2.283 ± 0.003

Finally, we compare the performance of the conventional VQE (using infinite measurements) with the proposed ML



Supplementary Fig. 16: **Pretraining performance of four learning surrogates on the family of H_5 molecules with $N = 10$.** The upper panel (a) and the lower panel (b) separately exhibit the simulation results under the ideal and noisy settings ($T \rightarrow \infty$ and $T = 100$). Each subplot corresponds to a specific bond length R in $H_{H_5}(R)$, and the box plots depict the estimated ground-state energy distributions for four learning surrogates. For each surrogate, the estimated energies without pre-training and after pre-training are visualized by the deep and light blue colors, respectively. Dashed horizontal lines represent the exact ground-state energies.

model. The statistical results are summarized in Table 5. A key observation is that the proposed ML model, under both ideal and noisy settings, generally produces energy estimates that are comparable to but slightly less accurate than those of the conventional VQE. This slight inferiority may be attributed to the increased gradient norm when the Hamiltonian involves more than 400 Pauli terms, which necessitates more training examples and larger truncation values to achieve improved performance.

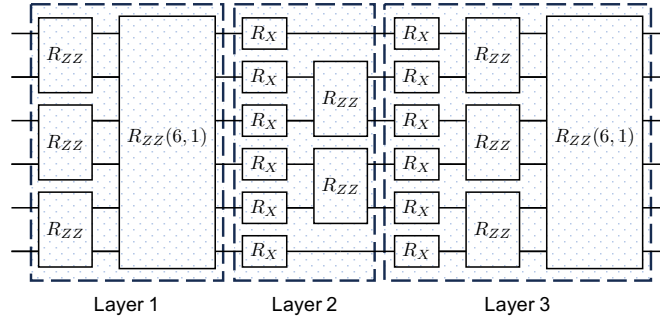
5. Comparison with LOWESA

We conduct the numerical simulations to compare the wall-clock runtime of our method and LOWESA using open-source implementations.

The problem setup follows the first scenario explained in SM J 2. The number of qubits is fixed to be $N = 32$. The employed ansatz $U(x)$ is an adaptation of the Trotterized transverse-field Ising Hamiltonian circuit, designed with a corresponding bricklayer topology. A visualization of this circuit is provided in Fig. 17. Specifically, the ansatz is designed with 50,000 layers. All angles of the R_{zz} gates and RX gates are set to zero, except for the 16th R_{zz} gate in the first layer, which is assigned arbitrary angles. As such, the explored ansatz is tailored to include a single tunable parameter. The initialized state is $|0\rangle^{\otimes N}$ and the observable employed takes the form as $O = \mathbb{I}^{\otimes 15} \otimes Z \otimes \mathbb{I}^{\otimes 16}$.

The hyper-parameter settings for LOWESA are as follows. The implementation is based on the open-source PauliPropagation.jl package available on GitHub. The truncation threshold for the weights is set to 5, and the minimum coefficient magnitude is set to 4. We provide 10 input angles, i.e.,

$$x \in \{-\pi, -0.8\pi, -0.6\pi, -0.4\pi, -0.2\pi, 0, 0.2\pi, 0.4\pi, 0.6\pi, 0.8\pi\},$$



Supplementary Fig. 17: **The employed ansatz with the bricklayer topology.** The figure visualizes the implementation of the employed ansatz $U(\mathbf{x})$ with three layers and $N = 6$ qubits. Each layer consists of alternating R_{ZZ} and R_X gates. The R_{ZZ} gate labeled $R_{ZZ}(6, 1)$ in the odd layers connects the 6-th and 1st qubits. For circuits with more than three layers, Layers 2 and 3 are alternated to construct deeper circuits while preserving the bricklayer topology. Specifically, the gate arrangement in Layer 2 is repeated for all even-numbered layers, while the gate arrangement in Layer 3 is repeated for all odd-numbered layers.

to the ansatz $U(x)$ and use LOWESA to estimate the corresponding expectation values.

The hyper-parameter settings of the proposed learning model h_s are as follows. The number of training examples is set as 1000. For simplicity, we consider the ideal scenario such that the infinite measurements are adopted to construct the dataset. Given access to the training dataset, we train the learning model by setting the truncation threshold Λ as 2. Once training is completed, we use the same 10 input angles with LOWESA as test examples to evaluate the prediction performance and inference time.

Supplementary Table 7: **Performance and wall-clock time of LOWESA and the proposed learning model.** The notation ' $a \pm b$ ' indicates that the mean value is a and the variance is b . The wall-clock time for the proposed learning approach focuses on the inference stage.

	LOWESA	Our work
Estimation (Prediction) error	0 ± 0.00	0 ± 0.0002
Wall-clock time (seconds)	2.003 ± 0.0107	$8.6808 \times 10^{-5} \pm 2.202 \times 10^{-8}$

The performance and computational efficiency between LOWESA and the proposed learning model are summarized in Table 7. Regarding the prediction (estimation) error, both approaches achieve comparable accuracy, with the proposed method showing a slight variance of 0.0002. However, in the wall-clock time, our proposed learning model demonstrates a dramatic reduction in inference time compared to LOWESA, achieving a reduction by approximately 2.3×10^4 times. This distinction underscores a scenario where the learning model surpasses classical simulators, emphasizing their complementary roles.

-
- [1] Haimeng Zhao, Laura Lewis, Ishaan Kannan, Yihui Quek, Hsin-Yuan Huang, and Matthias C Caro. Learning quantum states and unitaries of bounded gate complexity. *PRX Quantum*, 5(4):040306, 2024.
 - [2] 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.
 - [3] Hsin-Yuan Huang. Learning quantum states from their classical shadows. *Nature Reviews Physics*, 4(2):81–81, 2022.
 - [4] Andreas Elben, Steven T Flammia, Hsin-Yuan Huang, Richard Kueng, John Preskill, Benoît Vermersch, and Peter Zoller. The randomized measurement toolbox. *Nature Reviews Physics*, pages 1–16, 2022.
 - [5] Maria Schuld and Nathan Killoran. Is quantum advantage the right goal for quantum machine learning? *PRX Quantum*, 3:030101, Jul 2022.
 - [6] Ryan Sweke, Erik Recio-Armengol, Sofiene Jerbi, Elies Gil-Fuster, Bryce Fuller, Jens Eisert, and Johannes Jakob Meyer. Potential and limitations of random fourier features for dequantizing quantum machine learning. *Quantum*, 9:1640, 2025.
 - [7] Enrico Fontana, Manuel S Rudolph, Ross Duncan, Ivan Rungger, and Cristina Cirstoiu. Classical simulations of noisy variational quantum circuits. *arXiv preprint arXiv:2306.05400*, 2023.
 - [8] Manuel S Rudolph, Enrico Fontana, Zoë Holmes, and Lukasz Cincio. Classical surrogate simulation of quantum systems with lowesa. *arXiv preprint arXiv:2308.09109*, 2023.

- [9] Marco Cerezo, Andrew Arrasmith, Ryan Babbush, Simon C Benjamin, Suguru Endo, Keisuke Fujii, Jarrod R McClean, Kosuke Mitarai, Xiao Yuan, Lukasz Cincio, et al. Variational quantum algorithms. *Nature Reviews Physics*, 3(9):625–644, 2021.
- [10] Kishor Bharti, Alba Cervera-Lierta, Thi Ha Kyaw, Tobias Haug, Sumner Alperin-Lea, Abhinav Anand, Matthias Degroote, Hermann Heimonen, Jakob S Kottmann, Tim Menke, et al. Noisy intermediate-scale quantum algorithms. *Reviews of Modern Physics*, 94(1):015004, 2022.
- [11] Jarrod R McClean, Sergio Boixo, Vadim N Smelyanskiy, Ryan Babbush, and Hartmut Neven. Barren plateaus in quantum neural network training landscapes. *Nature communications*, 9(1):1–6, 2018.
- [12] 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):1–12, 2021.
- [13] Qiang Miao and Thomas Barthel. Equivalence of cost concentration and gradient vanishing for quantum circuits: an elementary proof in the riemannian formulation. *Quantum Science and Technology*, 9(4):045039, sep 2024.
- [14] Gregory Boyd and Bálint Koczor. Training variational quantum circuits with covar: covariance root finding with classical shadows. *Physical Review X*, 12(4):041022, 2022.
- [15] Stefan H Sack, Raimel A Medina, Alexios A Michailidis, Richard Kueng, and Maksym Serbyn. Avoiding barren plateaus using classical shadows. *PRX Quantum*, 3(2):020365, 2022.
- [16] Sofiene Jerbi, Casper Gyurik, Simon C Marshall, Riccardo Molteni, and Vedran Dunjko. Shadows of quantum machine learning. *Nature Communications*, 15(1):5676, 2024.
- [17] Steven T Flammia and Yi-Kai Liu. Direct fidelity estimation from few pauli measurements. *Physical review letters*, 106(23):230501, 2011.
- [18] Michał Horodecki, Paweł Horodecki, and Ryszard Horodecki. Separability of n-particle mixed states: necessary and sufficient conditions in terms of linear maps. *Physics Letters A*, 283(1-2):1–7, 2001.
- [19] 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.
- [20] Yanming Che, Clemens Gneiting, and Franco Nori. Exponentially improved efficient machine learning for quantum many-body states with provable guarantees. *Physical Review Research*, 6(3):033035, 2024.
- [21] Laura Lewis, Hsin-Yuan Huang, Viet T. Tran, Sebastian Lehner, Richard Kueng, and John Preskill. Improved machine learning algorithm for predicting ground state properties. *Nature Communications*, 15(1):895, 2024.
- [22] Cambyse Rouzé, Daniel Stilck França, Emilio Onorati, and James D Watson. Efficient learning of ground and thermal states within phases of matter. *Nature Communications*, 15(1):7755, 2024.
- [23] Emilio Onorati, Cambyse Rouzé, Daniel Stilck França, and James D Watson. Provably efficient learning of phases of matter via dissipative evolutions. *arXiv preprint arXiv:2311.07506*, 2023.
- [24] Anurag Anshu and Srinivasan Arunachalam. A survey on the complexity of learning quantum states. *Nature Reviews Physics*, pages 1–11, 2023.
- [25] Leonardo Banchi, Jason Luke Pereira, Sharu Theresa Jose, and Osvaldo Simeone. Statistical complexity of quantum learning. *Advanced Quantum Technologies*, page 2300311, 2023.
- [26] Hsin-Yuan Huang, Yunchao Liu, Michael Broughton, Isaac Kim, Anurag Anshu, Zeph Landau, and Jarrod R McClean. Learning shallow quantum circuits. pages 1343–1351, 2024.
- [27] Sofiene Jerbi, Joe Gibbs, Manuel S Rudolph, Matthias C Caro, Patrick J Coles, Hsin-Yuan Huang, and Zoë Holmes. The power and limitations of learning quantum dynamics incoherently. *arXiv preprint arXiv:2303.12834*, 2023.
- [28] Alba Cervera-Lierta, Jakob S Kottmann, and Alán Aspuru-Guzik. Meta-variational quantum eigensolver: Learning energy profiles of parameterized hamiltonians for quantum simulation. *PRX Quantum*, 2(2):020329, 2021.
- [29] He-Liang Huang, Yuxuan Du, Ming Gong, Youwei Zhao, Yulin Wu, Chaoyue Wang, Shaowei Li, Futian Liang, Jin Lin, Yu Xu, et al. Experimental quantum generative adversarial networks for image generation. *Physical Review Applied*, 16(2):024051, 2021.
- [30] Shi-Xin Zhang, Chang-Yu Hsieh, Shengyu Zhang, and Hong Yao. Neural predictor based quantum architecture search. *Machine Learning: Science and Technology*, 2(4):045027, 2021.
- [31] Nishant Jain, Brian Coyle, Elham Kashefi, and Niraj Kumar. Graph neural network initialisation of quantum approximate optimisation. *Quantum*, 6:861, 2022.
- [32] Cong Lei, Yuxuan Du, Peng Mi, Jun Yu, and Tongliang Liu. Neural auto-designer for enhanced quantum kernels. In *The Twelfth International Conference on Learning Representations*, 2024.
- [33] Enrico Fontana, Ivan Rungger, Ross Duncan, and Cristina Cîrstoiu. Efficient recovery of variational quantum algorithms landscapes using classical signal processing. *arXiv preprint arXiv:2208.05958*, 2022.
- [34] Nikita A. Nemkov, Evgeniy O. Kiktenko, and Aleksey K. Fedorov. Fourier expansion in variational quantum algorithms. *Phys. Rev. A*, 108:032406, Sep 2023.
- [35] Franz J Schreiber, Jens Eisert, and Johannes Jakob Meyer. Classical surrogates for quantum learning models. *Physical Review Letters*, 131(10):100803, 2023.
- [36] Jonas Landman, Slimane Thabet, Constantin Dalyac, Hela Mhiri, and Elham Kashefi. Classically Approximating Variational Quantum Machine Learning with Random Fourier Features, 2022. arXiv:2210.13200v1.
- [37] Berta Casas and Alba Cervera-Lierta. Multidimensional fourier series with quantum circuits. *Physical Review A*, 107(6):062612, 2023.
- [38] Beng Yee Gan, Po-Wei Huang, Elies Gil-Fuster, and Patrick Rebentrost. Concept learning of parameterized quantum models from limited measurements. *arXiv preprint arXiv:2408.05116*, 2024.
- [39] Giacomo Torlai, Guglielmo Mazzola, Juan Carrasquilla, Matthias Troyer, Roger Melko, and Giuseppe Carleo. Neural-

- network quantum state tomography. *Nature Physics*, 14(5):447–450, 2018.
- [40] Yuxuan Du, Yibo Yang, Tongliang Liu, Zhouchen Lin, Bernard Ghanem, and Dacheng Tao. Shadownet for data-centric quantum system learning. *arXiv preprint arXiv:2308.11290*, 2023.
 - [41] Shahnawaz Ahmed, Carlos Sánchez Muñoz, Franco Nori, and Anton Frisk Kockum. Quantum state tomography with conditional generative adversarial networks. *Physical Review Letters*, 127(14):140502, 2021.
 - [42] Peter Cha, Paul Ginsparg, Felix Wu, Juan Carrasquilla, Peter L McMahon, and Eun-Ah Kim. Attention-based quantum tomography. *Machine Learning: Science and Technology*, 3(1):01LT01, 2021.
 - [43] Giuseppe Carleo, Yusuke Nomura, and Masatoshi Imada. Constructing exact representations of quantum many-body systems with deep neural networks. *Nature communications*, 9(1):1–11, 2018.
 - [44] Juan Carrasquilla, Giacomo Torlai, Roger G Melko, and Leandro Aolita. Reconstructing quantum states with generative models. *Nature Machine Intelligence*, 1(3):155–161, 2019.
 - [45] Alistair WR Smith, Johnnie Gray, and MS Kim. Efficient quantum state sample tomography with basis-dependent neural networks. *PRX Quantum*, 2(2):020348, 2021.
 - [46] Tobias Schmale, Moritz Reh, and Martin Gärttner. Efficient quantum state tomography with convolutional neural networks. *npj Quantum Information*, 8(1):115, 2022.
 - [47] Xiaoqian Zhang, Maolin Luo, Zhaodi Wen, Qin Feng, Shengshi Pang, Weiqi Luo, and Xiaoqi Zhou. Direct fidelity estimation of quantum states using machine learning. *Physical Review Letters*, 127(13):130503, 2021.
 - [48] Yan Zhu, Ya-Dong Wu, Ge Bai, Dong-Sheng Wang, Yuexuan Wang, and Giulio Chiribella. Flexible learning of quantum states with generative query neural networks. *Nature Communications*, 13(1):6222, 2022.
 - [49] Amit Nir, Eran Sela, Roy Beck, and Yohai Bar-Sinai. Machine-learning iterative calculation of entropy for physical systems. *Proceedings of the National Academy of Sciences*, 117(48):30234–30240, 2020.
 - [50] Tailong Xiao, Jingzheng Huang, Hongjing Li, Jianping Fan, and Guihua Zeng. Intelligent certification for quantum simulators via machine learning. *npj Quantum Information*, 8(1):138, 2022.
 - [51] Ya-Dong Wu, Yan Zhu, Ge Bai, Yuexuan Wang, and Giulio Chiribella. Quantum similarity testing with convolutional neural networks. *Physical Review Letters*, 130(21):210601, 2023.
 - [52] Thomas M Cover. *Elements of information theory*. John Wiley & Sons, 1999.
 - [53] Ke Li. Second-order asymptotics for quantum hypothesis testing. *The Annals of Statistics*, 42(1):171 – 189, 2014.
 - [54] Jarrod R McClean, Jonathan Romero, Ryan Babbush, and Alán Aspuru-Guzik. The theory of variational hybrid quantum-classical algorithms. *New Journal of Physics*, 18(2):023023, 2016.
 - [55] Giacomo Torlai, Guglielmo Mazzola, Giuseppe Carleo, and Antonio Mezzacapo. Precise measurement of quantum observables with neural-network estimators. *Physical Review Research*, 2(2):022060, 2020.
 - [56] Alistair WR Smith, Kiran E Khosla, Chris N Self, and MS Kim. Qubit readout error mitigation with bit-flip averaging. *Science advances*, 7(47):eabi8009, 2021.
 - [57] Alexander Semenovich Holevo. Bounds for the quantity of information transmitted by a quantum communication channel. *Problemy Peredachi Informatsii*, 9(3):3–11, 1973.
 - [58] Ingemar Bengtsson and Karol Życzkowski. *Geometry of quantum states: an introduction to quantum entanglement*. Cambridge university press, 2017.
 - [59] Xinbiao Wang, Yuxuan Du, Zhuozhuo Tu, Yong Luo, Xiao Yuan, and Dacheng Tao. Transition role of entangled data in quantum machine learning. *Nature Communications*, 15(1):3716, 2024.
 - [60] Vladimir Vapnik. *The nature of statistical learning theory*. Springer science & business media, 1999.
 - [61] Matthias C Caro, Hsin-Yuan Huang, Marco 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.
 - [62] Hsin-Yuan Huang, Richard Kueng, and John Preskill. Information-theoretic bounds on quantum advantage in machine learning. *Physical Review Letters*, 126(19):190505, 2021.
 - [63] Yuxuan Du, Zhuozhuo Tu, Xiao Yuan, and Dacheng Tao. Efficient measure for the expressivity of variational quantum algorithms. *Physical Review Letters*, 128(8):080506, 2022.
 - [64] Yuxuan Du, Zhuozhuo Tu, Bujiao Wu, Xiao Yuan, and Dacheng Tao. Power of quantum generative learning. *arXiv preprint arXiv:2205.04730*, 2022.
 - [65] Scott Aaronson. The learnability of quantum states. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 463(2088):3089–3114, 2007.
 - [66] Riccardo Molteni, Casper Gyurik, and Vedran Dunjko. Exponential quantum advantages in learning quantum observables from classical data. *arXiv preprint arXiv:2405.02027*, 2024.
 - [67] John Watrous. Quantum computational complexity. *arXiv preprint arXiv:0804.3401*, 2008.
 - [68] Sanjeev Arora and Boaz Barak. *Computational complexity: a modern approach*. Cambridge University Press, 2009.
 - [69] G. Vidal and C. M. Dawson. Universal quantum circuit for two-qubit transformations with three controlled-not gates. *Phys. Rev. A*, 69:010301, Jan 2004.
 - [70] Michael A Nielsen and Isaac L Chuang. *Quantum computation and quantum information*. Cambridge University Press, 2010.
 - [71] Yuxuan Du, Yibo Yang, Dacheng Tao, and Min-Hsiu Hsieh. Problem-dependent power of quantum neural networks on multiclass classification. *Physical Review Letters*, 131(14):140601, 2023.
 - [72] Daniel Greenbaum. Introduction to quantum gate set tomography. *arXiv preprint arXiv:1509.02921*, 2015.
 - [73] Matthias C Caro. Learning quantum processes and hamiltonians via the pauli transfer matrix. *ACM Transactions on Quantum Computing*, 5(2):1–53, 2024.

- [74] Guillermo González-García, J Ignacio Cirac, and Rahul Trivedi. Pauli path simulations of noisy quantum circuits beyond average case. *arXiv preprint arXiv:2407.16068*, 2024.
- [75] Thomas Schuster, Chao Yin, Xun Gao, and Norman Y Yao. A polynomial-time classical algorithm for noisy quantum circuits. *arXiv preprint arXiv:2407.12768*, 2024.
- [76] Giacomo Torlai and Matthew Fishman. PastaQ: A package for simulation, tomography and analysis of quantum computers, 2020.
- [77] Abhinav Kandala, Antonio Mezzacapo, Kristan Temme, Maika Takita, Markus Brink, Jerry M Chow, and Jay M Gambetta. Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets. *nature*, 549(7671):242–246, 2017.
- [78] Vojtěch Havlíček, Antonio D Córcoles, Kristan Temme, Aram W Harrow, Abhinav Kandala, Jerry M Chow, and Jay M Gambetta. Supervised learning with quantum-enhanced feature spaces. *Nature*, 567(7747):209–212, 2019.
- [79] Amit Dutta, Gabriel Aeppli, Bikas K. Chakrabarti, Uma Divakaran, Thomas F. Rosenbaum, and Diptiman Sen. *Quantum Phase Transitions in Transverse Field Spin Models: From Statistical Physics to Quantum Information*. Cambridge University Press, 2015.
- [80] James Dborin, Vinul Wimalaweera, Fergus Barratt, Eric Ostby, Thomas E O’Brien, and Andrew G Green. Simulating groundstate and dynamical quantum phase transitions on a superconducting quantum computer. *Nature Communications*, 13(1):5977, 2022.
- [81] Elijah Pelofske, Andreas Bärtshi, and Stephan Eidenbenz. Short-depth qaoa circuits and quantum annealing on higher-order ising models. *npj Quantum Information*, 10(1):30, 2024.
- [82] Alexander Miessen, Daniel J Egger, Ivano Tavernelli, and Guglielmo Mazzola. Benchmarking digital quantum simulations above hundreds of qubits using quantum critical dynamics. *PRX Quantum*, 5(4):040320, 2024.
- [83] Youngseok Kim, Andrew Eddins, Sajant Anand, Ken Xuan Wei, Ewout Van Den Berg, Sami Rosenblatt, Hasan Nayfeh, Yantao Wu, Michael Zaletel, Kristan Temme, et al. Evidence for the utility of quantum computing before fault tolerance. *Nature*, 618(7965):500–505, 2023.
- [84] Sam McArdle, Suguru Endo, Alán Aspuru-Guzik, Simon C Benjamin, and Xiao Yuan. Quantum computational chemistry. *Reviews of Modern Physics*, 92(1):015003, 2020.
- [85] James I Colless, Vinay V Ramasesh, Dar Dahlen, Machiel S Blok, Mollie E Kimchi-Schwartz, Jarrod R McClean, Jonathan Carter, Wibe A de Jong, and Irfan Siddiqi. Computation of molecular spectra on a quantum processor with an error-resilient algorithm. *Physical Review X*, 8(1):011021, 2018.
- [86] Pei Liu, Ruixia Wang, Jing-Ning Zhang, Yingshan Zhang, Xiaoxia Cai, Huikai Xu, Zhiyuan Li, Jiaxiu Han, Xuegang Li, Guangming Xue, et al. Performing su (d) operations and rudimentary algorithms in a superconducting transmon qudit for d= 3 and d= 4. *Physical Review X*, 13(2):021028, 2023.
- [87] Maria Schuld, Ville Bergholm, Christian Gogolin, Josh Izaac, and Nathan Killoran. Evaluating analytic gradients on quantum hardware. *Physical Review A*, 99(3):032331, 2019.
- [88] Christopher M Bishop and Nasser M Nasrabadi. *Pattern recognition and machine learning*, volume 4. Springer, 2006.
- [89] Elies Gil-Fuster, Jens Eisert, and Carlos Bravo-Prieto. Understanding quantum machine learning also requires rethinking generalization. *Nature Communications*, 15(1):1–12, 2024.
- [90] Kosuke Mitarai, Makoto Negoro, Masahiro Kitagawa, and Keisuke Fujii. Quantum circuit learning. *Physical Review A*, 98(3):032309, 2018.
- [91] Adrián Pérez-Salinas, Alba Cervera-Lierta, Elies Gil-Fuster, and José I Latorre. Data re-uploading for a universal quantum classifier. *Quantum*, 4:226, 2020.
- [92] Iris Cong, Soonwon Choi, and Mikhail D Lukin. Quantum convolutional neural networks. *Nature Physics*, 15(12):1273–1278, 2019.
- [93] Johannes Herrmann, Sergi Masot Lima, Ants Remm, Petr Zapletal, Nathan A McMahon, Colin Scarato, François Swiadek, Christian Kraglund Andersen, Christoph Hellings, Sebastian Krinner, et al. Realizing quantum convolutional neural networks on a superconducting quantum processor to recognize quantum phases. *Nature communications*, 13(1):4144, 2022.
- [94] Diederik P Kingma. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- [95] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of machine learning*. MIT press, 2018.
- [96] Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. *Advances in neural information processing systems*, 20, 2007.
- [97] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.