Quantum phase classification via quantum hypothesis testing

Akira Tanji[1,*], **Hiroshi Yano**[2], **and Naoki Yamamoto**[1,2,+]

[1]Department of Applied Physics and Physico-Informatics, Keio University, Hiyoshi 3-14-1, Kohoku, Yokohama 223-8522, Japan
[2]Keio Quantum Computing Center, Keio University, Hiyoshi 3-14-1, Kohoku, Yokohama 223-8522, Japan
[*]tanjikeio@keio.jp
[+]yamamoto@appi.keio.ac.jp

arXiv:2504.04101v1 [quant-ph] 5 Apr 2025

ABSTRACT

Quantum phase classification is a fundamental problem in quantum many-body physics, traditionally approached using order parameters or quantum machine learning techniques such as quantum convolutional neural networks (QCNNs). However, these methods often require extensive prior knowledge of the system or large numbers of quantum state copies for reliable classification. In this work, we propose a classification algorithm based on the quantum Neyman-Pearson test, which is theoretically optimal for distinguishing between two quantum states. While directly constructing the quantum Neyman-Pearson test for many-body systems via full state tomography is intractable due to the exponential growth of the Hilbert space, we introduce a partitioning strategy that applies hypothesis tests to subsystems rather than the entire state, effectively reducing the required number of quantum state copies while maintaining classification accuracy. We validate our approach through numerical simulations, demonstrating its advantages over conventional methods, including the order parameter-based classifier and the QCNN. Our results show that the proposed method achieves lower classification error probabilities and significantly reduces the training cost compared to the QCNN and the recently developed classical machine learning algorithm enhanced with quantum data, while maintaining high scalability up to systems with 81 qubits. These findings highlight the potential of quantum hypothesis testing as a powerful tool for quantum phase classification, particularly in experimental settings where quantum measurements are combined with classical post-processing.

1 Introduction

Quantum phase classification[1–5] is a fundamental task for understanding the behavior of quantum many-body systems that undergo phase transitions. Unlike classical systems, where thermal fluctuations drive phase transitions, quantum phase transitions may occur even at zero temperature and are driven by changes in external parameters such as pressure, magnetic field, or chemical composition. These transitions are marked by quantum fluctuations and are characterized by changes in the ground state properties of the system. The study of quantum phase transitions provides insight into the critical behavior and universality classes that describe how different systems behave near critical points. A key tool for distinguishing different phases of matter is an order parameter[1–8], particularly in the context of symmetry-breaking. In many cases, phases can be classified based on whether the system exhibits a non-zero local order parameter, which signals spontaneous symmetry breaking. Mathematically, an order parameter o is defined as the expectation value of an observable O, given by $o = \lim_{L\to\infty}\langle O\rangle$, where $L \to \infty$ indicates the thermodynamic limit and the notation $\langle\cdot\rangle$ denotes the expectation value with respect to the ground state of the system in the case of quantum phases. The value of o characterizes the phase of the system, such that $o \neq 0$ if the system is ordered, and $o = 0$ if it is disordered. For example, the magnetization in a ferromagnet serves as an order parameter that distinguishes between the magnetically ordered and disordered phases[1–3]. However, this traditional framework of local order parameters falls short when it comes to identifying topological phases. Topological phases[7–10] of matter do not break any local symmetry and therefore cannot be characterized by a local order parameter. Instead, these phases are defined by global properties that reflect non-local correlations in the system. For example, topological phases exhibit topological invariants such as Chern numbers and $\hat{Z}$ invariants that remain robust under continuous deformations of the system[9,10].

To address the limitations of traditional approaches, quantum convolutional neural networks (QCNNs)[11–15] have been introduced. QCNNs are quantum circuit architectures inspired by classical convolutional neural networks[16,17], designed to analyze quantum data by applying a series of convolutional and pooling layers. QCNNs have been successfully applied to recognize various quantum phases, including symmetry-protected topological (SPT) phases, demonstrating their capability to identify complex quantum states. A notable feature of QCNNs is their resilience to the barren plateau problem, which ensures that they remain trainable with gradient-based methods even for large system sizes[14]. Additionally, QCNNs can be trained with small sets of labeled data, making them practical for tasks where training data is limited[18]. QCNNs have been implemented successfully on current quantum hardware, showing their feasibility for near-term quantum applications[15]. A notable subclass of QCNNs is Exact QCNNs[11,19,20], which provide an analytical solution for classifying quantum phases without requiring training. Numerical experiments have shown that Exact QCNNs exhibit advantages in sample complexity compared to traditional order parameter-based approaches, making them more effective in scenarios where minimizing the number of required measurements is crucial[11]. However, Exact QCNNs have the downside that they are known to exist only for a limited range of quantum phases[11,19,20].

Other quantum machine learning approaches have been developed that combine quantum experiments with classical processing. This approach belongs to the class of Classical Simulation enhanced with Quantum Experiments (CSIM$_{\text{QE}}$) studied in Ref.[21], where initial data acquisition is performed via quantum experiments in no more than polynomial time, producing polynomial-size data that enhances the classical simulation algorithm. One such approach involves the use of an efficient classical representation, known as the classical shadow[22], to construct a kernel function for quantum phase classification, among other applications[23]. Another approach is the low-weight Pauli propagation algorithm[24,25], which leverages classical shadows to efficiently simulate quantum circuits on classical hardware within the range of low-weight Pauli operators. This method has been successfully applied to quantum phase classification using QCNNs[25], referred to here as low-weight QCNNs. The study shows that QCNNs often operate within a classically simulable regime when restricted to low-body observables, raising questions about their quantum advantage. These methods demonstrate the potential of hybrid quantum-classical strategies for analyzing quantum phases while also emphasizing the need to identify problems where quantum computation offers a distinct advantage.

Following these lines of research, we propose a more simplified and efficient classification algorithm based on the quantum Neyman-Pearson test[26–28], which is known to be optimal for distinguishing between general two quantum states. The classical Neyman-Pearson framework has been applied in machine learning for classification problems, where it provides a principled way to control error probabilities in classification[29,30]. Given this background, we extend it to the quantum domain for quantum phase classification. While this approach enables quantum phase classification by leveraging quantum hypothesis testing to identify phase boundaries, directly constructing the quantum Neyman-Pearson test for quantum many-body states via full state tomography is challenging due to the exponential growth of the Hilbert space with system size. To address this issue, we introduce a partitioning strategy based on the partial tomography method[31,32] that decomposes a quantum many-body state into smaller subsystems, allowing hypothesis tests to be applied locally rather than to the entire state. Our method can be viewed as a classical simulation algorithm supported by quantum experiments, in the sense that it constructs measurements on test data through classical processing of information obtained via partial tomography, and thus belongs to the CSIM$_{\text{QE}}$ framework. This method reduces the number of copies of quantum states required for classification while improving classification accuracy by effectively extracting relevant phase information from local measurements, making our algorithm both scalable and highly reliable for near-term quantum experiments. We validate our approach through numerical simulations, demonstrating its advantages over conventional methods such as the order parameter, the QCNN, and the Exact QCNN. Our results show that it achieves lower classification error probabilities across various quantum phase classification tasks, which we attribute to the use of an approximately constructed quantum Neyman-Pearson test. Additionally, our method significantly improves training efficiency compared to the QCNN and the low-weight QCNN, as it does not rely on gradient-based variational learning but benefits from the approximate quantum Neyman-Pearson test. Notably, it achieves lower validation losses while requiring fewer training data copies used by the QCNN. Furthermore, it maintains high classification accuracy across different system sizes, including cases with up to 81 qubits. These results highlight the potential of our method as a practical and efficient approach for quantum phase classification in experimental settings where quantum experiments are combined with classical processing.

2 Methods

Quantum hypothesis testing (QHT)[26–28,33,34] is a fundamental tool in quantum information theory, designed to identify the true state of a quantum system from a set of possible states. Extending classical hypothesis testing principles to the quantum domain, QHT uses density operators to represent states and aims to develop optimal measurement strategies to minimize the probability of making incorrect decisions. This framework is essential for applications such as quantum communication and sensing. In the most basic scenario, QHT involves two hypotheses where the task is to distinguish between two potential states, ρ (the null hypothesis) and σ (the alternative hypothesis). A key challenge in this process is balancing and minimizing the types of errors that can occur. Specifically, the Type-I error happens when σ is chosen when ρ is true, while the Type-II error occurs when ρ is selected when σ is correct. In other words, the Type-I error happens when the alternative hypothesis σ is accepted despite the null hypothesis ρ being the true state, whereas the Type-II error occurs when the null hypothesis ρ is chosen even though the alternative hypothesis σ is the correct state. Mathematically, the Type-I error probability, denoted as α_n, is given by

$$\alpha_n = \mathrm{Tr}\big(\rho^{\otimes n}(I - M_n)\big) = 1 - \mathrm{Tr}\big(\rho^{\otimes n} M_n\big), \tag{1}$$

where $\{M_n, I - M_n\}$ are two-outcome Positive Operator-Valued Measures (POVMs) associated with deciding in favor of ρ and σ, respectively. Similarly, the Type-II error probability, denoted as β_n, is defined as

$$\beta_n = \mathrm{Tr}\big(\sigma^{\otimes n} M_n\big). \tag{2}$$

Note that, in the above definitions, quantum states are assumed to be prepared in multiple n copies, denoted by $\rho^{\otimes n}$ and $\sigma^{\otimes n}$, which are collectively or individually measured to distinguish between the two hypotheses. Measurement strategies must be carefully designed to manage these errors; in particular, symmetric approaches aim to minimize the total error probability, and asymmetric strategies focus on reducing one type of error while controlling the other within acceptable bounds. The quantum Neyman-Pearson test $\{S(n,a), I - S(n,a)\}$ is defined by the POVM element

$$S(n,a) = \big\{\rho^{\otimes n} - e^{na}\sigma^{\otimes n} > 0\big\}, \tag{3}$$

where a is a hyperparameter that controls the balance between Type-I and Type-II error probabilities. The notation $\{A > 0\}$ represents the projection operator onto the subspace spanned by the eigenvectors of a Hermitian operator A with positive eigenvalues, i.e.,

$$\{A > 0\} = \sum_{i:\lambda_i > 0} |\lambda_i\rangle\langle\lambda_i|\,, \quad \text{if } A = \sum_i \lambda_i\, |\lambda_i\rangle\langle\lambda_i|\,. \tag{4}$$

This test is well known in the theory of quantum hypothesis testing as the most powerful test for distinguishing two quantum states. That is, the test maximizes the power $1 - \beta_n$, which is the probability of correctly rejecting the null hypothesis ρ when the alternative hypothesis σ is true, while controlling the Type-I error probability α_n to be below a given significance level. The same holds if the roles of α_n and β_n are reversed. Moreover, this test is employed to achieve the asymptotic optimality described by the quantum Stein's lemma[35,36] and the quantum Hoeffding's theorem[37]. Note that, when the two quantum states ρ and σ commute, the quantum Neyman-Pearson test is equivalent to the classical Neyman-Pearson test as described in Appendix. However, constructing this measurement with full state tomography requires an exponential number of copies of the quantum states or classical processing with respect to system size, making it impractical for large-scale quantum systems.

To overcome this scalability issue, we propose a method for approximately constructing a quantum Neyman-Pearson test via the decomposition of a quantum many-body state into smaller subsystems followed by their partial state tomography. Applying this method to the quantum phase classification problem allows us to reduce the error probability for single-copy data near the phase boundary and the number of copies of data needed to construct the measurement. The key assumption is that many quantum phases could potentially be classified by dividing quantum many-body systems into groups of a few qubits. This stems from the fact that most condensed matter Hamiltonians exhibit few-body interactions, meaning they involve interactions that only extend among a few particles. Furthermore, many order parameters can be represented as linear combinations of local observables[1,2,6]. Even in topological phases, which are often characterized by global order parameters[9,10], classification can sometimes be achieved using few-body observables[23].

We here describe our classification method; the entire procedure is depicted in Fig. 1. The key concept is a partitioned quantum Neyman-Pearson test with POVM

$$\begin{aligned} S_j^{(0)}(n_{\mathrm{ent}}, a) &= \Big\{\rho_j^{\otimes n_{\mathrm{ent}}} - e^{n_{\mathrm{ent}}a}\sigma_j^{\otimes n_{\mathrm{ent}}} > 0\Big\}, \\ S_j^{(1)}(n_{\mathrm{ent}}, a) &= I - S_j^{(0)}(n_{\mathrm{ent}}, a), \end{aligned} \tag{5}$$

where ρ_j and σ_j denote the k-qubit reduced density matrices (k-RDMs) of the full quantum states ρ and σ corresponding to the j-th k-qubit subsystem. In the main text, we describe the method for the case $n_{\mathrm{ent}} = 1$; for the case $n_{\mathrm{ent}} > 1$, see the Additional numerical simulations section of the Appendix. The classifier, or equivalently the POVM corresponding to Eq. (5), is constructed using given training data (a set of quantum states). We then apply the constructed POVM to classify new data, which we call test data. More specifically, the training step proceeds as follows, as shown in Fig. 1(a):

1. Given N_{train} quantum many-body states $\{\rho^{(i)}\}_{i=1}^{N_{\mathrm{train}}}$ on L qubits as training data and corresponding binary labels $\{y^{(i)} \in \{0,1\}\}_{i=1}^{N_{\mathrm{train}}}$.

2. For each training data $\rho^{(i)}$, consider the k-RDMs $\{\rho_j^{(i)}\}_{j=1}^{L/k}$ and estimate them as $\{\hat{\rho}_j^{(i)}\}_{j=1}^{L/k}$ using partial tomography. (Here, we simply assumed that L is divisible by k.)

3. For each group j, calculate the ensemble average of matrices obtained via tomography for states labeled as 0, $\hat{\rho}_j = \sum_{i:y^{(i)}=0} p^{(i)}\hat{\rho}_j^{(i)}$, and for states labeled as 1, $\hat{\sigma}_j = \sum_{i:y^{(i)}=1} p^{(i)}\hat{\rho}_j^{(i)}$.

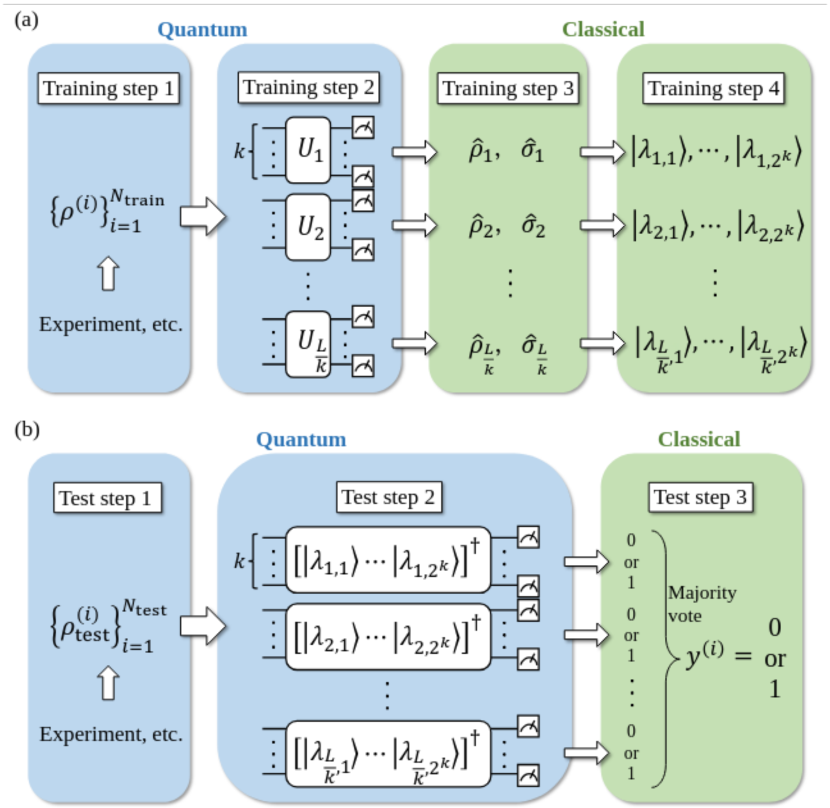
(a)
Quantum
Classical
Training step 1
Training step 2
Training step 3
Training step 4
$\{\rho^{(i)}\}_{i=1}^{N_{\mathrm{train}}}$
Experiment, etc.
k
U_1
U_2
$U_{\frac{L}{k}}$
$\hat{\rho}_1,\ \hat{\sigma}_1$
$\hat{\rho}_2,\ \hat{\sigma}_2$
$\hat{\rho}_{\frac{L}{k}},\ \hat{\sigma}_{\frac{L}{k}}$
$|\lambda_{1,1}\rangle,\cdots,|\lambda_{1,2^k}\rangle$
$|\lambda_{2,1}\rangle,\cdots,|\lambda_{2,2^k}\rangle$
$|\lambda_{\frac{L}{k},1}\rangle,\cdots,|\lambda_{\frac{L}{k},2^k}\rangle$
(b)
Quantum
Classical
Test step 1
Test step 2
Test step 3
$\{\rho_{\mathrm{test}}^{(i)}\}_{i=1}^{N_{\mathrm{test}}}$
Experiment, etc.
k
$[|\lambda_{1,1}\rangle\cdots|\lambda_{1,2^k}\rangle]^\dagger$
$[|\lambda_{2,1}\rangle\cdots|\lambda_{2,2^k}\rangle]^\dagger$
$[|\lambda_{\frac{L}{k},1}\rangle\cdots|\lambda_{\frac{L}{k},2^k}\rangle]^\dagger$
0 or 1
0 or 1
0 or 1
Majority vote
$y^{(i)} =$ 0 or 1

Figure 1. Schematic depiction of our method in the (a) training step and (b) test step. Steps within the blue boxes represent quantum processes, while those within the green boxes represent classical processes. In the training step (a), we perform partial tomography on all the quantum many-body states $\{\rho^{(i)}\}$ in a training dataset, obtaining their k-RDMs. In the test step (b), for each test state $\rho_{\mathrm{test}}^{(i)}$, the approximate quantum Neyman-Pearson test is conducted to obtain a prediction $y^{(i)}$.

4. For each group j, compute the eigenvalues $\{\lambda_{j,l}\}_{l=1}^{2^k}$ and eigenvectors $\{|\lambda_{j,l}\rangle\}_{l=1}^{2^k}$ of $\hat{\rho}_j - e^a\hat{\sigma}_j$, and construct the gate implementation of the unitary matrix $[|\lambda_{j,1}\rangle,\ldots,|\lambda_{j,2^k}\rangle]^\dagger$ to perform the partitioned quantum Neyman-Pearson test with POVMs $S_j^{(0)}(n_{\mathrm{ent}}=1,a) = \{\hat{\rho}_j - e^a\hat{\sigma}_j > 0\} = \sum_{l:\lambda_{j,l}>0}|\lambda_{j,l}\rangle\langle\lambda_{j,l}|$ and $S_j^{(1)}(n_{\mathrm{ent}}=1,a) = \sum_{l:\lambda_{j,l}\leq 0}|\lambda_{j,l}\rangle\langle\lambda_{j,l}|$, which are created based on the estimators $\hat{\rho}_j$ and $\hat{\sigma}_j$.

The test step proceeds as shown in Fig. 1(b):

1. Given N_{test} quantum many-body states $\{\rho_{\mathrm{test}}^{(i)}\}_{i=1}^{N_{\mathrm{test}}}$ on L qubits as test data.

2. For each test data $\rho_{\mathrm{test}}^{(i)}$, consider the k-RDMs $\{\rho_{\mathrm{test},j}^{(i)}\}_{j=1}^{L/k}$, and for each group j, perform the POVMs $\{S_j^{(0)}(n_{\mathrm{ent}} = 1,a),\ S_j^{(1)}(n_{\mathrm{ent}} = 1,a)\}$ corresponding to the quantum Neyman-Pearson test on each group, utilizing the unitary matrix obtained in Training step 4.

3. For each i, take a majority vote of the measurement results across all groups $j = 1,\ldots,L/k$, classifying the test data as $y^{(i)} = 0$ if the former POVM is more frequently measured, and as $y^{(i)} = 1$ otherwise.

In Training step 1, we assume that quantum many-body states serving as training data can be prepared on a quantum device using quantum state preparation algorithms or experimental methods, with known corresponding phase labels. We obtain estimates of k-RDMs $\{\hat{\rho}_j^{(i)}\}_{j=1}^{L/k}$ via partial tomography techniques, such as a classical shadow[22]. Training step 3 involves matrix calculations on a classical computer, where ensemble averaging corresponds to replacing composite quantum hypothesis tests with simple ones, with probabilities $p^{(i)}$ typically uniform in phase classification tasks but potentially variable. Training step 4 also involves classical matrix calculations, constructing unitaries for performing the approximate quantum Neyman-Pearson test, the gate implementation of which can be efficiently constructed for small k-qubit groups[38,39]. Here, a is an arbitrary real number, which is set to 0 when no preference is given to either quantum phase. Although it is possible to set $n_{\text{ent}} > 1$, numerical results presented in the Additional numerical simulations section of the Appendix indicate that this had limited significance.

In Test step 1, we assume that test data can be prepared on a quantum device similarly to training data. Test step 2 implements the two-outcome POVMs for each group, as shown in Fig. 1(b), by measuring in the basis $\{|\lambda_{j,l}\rangle\}_{l=1}^{2^k}$ and post-selecting based on whether the eigenvalue $\lambda_{j,l}$ of the measured basis state is positive or not. Test step 3 completes the classification of the quantum phase by taking the majority vote POVM across all groups, which aggregates the results of partitioned quantum Neyman-Pearson tests applied to k-RDMs. It is defined as

$$M_n = \Pi_{\text{maj}}^{\otimes(n/n_{\text{ent}})}(n_{\text{ent}}, a), \tag{6}$$

where n_{ent} is the same number in Eq. (5), i.e., the number of copies used per subsystem in each individual Neyman-Pearson test. The majority vote POVM element is given by

$$\Pi_{\text{maj}}(n_{\text{ent}}, a) = \sum_{\substack{x_1,\ldots,x_{L/k} \\ \sum_j x_j < L/2k}} \bigotimes_{j=1}^{L/k} S_j^{(x_j)}(n_{\text{ent}}, a), \tag{7}$$

where L is the entire system size. In this step, each j-th k-qubit subsystem is tested independently using a partitioned quantum Neyman-Pearson test. Then, the majority vote POVM element $\Pi_{\text{maj}}(n_{\text{ent}}, a)$ selects the outcome for which the majority of the subsystems yield $S_j^{(0)}(n_{\text{ent}}, a)$. The introduction of the majority vote is motivated by the absence of a clear procedure for aggregating measurement results obtained from partitioned quantum Neyman-Pearson tests. Since these tests are applied independently to each k-qubit subsystem, a majority vote offers a simple and natural way to combine local information into a global decision. Although a rigorous justification for this choice is not provided, it serves as a reasonable starting point for our method.

3 Numerical results

This section presents numerical results for quantum phase classification using various methods from the three aspects: error probabilities, training costs, and scalability. More precisely, we first evaluate the Type-I and Type-II error probabilities to assess classification performance. We then compare our method with the QCNN and the low-weight QCNN[24,25], in terms of the training costs. Furthermore, we examine the scalability of our method by analyzing its performance across different system sizes up to 81 qubits. In the Appendix, the settings of simulations and additional numerical simulations are provided.

3.1 Model and settings

We conduct numerical simulations for quantum phase classification of the one-dimensional cluster-Ising model

$$H = \sum_{i=1}^{L} \left(X_i - J_1 Z_i Z_{i+1} - J_2 Z_{i-1} X_i Z_{i+1} \right), \tag{8}$$

where $X_i (Z_i)$ are the Pauli $X(Z)$ operators on the i-th qubit. J_1 and J_2 are tunable coupling coefficients. The ground states of this many-body Hamiltonian, as shown in Fig. 2, exhibit four distinct phases: ferromagnetic (FM), antiferromagnetic (AFM), symmetry-protected topological (SPT), and trivial[40]. The order parameter for the FM phase

$$O_{\text{FM}} = \frac{1}{L} \sum_{i=1}^{L} Z_i \tag{9}$$

is a linear combination of local observables, and the trivial phase is similarly detected by a linear combination of local observables. In contrast, the order parameter for the SPT phase is expressed by a global observable

$$O_{\text{SPT}} = Z_1 X_2 X_4 \ldots X_{L-3} X_{L-1} Z_L, \tag{10}$$

which characterizes the topology of the system[7,8]. We refer to the classification between the trivial and FM phases as *Trivial vs. FM*, and the classification between the trivial and SPT phases as *Trivial vs. SPT*. For each case, we employed five methods for quantum phase classification: order parameter, QCNN, Exact QCNN, low-weight QCNN, and our method. The test data, which are quantum states we want to phase classify, are common across all methods and consist of 100 ground states near the phase boundaries depicted in each of Fig. 2(a) and Fig. 2(b), respectively. To calculate the error probabilities, we take the average error probabilities for all test data in each phase, meaning that we replace the composite hypotheses with simple ones. The training data, needed to construct the QCNN and our method, comprise 20 ground states shown in Fig. 2(a) and Fig. 2(b), with labels assigned as $y^{(i)} = 0$ for the trivial phase and $y^{(i)} = 1$ for the other phase. For numerical simulations of large system sizes ($L = 27$ and 81 qubits), we utilize Matrix Product States (MPS)[41,42]. We employ the finite-size Density Matrix Renormalization Group (DMRG) algorithm[41,42] with the maximum bond dimension 200 to prepare approximate ground states in the MPS representation. On the other hand, for small system sizes ($L = 15$ qubits), we use state vectors and the exact diagonalization algorithm.

Our method uses $k = 2$ qubits per group in the case of Trivial vs. FM, and $k = 3$ qubits in the case of Trivial vs. SPT. This choice is due to the fact that the FM phase and the SPT phase in the model described above can be classified using low-body observables[23]. For dividing the quantum many-body state, we assign k neighboring qubits to the same group, starting from one end of the chain, since the model is one-dimensional. While this grouping approach is straightforward for one-dimensional models, it is not for models in two or higher dimensions. However, we show in the Additional numerical simulations section of the Appendix that even a simple dividing strategy works sufficiently well for two-dimensional models. We employ partial tomography as Training step 2, utilizing a classical shadow[22] (-like) approach for its rapid convergence facilitated by random measurements. More precisely, for each partitioned group j, unitary operators $\{U_j\}$ are sampled uniformly at random from the Haar measure over the unitary group $\mathscr{U}(2^k)$, followed by computational basis measurements to obtain snapshots $\{(2^k+1)U_j^\dagger |\hat{b}\rangle\langle\hat{b}| U_j - \mathbb{I}\}$. The sample average of these snapshots directly forms $\hat{\rho}_j^{(i)}$, constructing it efficiently from these measurements. The term '-like' indicates that these snapshots and their averages are computed using matrix calculations, not through stabilizer states. Additionally, it is shown that random Clifford unitaries, rather than random Haar unitaries, are sufficient for this approach in the Additional numerical simulations section of the Appendix.

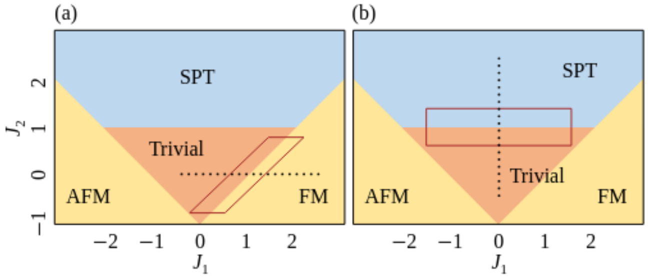

Figure 2. Quantum phase diagram of the ground state of the Hamiltonian in Eq. (8), along with the training and test data for (a) Trivial vs. FM and (b) Trivial vs. SPT cases. The 20 black dots represent the training data, while the test data consist of 100 randomly selected points within the red box near the phase boundary.

3.2 Error probabilities

We calculate the Type-I and Type-II error probabilities, α_n defined in Eq. (1) and β_n in Eq. (2), for the test data of quantum phase classification in the cases of Trivial vs. FM and Trivial vs. SPT, using four methods: order parameter, QCNN, Exact QCNN, and our method. The results for a 27-qubit system ($L = 27$) are shown in Fig. 3. Since lower error probabilities with fewer copies indicate better performance, methods represented in the lower left of Fig. 3(a), and lower positions in Fig. 3(b), are preferable. The total number of training data copies used for our method is 600 or 2,400; more precisely, for the former case, training shots $= N_{\text{train}} \times T_{\text{state}} = 20 \times 30 = 600$, where T_{state} is the number of shots per state (or equivalently the number of snapshot for constructing the shadow) in the training dataset. In contrast, the QCNN requires exact calculations of output expectation values to ignore the estimation errors, resulting in a total of $N_{\text{train}} \times N_{\text{epoch}} \times 2 = 20 \times 150 \times 2 = 6{,}000$ evaluations, where N_{epoch} is the number of epochs used for training the QCNN and ×2 means two times expectation estimation involved in the optimizer described below. The training cost for our method is therefore significantly lower compared to that of the QCNN. Additionally, the Simultaneous Perturbation Stochastic Approximation (SPSA) optimizer[43,44] is used for training the QCNN, where we estimate two expectation values to evaluate the gradient of the loss function.

Note that the problem settings differ between those for the order parameter and the Exact QCNN, and those for our method and the QCNN. The order parameter and the Exact QCNN operate without training data but with prior knowledge about the quantum phases, whereas our method and the QCNN rely on training data without any prior information on the phase. In the former setting, for the Trivial vs. SPT case, the Exact QCNN demonstrates significant improvements in error probabilities compared to the order parameter. Yet, for the Trivial vs. FM case, the order parameter combined with a Bayesian test (described in the Details of methods section of the Appendix) using an appropriately chosen prior distribution achieves lower error probabilities than the Exact QCNN. In the latter setting, in both cases, our method demonstrates substantial improvements over the QCNN in terms of error probabilities, despite using significantly fewer training data copies (as well as training costs, detailed in the next subsection). While a direct comparison must be made with caution due to the above differences in problem settings, our method shows great potential for outperforming the order parameter and the Exact QCNN in terms of error probabilities for both cases. Particularly for the Trivial vs. SPT case, it is especially remarkable that our method is expected to achieve much lower error probabilities than the commonly supported method, the order parameter. These results are obtained for a small number of test data copies, specifically $n = 1$ or at most 20. However, as suggested by Fig. 3(b), our method is anticipated to achieve better performance than the other methods even for larger numbers of test data copies.

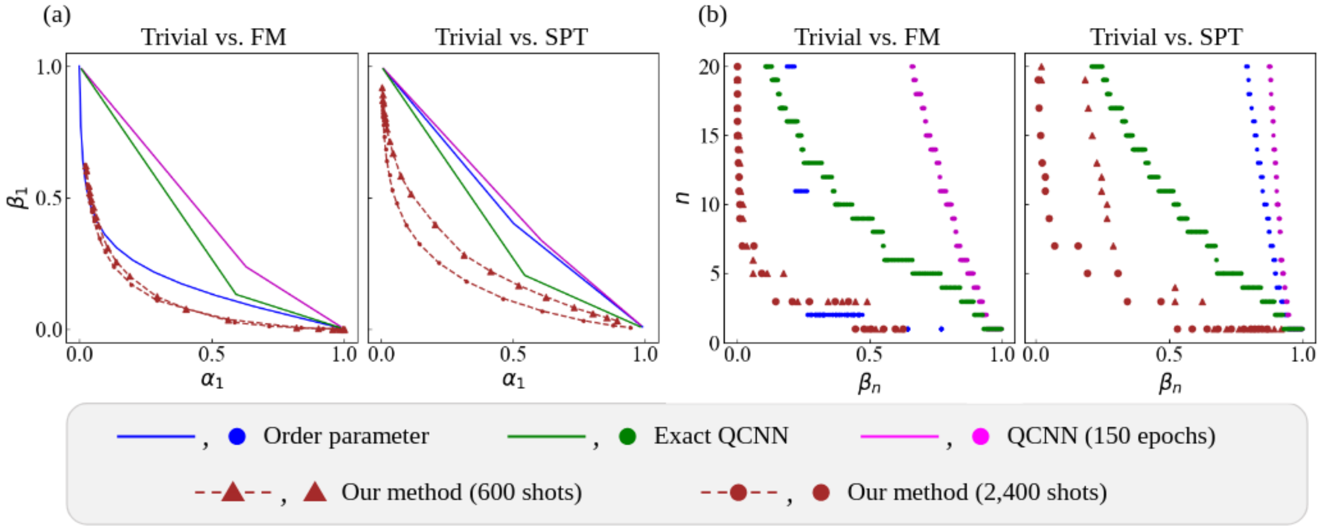

Figure 3. Type-I and Type-II error probabilities, α_n and β_n, for the test data in the order parameter, Exact QCNN, QCNN, and our method on $L = 27$ qubits in the Trivial vs. FM case and Trivial vs. SPT case. Panel (a) shows the error probabilities α_1 and β_1 for a single-copy test dataset ($n = 1$), whereas Panel (b) shows the number of test data copies n required to achieve β_n under the condition $\alpha_n \leq 5\,\%$. For our method, the hyperparameter a in Eq. (3) and Training step 4 is selected from 20 evenly spaced points in $[-1, 1]$.

3.3 Training costs

We present the validation loss, expressed as the Mean Squared Error (MSE) loss between the test data and validation labels with respect to the number of training data copies, evaluated across several cases:

$$\mathrm{MSE} = \frac{1}{N_{\mathrm{test}}} \sum_{i=1}^{N_{\mathrm{test}}} \left(f(\rho_{\mathrm{test}}^{(i)}) - y_{\mathrm{test}}^{(i)} \right)^2, \tag{11}$$

where $y_{\mathrm{test}}^{(i)} \in \{0, 1\}$ is the validation label for the test data $\rho_{\mathrm{test}}^{(i)}$, and where $f(\rho_{\mathrm{test}}^{(i)})$ denotes the probability that $\rho_{\mathrm{test}}^{(i)}$ is classified as validation label $y_{\mathrm{test}}^{(i)} = 1$.

We compare our method with the QCNN, in terms of the number of training data. In our method, the probability of classifying a test label $y_{\mathrm{test}}^{(i)} = 1$ is set as $f(\rho_{\mathrm{test}}^{(i)})$ in Eq. (11), using $a = 0$ to give equal weight to both labels. For the QCNN, the output expectation value is used as $f(\rho_{\mathrm{test}}^{(i)})$. The resultant learning curves for a 15-qubit system ($L = 15$) are shown in Fig. 4. Note that each of the four panels has different scales and ranges for the horizontal and vertical axes. Note also that the QCNN uses 1,000 copies of training data per expectation value estimation in this experiment (i.e., $T_{\mathrm{est}} = 1{,}000$). More precisely, Panel (a) shows results for our method, with training shots calculated as $N_{\mathrm{train}} \times T_{\mathrm{state}} = 20 \times T_{\mathrm{state}}$, shown on an axis in units of 10^3. Panel (b) shows results for the QCNN, with training shots calculated as $N_{\mathrm{train}} \times T_{\mathrm{est}} \times N_{\mathrm{epoch}} \times 2 = 20 \times 1{,}000 \times N_{\mathrm{epoch}} \times 2 = 40{,}000 \times N_{\mathrm{epoch}}$, shown on an axis in units of 10^6. These results show that our method achieves lower validation losses while using fewer than one-thousandth of the training data copies required by the QCNN. This significant improvement is attributed to the fact that gradient-based variational training methods generally require a large number of training data copies, which is not needed for our method.

Next, we compare our method with the low-weight QCNN, in terms of the required number of training data. The learning curves for a 15-qubit system ($L = 15$) for both our method and the low-weight QCNN are shown in Fig. 5; the results for our method are the same as those in Fig. 4, while the total training shots of the low-weight QCNN is calculated as

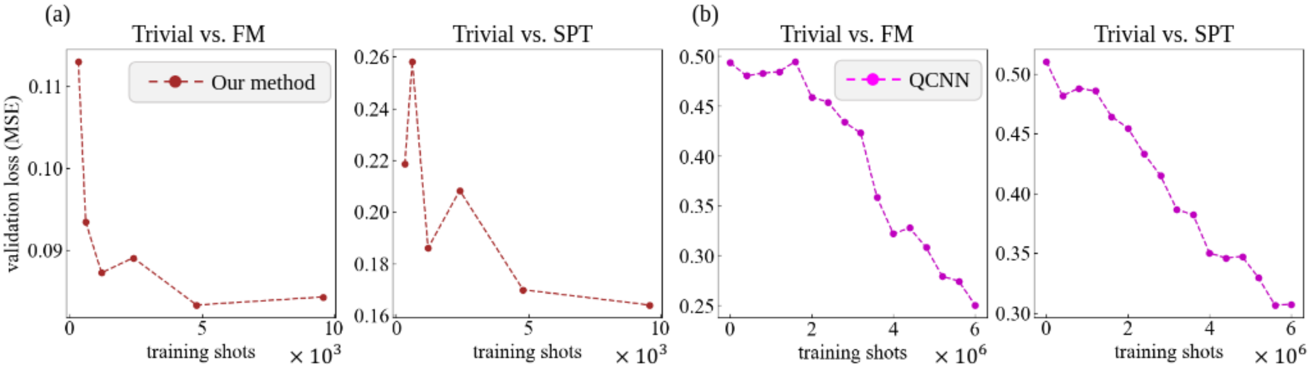

Figure 4. Learning curves for our method and the QCNN on $L = 15$ qubits in the Trivial vs. FM case and Trivial vs. SPT case. Panels (a) and (b) show the results for our method and the QCNN, respectively. The training shots represent the total number of training data, and the validation loss (MSE) is defined in Eq. (11).

$N_{\text{train}} \times T_{\text{state}} = 20 \times T_{\text{state}}$. This figure indicates that our method achieves a lower validation loss compared to the low-weight QCNN with the same number of training data. Furthermore, the low-weight QCNN requires fewer training data than the QCNN shown in Fig. 4 to reach comparable validation loss levels.

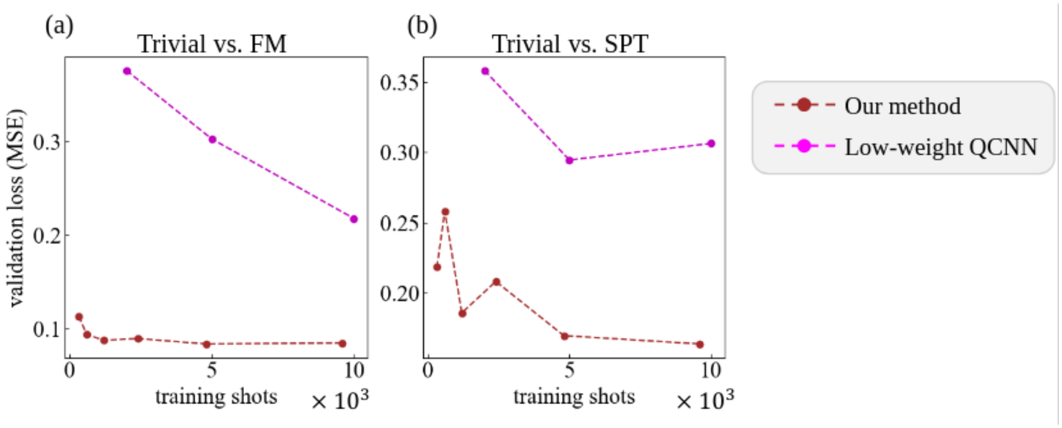

Figure 5. Learning curves for our method and the low-weight QCNN on $L = 15$ qubits. Panel (a) corresponds to the Trivial vs. FM case, and Panel (b) to the Trivial vs. SPT case; both are shown in units of 10^3. The results for our method are the same as those in Fig. 4.

We also compare our method and the low-weight QCNN, in terms of the classical computational time complexity required to construct the measurement process. In our method, the system size L is divided into groups of k qubits. For each group, classical processing involves taking an ensemble average of matrices obtained from partial tomography (Training step 3) and performing an eigendecomposition to construct the gate implementation for the quantum Neyman-Pearson test (Training step 4). This classical processing is repeated L/k times for each group. The ensemble averaging for a k-qubit matrix naively requires $\mathcal{O}\left(2^{2k}\right)$ time, eigendecomposition requires $\mathcal{O}\left(2^{3k}\right)$ time[45, 46], and constructing a general unitary gate implementation naively requires $2^{\mathcal{O}(k)}$ time[38]. Thus, the classical computational time complexity of our method is $\frac{L}{k} \times \left(\mathcal{O}\left(2^{2k}\right) + \mathcal{O}\left(2^{3k}\right) + 2^{\mathcal{O}(k)}\right) = 2^{\mathcal{O}(k)}L$. On the other hand, simulating a QCNN of system size L (typically with $\mathcal{O}(\log(L))$ depth[11]) up to weight k' for the low-weight QCNN requires $\mathcal{O}\left(L^{k'} \log(L)\right)$ time for each expectation value calculation[24]. Our method therefore has a lower classical computational time complexity than the low-weight QCNN. For example, if constant k and k' are chosen, the low-weight QCNN runs in $\mathcal{O}(\text{poly}(L)\log(L))$ time for each expectation value calculation, whereas our method runs in $\mathcal{O}(L)$ time.

3.4 Scalability

Finally, we examine the scalability of our method by comparing its Type-I and Type-II error probabilities of the test data for different system sizes L. The number of copies for the quantum Neyman-Pearson test is fixed to $n = 1$. The results for systems with $L = 15$, 27, and 81 qubits are shown in Fig. 6. For each system size, only 600 copies of training data are used, and the other settings such as the hyperparameter a are identical to those for our method in Fig. 3.

Fig. 6 suggests that our method achieves lower error probabilities as the system size increases. This trend probably stems from the fact that quantum phase transitions become clearer in larger spin chains, as suggested by the definition of order parameters $o = \lim_{L\to\infty}\langle O\rangle$. Intuitively, while partial tracing over all but small groups of qubits would lead to a nearly maximally mixed state, in practice, information appears to remain in each small group even for large systems, such as the $L = 81$ qubits case. Determining which types of quantum phases exhibit this property remains a non-trivial question and is one of the key challenges in assessing the applicability of our method to complex and large-scale systems.

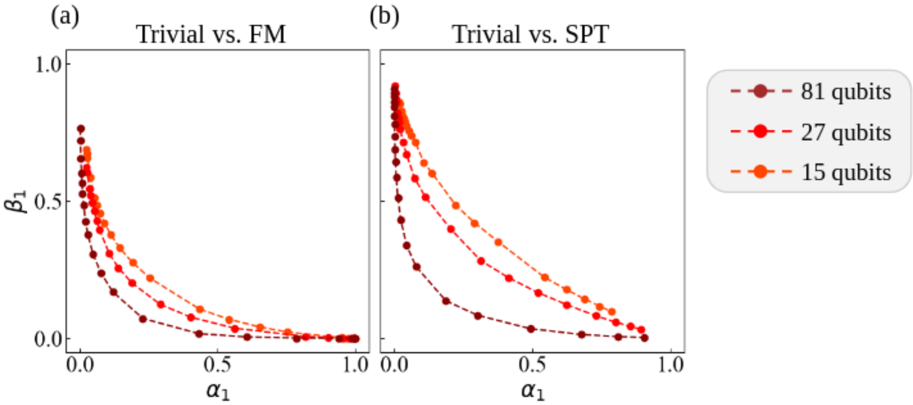

Figure 6. Type-I and Type-II error probabilities, α_1 and β_1, for the test data in our method on $L = 15$, 27, 81 qubits. Panel (a) corresponds to the Trivial vs. FM case, while Panel (b) represents the Trivial vs. SPT case.

4 Summary and discussion

This work has considered a scenario where quantum many-body states are available on quantum devices, either experimentally or through quantum state preparation algorithms. We have proposed a method to classify quantum phases by segmenting the quantum many-body system into small groups of qubits, performing partial tomography on each group, and constructing the quantum Neyman-Pearson test (the optimal strategy for distinguishing between two quantum states in quantum hypothesis testing) for each group.

We conducted a numerical analysis on quantum phase classification by performing partial tomography and utilizing the resulting tomographic data and the prior information about phases to construct measurements for classifying an unknown state. Partial tomography is efficient with respect to the system size; however, it has the downside that it cannot acquire global information about a quantum state, which may hinder accurate phase classification. Nevertheless, this study numerically demonstrates that leveraging partial tomographic data effectively enables a more efficient classification of many quantum phases in many-body quantum states compared to existing methods. In particular, we observe the following results: our method yields lower error probabilities than the QCNN with fewer copies of training data. Furthermore, without prior knowledge of the quantum phase, our approach attains lower error probabilities than the Exact QCNN and the order parameter. The partial tomography performs adequately with a classical shadow(-like) approach based on random Clifford unitaries. In comparison with the low-weight QCNN, our method achieves lower error probabilities with a similar number of training data and improves the required classical processing time from polynomial-log factor in system size L to linear time. Our approach demonstrates adequate performance for large systems of up to 81 qubits.

Our future challenges include investigating several fundamental aspects of our method as a machine learning model in greater detail. Specifically, we aim to explore generally achievable error probabilities for quantum phase classification (generalization), the methods of partial tomography that reduce error probabilities with fewer training data copies (trainability), and the effectiveness when applied to larger quantum many-body systems with more complex quantum phases (expressivity). Regarding expressivity, our method may be limited to classifying quantum phases that can be distinguished by low-body observables with respect to quantum states. This limitation is likely similar to the scope of applicability of the classical shadow-based method described in Refs.[23,25], where classification is efficient when phases are identifiable through low-body information. Therefore, investigating whether our method effectively generalizes to more complex phases, such as those with long-range entanglement or topological order, remains a particularly important direction for future research. Additionally, recent developments in QML have encountered obstacles, such as barren plateaus, and we suggest using quantum experiments and classical processing as one approach to overcome this barrier. To advance these efforts and further promote the development of QML, we will continue research in the hybrid domain of QML and QHT.

Data availability

The datasets generated during and analysed during the current study are available in the following GitHub repository: https://github.com/Tanji-A/Quantum-phase-classification-via-quantum-hypothesis-testing.

References

1. Sachdev, S. Quantum Phase Transitions (Cambridge University Press, 2011), second edition. edn.
2. Vojta, M. Quantum phase transitions. Reports on Prog. Phys. **66**, 2069 (2003).
3. Belitz, D. & Kirkpatrick, T. R. Quantum Phase Transitions, 399–424 (Springer Netherlands, 2000).
4. Shopova, D. V. & Uzunov, D. I. Some basic aspects of quantum phase transitions. Phys. Reports **379**, 1–67 (2003).
5. Sondhi, S. L., Girvin, S. M., Carini, J. P. & Shahar, D. Continuous quantum phase transitions. Rev. Mod. Phys. **69**, 315–333 (1997).
6. Shi, Q. Q., Zhou, H. Q. & Batchelor, M. T. Universal order parameters and quantum phase transitions: A finite-size approach. Sci. Reports **5**, 7673 (2015).
7. Haegeman, J., Pérez-García, D., Cirac, I. & Schuch, N. Order parameter for symmetry-protected phases in one dimension. Phys. Rev. Lett. **109**, 050402 (2012).
8. Pollmann, F. & Turner, A. M. Detection of symmetry-protected topological phases in one dimension. Phys. Rev. B **86**, 125441 (2012).
9. Haldane, F. D. M. Nobel lecture: Topological quantum matter. Rev. Mod. Phys. **89**, 040502 (2017).
10. Qi, X. L. & Zhang, S. C. Topological insulators and superconductors. Rev. Mod. Phys. **83**, 1057–1110 (2011).
11. Cong, I., Choi, S. & Lukin, M. D. Quantum convolutional neural networks. Nat. Phys. **15**, 1273–1278 (2019).
12. Monaco, S., Kiss, O., Mandarino, A., Vallecorsa, S. & Grossi, M. Quantum phase detection generalization from marginal quantum neural network models. Phys. Rev. B **107**, L081105 (2023).
13. Liu, Y. J., Smith, A., Knap, M. & Pollmann, F. Model-independent learning of quantum phases of matter with quantum convolutional neural networks. Phys. Rev. Lett. **130**, 220603 (2023).
14. Pesah, A. et al. Absence of barren plateaus in quantum convolutional neural networks. Phys. Rev. X **11**, 041011 (2021).
15. Herrmann, J. et al. Realizing quantum convolutional neural networks on a superconducting quantum processor to recognize quantum phases. Nat. Commun. **13**, 4144 (2022).
16. LeCun, Y. & Bengio, Y. Convolutional networks for images, speech, and time series, 255–258 (MIT Press, 1998).
17. Lecun, Y., Bengio, Y. & Hinton, G. Deep learning. Nature **521**, 436–444 (2015).
18. Caro, M. C. et al. Generalization in quantum machine learning from few training data. Nat. Commun. **13**, 4919 (2022).
19. Sander, L. C., McMahon, N. A., Zapletal, P. & Hartmann, M. J. Quantum convolutional neural network for phase recognition in two dimensions. Preprint at http://arxiv.org/abs/2407.04114 (2024).
20. Lake, E., Balasubramanian, S. & Choi, S. Exact quantum algorithms for quantum phase recognition: Renormalization group and error correction. PRX Quantum **6**, 10350 (2025).
21. Cerezo, M. et al. Does provable absence of barren plateaus imply classical simulability? or, why we need to rethink variational quantum computing. Preprint at http://arxiv.org/abs/2312.09121 (2023).
22. Huang, H. Y., Kueng, R. & Preskill, J. Predicting many properties of a quantum system from very few measurements. Nat. Phys. **16**, 1050–1057 (2020).
23. Huang, H. Y., Kueng, R., Torlai, G., Albert, V. V. & Preskill, J. Provably efficient machine learning for quantum many-body problems. Science **377**, eabk3333 (2022).
24. Angrisani, A. et al. Classically estimating observables of noiseless quantum circuits. Preprint at http://arxiv.org/abs/2409.01706 (2024).
25. Bermejo, P. et al. Quantum convolutional neural networks are (effectively) classically simulable. Preprint at http://arxiv.org/abs/2408.12739 (2024).

26. Helstrom, C. W. Quantum detection and estimation theory. J. Stat. Phys. **1**, 231–252 (1969).

27. Hayashi, M. Quantum Hypothesis Testing and Discrimination of Quantum States, 69–91 (Springer Berlin Heidelberg, 2006).

28. Nagaoka, H. & Hayashi, M. An information-spectrum approach to classical and quantum hypothesis testing for simple hypotheses. IEEE Transactions on Inf. Theory **53**, 534–549 (2007).

29. Tong, X., Feng, Y. & Li, J. J. Neyman-pearson classification algorithms and np receiver operating characteristics. Sci. Adv. **4**, eaao1659 (2018).

30. Scott, C. & Nowak, R. A neyman-pearson approach to statistical learning. IEEE Transactions on Inf. Theory **51**, 3806–3819 (2005).

31. Zhao, A., Rubin, N. C. & Miyake, A. Fermionic partial tomography via classical shadows. Phys. Rev. Lett. **127**, 110504 (2021).

32. Bonet-Monroig, X., Babbush, R. & O'Brien, T. E. Nearly optimal measurement scheduling for partial tomography of quantum states. Phys. Rev. X **10**, 031064 (2020).

33. Tej, J. P., Ahmed, S. R., Devi, A. R. U. & Rajagopal, A. K. Quantum hypothesis testing and state discrimination. Preprint at http://arxiv.org/abs/1803.04944 (2018).

34. de Boer, J., Godet, V., Kastikainen, J. & Keski-Vakkuri, E. Quantum hypothesis testing in many-body systems. SciPost Phys. Core **4**, 019 (2021).

35. Hiai, F. & Petz, D. The proper formula for relative entropy and its asymptotics in quantum probability. Commun. Math. Phys. **143**, 99–114 (1991).

36. Ogawa, T. & Nagaoka, H. Strong converse and stein's lemma in quantum hypothesis testing. IEEE Transactions on Inf. Theory **46**, 2428–2433 (2000).

37. Ogawa, T. & Hayashi, M. On error exponents in quantum hypothesis testing. IEEE Transactions on Inf. Theory **50**, 1368–1372 (2004).

38. Dawson, C. M. & Nielsen, M. A. The solovay-kitaev algorithm. Quantum Inf. Comput. **6**, 81–95 (2006).

39. Ge, Y. et al. Quantum circuit synthesis and compilation optimization: Overview and prospects. Preprint at http://arxiv.org/abs/2407.00736 (2024).

40. Verresen, R., Moessner, R. & Pollmann, F. One-dimensional symmetry protected topological phases and their transitions. Phys. Rev. B **96**, 165124 (2017).

41. Schollwöck, U. The density-matrix renormalization group. Rev. Mod. Phys. **77**, 259–315 (2005).

42. Vidal, G. Classical simulation of infinite-size quantum lattice systems in one spatial dimension. Phys. Rev. Lett. **98**, 070201 (2007).

43. Spall, J. C. Implementation of the simultaneous perturbation algorithm for stochastic optimization. IEEE Transactions on Aerosp. Electron. Syst. **34**, 817–823 (1998).

44. Spall, J. C. An overview of the simultaneous perturbation method for efficient optimization. Johns Hopkins APL Technical Digest (Applied Physics Laboratory) **19**, 482–492; https://secwww.jhuapl.edu/techdigest/Content/techdigest/pdf/V19-N04/19-04-Spall.pdf (1998).

45. Francis, J. G. F. The qr transformation a unitary analogue to the lr transformation–part 1. The Comput. J. **4**, 265–271 (1961).

46. Francis, J. G. F. The qr transformation–part 2. The Comput. J. **4**, 332–345 (1962).

47. Mezzadri, F. How to generate random matrices from the classical compact groups. Notices Am. Math. Soc. **54**, 592–604 (2007).

48. Bravyi, S. & Maslov, D. Hadamard-free circuits expose the structure of the clifford group. IEEE Transactions on Inf. Theory **67**, 4546–4563 (2021).

49. Dusuel, S., Kamfor, M., Orús, R., Schmidt, K. P. & Vidal, J. Robustness of a perturbed topological phase. Phys. Rev. Lett. **106**, 107203 (2011).

50. Fisher, R. A. The design of experiments (Oliver & Boyd, 1935).

51. Neyman, J. & Pearson, E. S. Ix. on the problem of the most efficient tests of statistical hypotheses. Philos. Transactions Royal Soc. London. Ser. A, Containing Pap. a Math. or Phys. Character **231**, 289–337 (1933).

52. Karlin, S. & Rubin, H. The theory of decision procedures for distributions with monotone likelihood ratio. The Annals Math. Stat. **27**, 272–299 (1956).

53. Lehmann, E. L. & Romano, J. P. Testing Statistical Hypotheses (Springer Science & Business Media, 2006).

54. Vidal, G. Class of quantum many-body states that can be efficiently simulated. Phys. Rev. Lett. **101**, 110501 (2008).

Acknowledgements

This work is supported by MEXT Quantum Leap Flagship Program Grants No. JPMXS0118067285 and No. JPMXS0120319794 and JST Grant No. JPMJPF2221.

A Settings of numerical simulations

We first describe how phase classification was performed using the methods employed in the numerical results presented in the main text.

Order parameter. The classification using the order parameter, detailed in Section C.2, involves applying projective measurements defined by the observables O_{FM} for the Trivial vs. FM case and O_{SPT} for the Trivial vs. SPT case in the main text. These measurements are performed on the test data, and are utilized for testing whether the expectation values of the order parameter are zero or not, using the Bayesian test for the Trivial vs. FM case and the classical Neyman-Pearson test for the Trivial vs. SPT case.

Exact QCNN. The Exact QCNNs used in our numerical simulation are shown in Fig. 7. The Exact QCNN for the FM phase was developed in this study, whereas the one for the SPT phase was taken from Ref.[11]. Both were constructed based on theoretical insights, such as order parameters and the renormalization group. For classification using the Exact QCNN, also detailed in Section C.3, the test data are input into the circuit shown in Fig. 7(a) for the Trivial vs. FM case and in Fig. 7(b) for the Trivial vs. SPT case. The output expectation values are then evaluated using the classical Neyman-Pearson test to test whether they exceed 0.5. Test data with output values above 0.5 are classified as label $y^{(i)} = 1$, and those not exceeding 0.5 as label $y^{(i)} = 0$.

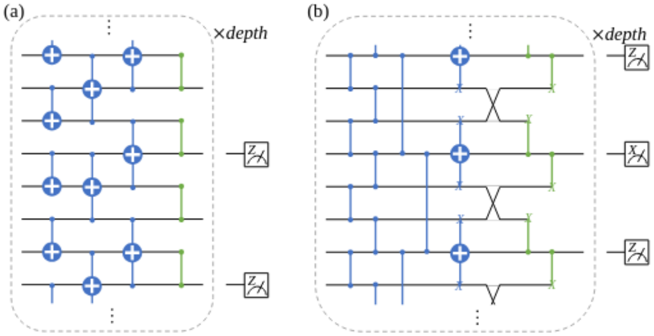

Figure 7. Exact QCNN circuits for (a) the FM and (b) SPT phases. The details of (a) are provided in Section C.3, while (b) is proposed in Ref.[11]. The circuit consists of convolutional layers (blue) and pooling layers (green, reduce the qubits), repeated for a specified depth, followed by Pauli measurements on the remaining qubits.

QCNN. The QCNN is trained using the ansatz shown in Fig. 8, initialized randomly. The training data are input, and the circuit parameters θ are updated over epochs using the SPSA optimizer to minimize the MSE loss

$$\mathrm{MSE}(\theta) = \frac{1}{N_{\mathrm{train}}} \sum_{i=1}^{N_{\mathrm{train}}} \left(f(\theta, \rho^{(i)}) - y^{(i)} \right)^2, \tag{12}$$

where $f(\theta, \rho^{(i)})$ denotes the output expectation value when the training data $\rho^{(i)}$ is input into the QCNN parameterized by θ. This training is performed for both the Trivial vs. FM and the Trivial vs. SPT cases. The trained QCNN classifies the test data using the classical Neyman-Pearson test, following the same procedure as described for the Exact QCNN: output values above 0.5 are assigned label $y^{(i)} = 1$, and those not exceeding 0.5 are assigned label $y^{(i)} = 0$.

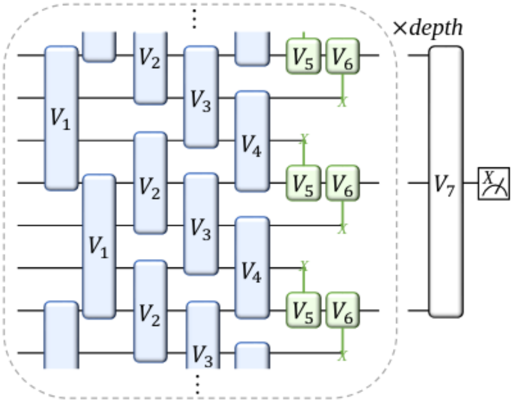

Figure 8. QCNN ansatz proposed in Ref.[11]. The circuit consists of convolutional layers (blue) and pooling layers (green, reduce the qubits), repeated for a specified depth, followed by fully connected layer (black) and Pauli measurements on the remaining qubits. The unitaries are parameterized as $V = \exp(-i\sum_j \theta_j \Lambda_j)$, where $\{\Lambda_j\}$ are generalized Gell-Mann matrices and $\{\theta_j\}$ are real parameters.

low-weight QCNN. The low-weight QCNN uses classical shadow snapshots of the training data obtained through quantum experiments and is trained by classical simulation using only the low-weight components of the Heisenberg-evolved observables, i.e., Pauli basis elements with a small number of non-identity qubits. Since the snapshots are collected once and can be reused for any number of epochs, the low-weight QCNN is trained for 300 epochs, where the validation loss reaches a stable level. The low-weight QCNN limit the weight of Pauli basis elements up to three, employs random Pauli measurements for classical shadow, and the other settings such as the ansatz and loss function follow the same as the QCNN.

B Additional numerical simulations

Here, we present additional results to complement the numerical findings presented in the main text.

B.1 Training costs with different tomography methods

We compare the methods used in partial tomography for Training step 2 of our method in terms of the required number of training data copies. Recall that a classical shadow(-like) approach with random measurements was employed for partial tomography. While the numerical results in the main text show results obtained using random Haar unitaries generated by classical algorithms as proposed in Ref.[47], implementing accurately random Haar unitaries is challenging. Hence, we also consider measurements based on random Clifford unitaries, generated by classical algorithms as described in Ref.[48], which are easier to implement on small groups of qubits.

The learning curves for a 15-qubit system ($L = 15$) using both random Haar and random Clifford unitaries are shown in Fig. 9. The figure demonstrates that training progresses at nearly the same rate with either unitary type, indicating that random Clifford unitaries provide sufficient performance when using the classical shadow(-like) approach. This result likely stems from dividing the system into small groups of qubits, which reduces the influence of the tomography method on convergence speed. However, what measurements are suitable for rapid reduction of error probabilities and verification loss is non-trivial and is one of our challenging problems.

B.2 Error probabilities for $n_{\text{ent}} > 1$

The approximate quantum Neyman-Pearson test $\{S_j^{(0)}(n_{\text{ent}}, a), S_j^{(1)}(n_{\text{ent}}, a)\}$ in the main text involves an entangled measurement on n_{ent} copies of quantum states. However, all numerical simulations in the main text were performed with $n_{\text{ent}} = 1$, as our method requires the eigendecomposition of kn_{ent} qubits, which becomes exponentially difficult with increasing n_{ent}. This section demonstrates that increasing n_{ent} beyond 1 provides limited benefits.

For $n_{\text{ent}} > 1$, the training steps 1 to 3 remain unchanged. In training step 4, the eigenvalues and eigenvectors of $\hat{\rho}_j^{\otimes n_{\text{ent}}} - e^{n_{\text{ent}} a} \hat{\sigma}_j^{\otimes n_{\text{ent}}}$ are calculated instead of $\hat{\rho}_j - e^a \hat{\sigma}_j$. In test step 2, measurements using the POVMs corresponding to the quantum Neyman-Pearson test are performed on n_{ent} copies of the test data.

For $n_{\text{ent}} = 1$ and 3, the Type-I and Type-II error probabilities for $n = 3$ copies of test data, α_3 and β_3, are shown in Fig. 10. In the partial tomography of training data, if an infinite number of copies are used, the resulting states are obtained by performing partial traces that retain only each k-qubit group. The infinite-shots results therefore correspond to the quantum

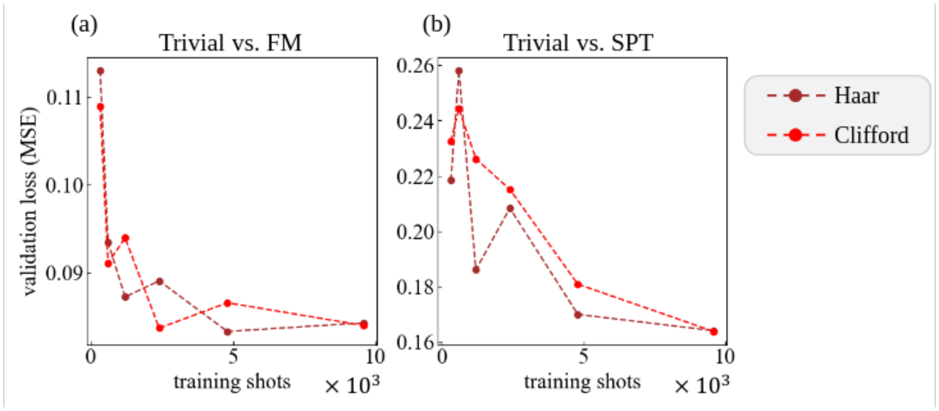

Figure 9. Learning curves for $L = 15$ qubits using random Haar and random Clifford unitaries. Panel (a) corresponds to the Trivial vs. FM case, and panel (b) to the Trivial vs. SPT case; both are shown in units of 10^3. The results for Haar are the same as those for our method in main text, while the results for Clifford follow the same settings as Haar except for the unitaries used in partial tomography.

Neyman-Pearson test constructed using these partial traced states for each k-qubit group. Fig. 10 indicates that there is no significant difference in error probabilities between $n_{\text{ent}} = 1$ and 3, regardless of finite or infinite shots and whether the system size is $L = 15$ or 27 qubits. However, for $n_{\text{ent}} = 3$, eigendecomposition and gate construction for the quantum Neyman-Pearson test are required for $kn_{\text{ent}} = 2 \times 3 = 6$ qubits, resulting in a significant increase in classical computational complexity. While larger n_{ent} might reduce error probabilities, the exponential increase in classical computational time with n_{ent} makes $n_{\text{ent}} = 1$ the most practical choice.

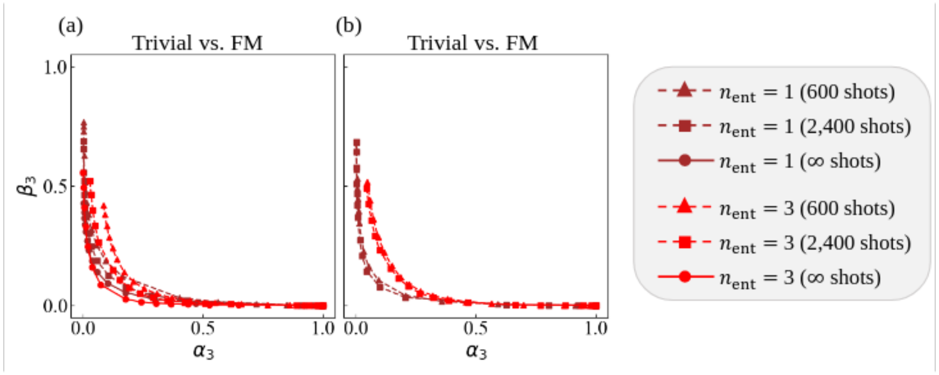

Figure 10. Type-I and Type-II error probabilities, α_3 and β_3, for $n = 3$ copies of test data with $n_{\text{ent}} = 1$ and 3. Panel (a) results for an $L = 15$ qubit system, and panel (b) results for an $L = 27$ qubit system, both in the Trivial vs. FM case ($k = 2$). The total number of training data copies is 600 or 2,400, with infinite-copy results also included for (a).

B.3 Two-dimensional model

In this section, we demonstrate that our method could work effectively for two-dimensional models where the dividing strategy is non-trivial, even with a simple dividing method. We conduct numerical simulations for quantum phase classification of the two-dimensional Toric code Hamiltonian with magnetic fields

$$H = H_{\text{TC}} - h_X \sum_{i=1}^{L} X_i - h_Z \sum_{i=1}^{L} Z_i, \tag{13}$$

where X_i (Z_i) are the Pauli X (Z) operators on the i-th qubit, and h_X (h_Z) are the tunable strengths of the magnetic fields in the X (Z) direction. Here, the Toric code Hamiltonian is

$$H_{\text{TC}} = -\sum_p A_p - \sum_s B_s, \tag{14}$$

where $A_p = \prod_{i \in \text{plaquette}(p)} X_i$ and $B_s = \prod_{i \in \text{star}(s)} Z_i$ are the plaquette and star operators, respectively. The ground states of this Hamiltonian undergo a quantum phase transition between the topological phase of the Toric code and the trivial magnetic phase[19,49]. For instance, changing h_Z at $h_X = 0$ induces a phase transition at $h_Z = 0.34$, and at $h_X = 0.1$, the phase transition occurs at $h_Z = 0.35$[19]. We classify the quantum phases using the Exact QCNN and our method. The test data, which are quantum states we want to phase classify, are common between the two methods and consist of 100 ground states near $h_X = 0.1$ and $h_Z = 0.35$. The training data required for our method are ground states along $h_X = 0$ with h_Z evenly divided into 20 points in $[0, 0.68]$, with labels assigned as $y^{(i)} = 0$ for the trivial magnetic phase and $y^{(i)} = 1$ for the topological phase. For numerical simulations, we use the MPS mapped to one dimension, as illustrated by the red dash-dot line in Fig. 11(a). Approximate ground states are prepared using the finite-size DMRG algorithm with a maximum bond dimension of 1500.

We describe the settings for each method.

Exact QCNN. For classification using the Exact QCNN, detailed in Section C.3, the test data are input into the circuit proposed in Ref.[19], and then the output expectation values are tested whether they exceed 0.5 using the classical Neyman-Pearson test. Test data with output values above 0.5 are classified as label $y^{(i)} = 1$, and those not exceeding 0.5 as label $y^{(i)} = 0$. The limited system size in our numerical simulation ($L = 18$ qubits) restricts the circuit depth, which may hinder the Exact QCNN from achieving its expected performance.

Our method. For our method, we use $k = 2$ qubit groups and the algorithm described in the Methods section of the main text. We divide the quantum many-body states into groups of $k = 2$ qubits based on the one-dimensional ordering shown in Fig. 11(a), where neighboring qubits are grouped sequentially from one end of the chain. Other settings, such as the partial tomography method, follow the configurations in the main text.

The Type-I and Type-II error probabilities, α_n and β_n, for the test data on $L = 18$ qubits are shown in Figs. 11(b)(c). These figures indicate that our method achieves sufficient performance even with simple dividing strategy. Note that the Exact QCNN is provided with prior knowledge of quantum phases but no training data, whereas our method uses training data without prior knowledge of the quantum phases, and this difference in problem setup precludes a direct comparison of performance.

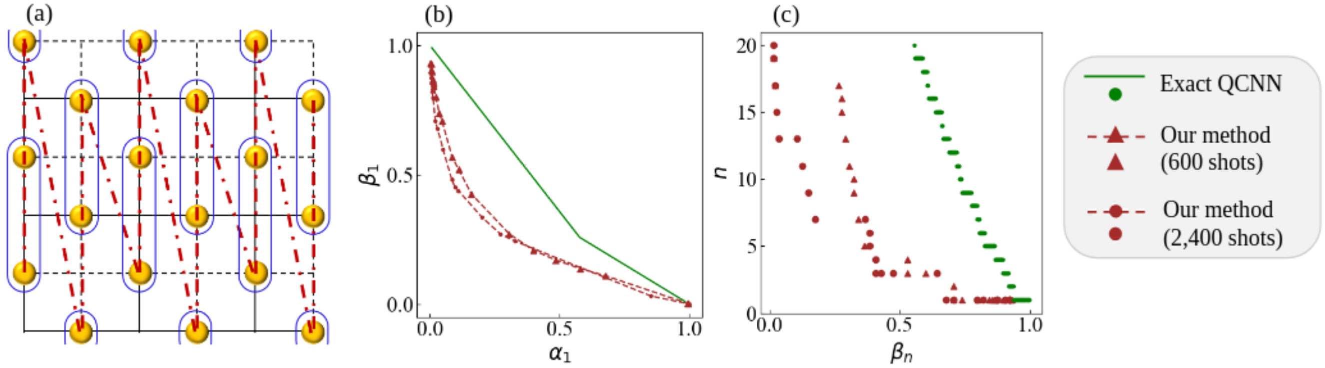

Figure 11. (a) Toric code lattice with $L = 18$ qubits. The red dashed-dotted line indicates the mapping order to one dimension for the MPS representation, and the blue circles represent the dividing strategy used in our method. (b)(c) Type-I and Type-II error probabilities, α_n and β_n, for the test data in the Exact QCNN and our method on $L = 18$ qubits. Panel (b) shows the error probabilities α_1 and β_1 for a single-copy test dataset ($n = 1$), whereas panel (c) shows the number of test data copies n required to achieve β_n under the condition $\alpha_n \leq 5\%$. The total number of training data copies used for our method is 600 or 2,400, while the Exact QCNN is the circuit proposed in Ref.[19], which does not require training. It is possible that the Exact QCNN does not perform sufficiently due to the structural constraints on the $L = 18$ qubit system size.

C Details of methods other than our method

Here, we provide detailed explanations of the methods other than our method used for comparison in the numerical simulations presented in the main text.

C.1 Classical hypothesis testing

Classical hypothesis testing[50–53] serves as a core approach in statistical inference, developed to evaluate competing hypotheses based on sample data. In this framework, two hypotheses are considered: the null hypothesis H_0, which represents a default state or baseline assumption, and the alternative hypothesis H_1, which represents an effect or deviation from the baseline. Classical hypothesis testing procedures are used to make decisions under uncertainty, particularly in fields like scientific research, quality control, and medical diagnostics, where making reliable decisions based on observed data is essential.

The decision-making process involves controlling error probabilities associated with incorrect decisions. Specifically, a Type-I error occurs if we reject H_0 when it is true, with probability denoted by α, and a Type-II error occurs if we fail to reject H_0 when H_1 is true, with probability denoted by β. Balancing these errors is central to classical testing; the significance level α is usually predefined, while the power of the test, $1-\beta$, represents the probability of correctly rejecting H_0 when H_1 is true.

A fundamental result in hypothesis testing is the Neyman-Pearson lemma, which provides a criterion for constructing the most powerful test for simple hypotheses at a given significance level α.

Lemma 1 (Neyman-Pearson Lemma[51])**.** *Let θ be a parameter determining the probability distribution of a random variable X, which has a probability density $p(x \mid \theta)$. Consider the simple null hypothesis $H_0 : \theta = \theta_0$ and the simple alternative hypothesis $H_1 : \theta = \theta_1$. Define the likelihood ratio as*

$$\Lambda(x) = \frac{p(x \mid \theta_1)}{p(x \mid \theta_0)}. \tag{15}$$

A test ϕ of the form

$$\phi(x) = \begin{cases} 1, & \text{if } \Lambda(x) > c \\ \gamma, & \text{if } \Lambda(x) = c \\ 0, & \text{if } \Lambda(x) < c \end{cases}, \tag{16}$$

exists such that it is the most powerful test at level α, maximizing the power $1-\beta$. Here, c is the smallest constant satisfying $\Pr\{\Lambda(X) > c \mid \theta = \theta_0\} \leq \alpha$, and $\gamma \in [0,1]$ is chosen to satisfy $\Pr\{\Lambda(X) > c \mid \theta = \theta_0\} + \gamma \Pr\{\Lambda(X) = c \mid \theta = \theta_0\} = \alpha$.

On the other hand, it is challenging to find a uniformly most powerful (UMP) test for all parameter values under composite hypotheses. However, under specific conditions such as a monotone likelihood ratio (MLR), it is possible to construct the UMP test. Here, the MLR is defined as a likelihood ratio such that the probability distribution differs for any $\theta_0 < \theta_1$ and the ratio $p(x|\theta_1)/p(x|\theta_0)$ is a non-decreasing function of a real-valued function $T(x)$. The Neyman-Pearson lemma 1 is extended to composite hypotheses with MLR as the following lemma.

Lemma 2 ([52])**.** *Let θ be a real parameter, and let X be a random variable with a probability density $p(x \mid \theta)$ with MLR in a statistic $T(x)$. Consider the following hypotheses:*

$$\begin{aligned} &\text{(i) } H_0 : \theta = \theta_{\text{th}}, \quad H_1 : \theta > \theta_{\text{th}} \\ &\text{(ii) } H_0 : \theta \leq \theta_{\text{th}}, \quad H_1 : \theta > \theta_{\text{th}}. \end{aligned}$$

In both cases (i) *and* (ii)*, there exists a UMP test ϕ of the form*

$$\phi(x) = \begin{cases} 1, & \text{if } T(x) > c \\ \gamma, & \text{if } T(x) = c \\ 0, & \text{if } T(x) < c \end{cases}, \tag{17}$$

at level α, maximizing the power $1-\beta$ for all θ. Here, c is the smallest constant satisfying $\Pr\{T(X) > c \mid \theta = \theta_{\text{th}}\} \leq \alpha$, and $\gamma \in [0,1]$ is chosen to satisfy $\Pr\{T(X) > c \mid \theta = \theta_{\text{th}}\} + \gamma \Pr\{T(X) = c \mid \theta = \theta_{\text{th}}\} = \alpha$.

C.2 Order parameter

Classical Neyman-Pearson test using the order parameter of the SPT phase In the context of Trivial vs. SPT, the classification between the trivial phase and SPT phase is achieved by testing whether the expectation value of the order parameter O_{SPT} is zero or not. To establish this, we first prove the following theorem (case (i) is applied in this subsection, while case (ii) is used in the next subsection) concerning the task of testing the expectation value of a Pauli string through projective measurements.

Theorem 1. *Let O be a general Pauli string represented by the spectral decomposition*

$$O = (+1)\Pi_+ + (-1)\Pi_-, \tag{18}$$

where $\Pi_\pm$ are projection operators onto the eigenspaces corresponding to eigenvalues ± 1. For a given quantum state ρ, consider the projective measurement described by O, with probabilities $p(\pm 1) = \mathrm{Tr}(\rho \Pi_\pm)$. Suppose this measurement is performed n times, and let x denote the number of times the eigenvalue $+1$ is observed. Define $\langle O \rangle = \mathrm{Tr}(\rho O) = p(+1) - p(-1)$ as the expectation value of the Pauli string O. Consider the following hypotheses:

$$
\begin{aligned}
&\text{(i) } H_0 : \langle O \rangle = \langle O \rangle_{\mathrm{th}}\,, \quad H_1 : \langle O \rangle > \langle O \rangle_{\mathrm{th}} \\
&\text{(ii) } H_0 : \langle O \rangle \leq \langle O \rangle_{\mathrm{th}}\,, \quad H_1 : \langle O \rangle > \langle O \rangle_{\mathrm{th}}\,.
\end{aligned}
$$

In both cases (i) *and* (ii)*, there exists a UMP test ϕ of the form*

$$
\phi(x) = \begin{cases} 1, & \text{if } x > c \\ \gamma, & \text{if } x = c\,, \\ 0, & \text{if } x < c \end{cases} \tag{19}
$$

at level α, maximizing the power $1 - \beta$ for all $\langle O \rangle$. Here, c is the smallest non-negative integer satisfying $\mathrm{Pr}(x > c \mid \langle O \rangle = \langle O \rangle_{\mathrm{th}}) \leq \alpha$, and $\gamma \in [0,1]$ is chosen to satisfy $\mathrm{Pr}(x > c \mid \langle O \rangle = \langle O \rangle_{\mathrm{th}}) + \gamma \mathrm{Pr}(x = c \mid \langle O \rangle = \langle O \rangle_{\mathrm{th}}) = \alpha$.

Proof. The random variable X, which is the number of times that the eigenvalue $+1$ is observed in n projective measurements described by O, follows a binomial distribution

$$
p(x \mid n, p_{\mathrm{bin}}) = \binom{n}{x} p_{\mathrm{bin}}^{x} (1 - p_{\mathrm{bin}})^{n-x}, \tag{20}
$$

where $p_{\mathrm{bin}} = p(+1) = \frac{1+\langle O \rangle}{2}$. In case (i), the hypotheses are rewritten as

$$
H_0 : p_{\mathrm{bin}} = p_0 = \frac{1 + \langle O \rangle_{\mathrm{th}}}{2}, \quad H_1 : p_{\mathrm{bin}} = p_1 > \frac{1 + \langle O \rangle_{\mathrm{th}}}{2}, \tag{21}
$$

and, in case (ii), the hypotheses are

$$
H_0 : p_{\mathrm{bin}} = p_0 \leq \frac{1 + \langle O \rangle_{\mathrm{th}}}{2}, \quad H_1 : p_{\mathrm{bin}} = p_1 > \frac{1 + \langle O \rangle_{\mathrm{th}}}{2}. \tag{22}
$$

The likelihood ratio for these hypotheses is

$$
\Lambda(x) = \frac{p(x \mid p_{\mathrm{bin}} = p_1)}{p(x \mid p_{\mathrm{bin}} = p_0)} = \frac{\binom{n}{x} p_1^x (1-p_1)^{n-x}}{\binom{n}{x} p_0^x (1-p_0)^{n-x}} = \left(\frac{p_1(1-p_0)}{p_0(1-p_1)} \right)^x \left(\frac{1-p_1}{1-p_0} \right)^n. \tag{23}
$$

Since $p_1 > p_0$, it follows that

$$
\frac{p_1(1-p_0)}{p_0(1-p_1)} > \frac{p_0(1-p_1)}{p_0(1-p_1)} = 1. \tag{24}
$$

Thus, $\Lambda(x)$ is a monotonically increasing function of the statistic $T(x) = x$, i.e., the MLR in $T(x) = x$. By Lemma 2, the test function defined in Eq. (19) is the UMP test for all $\langle O \rangle$ in both cases (i) and (ii). At a given significance level α, c is the smallest non-negative integer satisfying

$$
\mathrm{Pr}(x > c \mid \langle O \rangle = \langle O \rangle_{\mathrm{th}}) = \mathrm{Pr}\left(x > c \mid p_{\mathrm{bin}} = \frac{1 + \langle O \rangle_{\mathrm{th}}}{2} \right) = \sum_{x>c} \binom{n}{x} \left(\frac{1 + \langle O \rangle_{\mathrm{th}}}{2} \right)^x \left(\frac{1 - \langle O \rangle_{\mathrm{th}}}{2} \right)^{n-x} \leq \alpha, \tag{25}
$$

and $\gamma \in [0,1]$ is chosen to satisfy

$$
\begin{aligned}
&\mathrm{Pr}(x > c \mid \langle O \rangle = \langle O \rangle_{\mathrm{th}}) + \gamma \mathrm{Pr}(x = c \mid \langle O \rangle = \langle O \rangle_{\mathrm{th}}) \\
=&\mathrm{Pr}\left(x > c \mid p_{\mathrm{bin}} = \frac{1 + \langle O \rangle_{\mathrm{th}}}{2} \right) + \gamma \mathrm{Pr}\left(x = c \mid p_{\mathrm{bin}} = \frac{1 + \langle O \rangle_{\mathrm{th}}}{2} \right) \\
=&\sum_{x>c} \binom{n}{x} \left(\frac{1 + \langle O \rangle_{\mathrm{th}}}{2} \right)^x \left(\frac{1 - \langle O \rangle_{\mathrm{th}}}{2} \right)^{n-x} + \gamma \binom{n}{c} \left(\frac{1 + \langle O \rangle_{\mathrm{th}}}{2} \right)^c \left(\frac{1 - \langle O \rangle_{\mathrm{th}}}{2} \right)^{n-c} \\
=&\alpha.
\end{aligned} \tag{26}
$$

□

By performing the projective measurement described by the order parameter O_{SPT} n times, we test whether the expectation value $\langle O_{\mathrm{SPT}}\rangle$ is zero or not. The test data $\{\rho_{\mathrm{test}}^{(i)}\}_{i=1}^{N_{\mathrm{test}}}$ that we used all have expectation values of O_{SPT} greater than or equal to zero. Thus, the hypothesis corresponding to the trivial phase (labeled $y^{(i)}=0$) is $H_0 : \mathrm{Tr}(\rho_{\mathrm{test}}^{(i)} O_{\mathrm{SPT}}) = 0$, and the hypothesis corresponding to the SPT phase (labeled $y^{(i)}=1$) is $H_1 : \mathrm{Tr}(\rho_{\mathrm{test}}^{(i)} O_{\mathrm{SPT}}) > 0$. The test described in case (i) of Thm. 1 with $\langle O\rangle_{\mathrm{th}} = 0$ is therefore UMP for these hypotheses. We refer to this test as the classical Neyman-Pearson test for the SPT phase.

Here, the Type-I error probability is given by

$$\alpha_n^{(i)} = \sum_{x>c} \binom{n}{x} \left(\frac{1+\langle O\rangle^{(i)}}{2}\right)^x \left(\frac{1-\langle O\rangle^{(i)}}{2}\right)^{n-x} + \gamma \binom{n}{c} \left(\frac{1+\langle O\rangle^{(i)}}{2}\right)^c \left(\frac{1-\langle O\rangle^{(i)}}{2}\right)^{n-c}, \tag{27}$$

where $\langle O\rangle^{(i)} = \mathrm{Tr}(\rho_{\mathrm{test}}^{(i)} O_{\mathrm{SPT}})$ for test data labeled $y^{(i)}=0$. The Type-II error probability is given by

$$\beta_n^{(i)} = \sum_{x<c} \binom{n}{x} \left(\frac{1+\langle O\rangle^{(i)}}{2}\right)^x \left(\frac{1-\langle O\rangle^{(i)}}{2}\right)^{n-x} + (1-\gamma) \binom{n}{c} \left(\frac{1+\langle O\rangle^{(i)}}{2}\right)^c \left(\frac{1-\langle O\rangle^{(i)}}{2}\right)^{n-c}, \tag{28}$$

where $\langle O\rangle^{(i)} = \mathrm{Tr}(\rho_{\mathrm{test}}^{(i)} O_{\mathrm{SPT}})$ for test data labeled $y^{(i)}=1$. The error probabilities α_n and β_n computed in the main text are the averages of $\alpha_n^{(i)}$ and $\beta_n^{(i)}$ over the test data for each phase, respectively.

Bayesian test using the order parameter of the FM phase In the context of Trivial vs. FM, the classification between the trivial phase and the FM phase is performed by testing whether the expectation value of the order parameter O_{FM}, defined as a linear combination of local observables, is zero or not. While the optimal classical post-processing for the SPT order parameter, QCNN, and Exact QCNN after measurements is determined based on Thm. 1, the same does not hold for O_{FM}. This is because, unlike other cases where the measurement results follow a Bernoulli distribution, the measurement outcomes associated with O_{FM} follow a multinomial distribution. For multinomial distributions, there is no general result providing a UMP test analogous to Lemma 2. Therefore, since knowing the order parameter implies possessing prior knowledge about quantum phases, we employ a Bayesian test to make effective use of this knowledge.

The eigenvalue decomposition of the order parameter O_{FM} is given by

$$O_{\mathrm{FM}} = \frac{1}{L}\sum_{i=1}^{L} Z_i = \sum_{m=0}^{L} \lambda_m \Pi_m, \tag{29}$$

where $\lambda_m = 1 - 2\frac{m}{L}$ are the eigenvalues, and $\Pi_m = \sum_{|\lambda|=m} |\lambda\rangle\langle\lambda|$ represents the projection onto the subspace spanned by computational basis states with Hamming weight $|\lambda| = m$. Performing the projective measurement described by O_{FM} n times, we denote X_m as the number of times Π_m is observed and define the random variable $\mathbf{X} = (X_0, ..., X_L)$. To formalize this setting, we introduce the multinomial and Dirichlet distributions. The multinomial distribution $\mathrm{Mul}(n, \mathbf{p})$ describes the probability of obtaining category counts $\mathbf{X} = (X_0, ..., X_L)$ given event probabilities $\mathbf{p} = (p_0, ..., p_L)$, where $p_m \geq 0$ and $\sum_{m=0}^{L} p_m = 1$. The probability mass function is

$$p_{\mathrm{Mul}}(\mathbf{x} \mid n, \mathbf{p}) = \frac{n!}{x_0! \dots x_L!} \prod_{m=0}^{L} p_m^{x_m}, \tag{30}$$

where $\sum_{m=0}^{L} x_m = n$. The Dirichlet distribution $\mathrm{Dir}(\alpha)$ is defined over the L-dimensional simplex $\mathbf{P} = (P_0, ..., P_L)$, where $P_m \geq 0$ and $\sum_{m=0}^{L} P_m = 1$. It is parameterized by pseudo-counts $\alpha = (\alpha_0, ..., \alpha_L)$, with the probability density function

$$p_{\mathrm{Dir}}(\mathbf{p} \mid \alpha) = \frac{1}{B(\alpha)} \prod_{m=0}^{L} p_m^{\alpha_m - 1}, \tag{31}$$

where $B(\alpha)$ is the multivariate Beta function given by

$$B(\alpha) = \frac{\prod_{m=0}^{L} \Gamma(\alpha_m)}{\Gamma(\sum_{m=0}^{L} \alpha_m)}, \tag{32}$$

with $\Gamma(\cdot)$ denoting the Gamma function.

The hypothesis corresponding to the trivial phase (labeled $y^{(i)} = 0$) is $H_0 : \mathrm{Tr}(\rho^{(i)}_{\mathrm{test}} O_{\mathrm{FM}}) = 0$, and the true distribution of $\mathbf{X}$ follows $\mathrm{Mul}(n, \mathbf{p} = (\mathrm{Tr}(\rho^{(i)}_{\mathrm{test}}\Pi_0), ..., \mathrm{Tr}(\rho^{(i)}_{\mathrm{test}}\Pi_L)), H_0) = \mathrm{Mul}^{(i)}_0$. As a prior for the multinomial probabilities $\mathbf{P}$, we choose a Dirichlet distribution $\mathrm{Dir}(\alpha = \alpha_0, H_0)$, where

$$\alpha_0 = \frac{1}{2^L}\left(\binom{L}{0}, ..., \binom{L}{L}\right). \tag{33}$$

This choice is motivated by the fact that the trivial phase in the one-dimensional cluster-Ising model is dominated by $\sum_{i=1}^{L} X_i$, whose ground states are uniform superpositions of computational basis states. Since each Π_m represents a sum of projections onto $\binom{L}{m}$ individual computational basis states, we choose the Dirichlet prior so that α_m is proportional to $\binom{L}{m}$. The marginal likelihood under H_0 is then computed as

$$\begin{aligned} p(\mathbf{x} \mid n, H_0) &= \int p_{\mathrm{Mul}}(\mathbf{x} \mid n, \mathbf{p}) p_{\mathrm{Dir}}(\mathbf{p} \mid \alpha = \alpha_0, H_0) d\mathbf{p} \\ &= \int \frac{n!}{x_0! \dots x_L!} \prod_{m=0}^{L} p_m^{x_m} \frac{1}{B(\alpha_0)} \prod_{m=0}^{L} p_m^{\alpha_m - 1} d\mathbf{p} \\ &= \frac{n!}{x_0! \dots x_L!} \frac{1}{B(\alpha_0)} \int \prod_{m=0}^{L} p_m^{\alpha_m + x_m - 1} d\mathbf{p} \\ &= \frac{n!}{x_0! \dots x_L!} \frac{B(\alpha_0 + \mathbf{x})}{B(\alpha_0)}. \end{aligned} \tag{34}$$

On the other hand, the hypothesis corresponding to the FM phase (labeled $y^{(i)} = 1$) is $H_1 : \mathrm{Tr}(\rho^{(i)}_{\mathrm{test}} O_{\mathrm{FM}}) \neq 0$, and the true distribution of $\mathbf{X}$ follows $\mathrm{Mul}(n, \mathbf{p} = (\mathrm{Tr}(\rho^{(i)}_{\mathrm{test}}\Pi_0), ..., \mathrm{Tr}(\rho^{(i)}_{\mathrm{test}}\Pi_L)), H_1) = \mathrm{Mul}^{(i)}_1$. As a prior, we choose $\mathrm{Dir}(\alpha = \alpha_1, H_1)$, where

$$\alpha_1 = \frac{2}{(L+1)(L+2)}(1, ..., L+1) \quad \text{or} \quad \frac{2}{(L+1)(L+2)}(L+1, ..., 1). \tag{35}$$

This choice is motivated by the fact that the FM phase in the one-dimensional cluster-Ising model is dominated by $-\sum_{i=1}^{L} Z_i Z_{i+1}$, whose ground states are superpositions of $|0\rangle^{\otimes L}$ and $|1\rangle^{\otimes L}$. The observed Π_m values are thus more likely to be biased toward either small or large m. The marginal likelihood under H_1 is similarly given by

$$p(\mathbf{x} \mid n, H_1) = \frac{n!}{x_0! \dots x_L!} \frac{B(\alpha_1 + \mathbf{x})}{B(\alpha_1)}. \tag{36}$$

In Bayesian hypothesis testing, the Bayes factor is defined as the ratio of marginal likelihoods,

$$\mathrm{BF}_{10} = \frac{p(\mathbf{x} \mid n, H_1)}{p(\mathbf{x} \mid n, H_0)} = \frac{B(\alpha_1 + \mathbf{x}) B(\alpha_0)}{B(\alpha_0 + \mathbf{x}) B(\alpha_1)}, \tag{37}$$

and the test is performed by comparing BF_{10} to a threshold c. Since the prior distribution under H_1 is not uniquely determined, we adopt the Bayes factor corresponding to the distribution that deviates most from the threshold. The Type-I and Type-II error probabilities are then given by

$$\alpha_n^{(i)} = \Pr\Big(\mathrm{BF}_{10} \geq c \mid \mathbf{X} \sim \mathrm{Mul}^{(i)}_0, H_0\Big), \quad \beta_n^{(i)} = \Pr\Big(\mathrm{BF}_{10} < c \mid \mathbf{X} \sim \mathrm{Mul}^{(i)}_1, H_1\Big). \tag{38}$$

The error probabilities α_n and β_n computed in the main text are the averages of $\alpha_n^{(i)}$ and $\beta_n^{(i)}$ over the test data for each phase, respectively.

C.3 QCNN and Exact QCNN

Exact QCNN for the FM phase Exact QCNNs are constructed based on the Multiscale Entanglement Renormalization Ansatz (MERA)[54] and the Multiscale String Operator (MSO)[11,20]. The MSO is defined as a sum of products of exponentially many string order parameters that act on different locations. For example, for the SPT phase, the MSO is expressed as

$$\mathrm{MSO} = \sum_{ab} C^{(1)}_{ab} \mathscr{S}_{ab} + \sum_{a_1 b_1 a_2 b_2} C^{(2)}_{a_1 b_1 a_2 b_2} \mathscr{S}_{a_1 b_1} \mathscr{S}_{a_2 b_2} + \cdots, \tag{39}$$

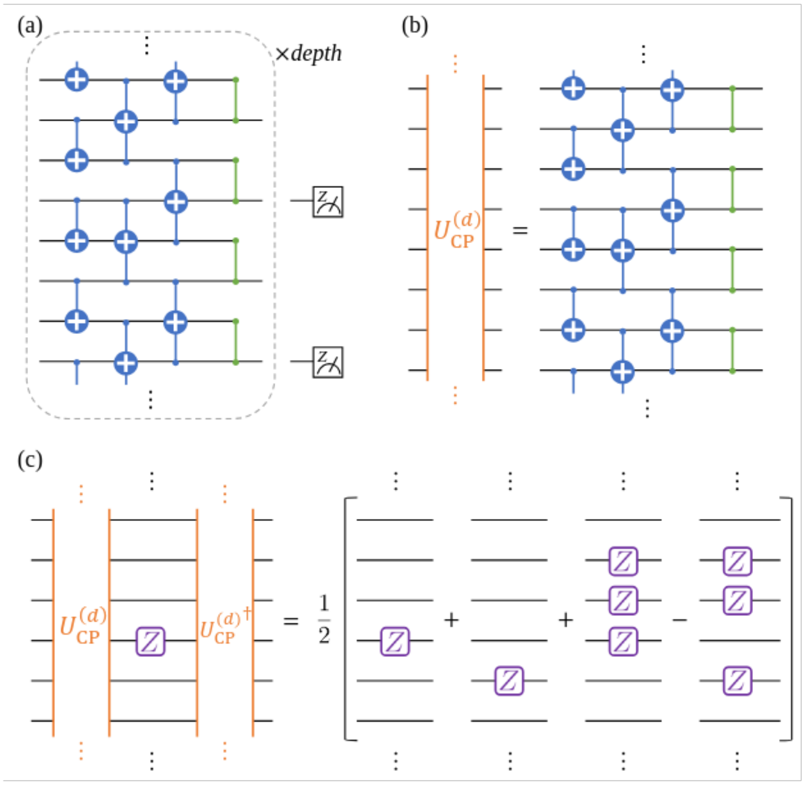

Figure 12. Exact QCNN for the FM phase. Panel (a) shows the full Exact QCNN circuit, identical to Fig. 7(a), where the convolution and pooling layers at a certain depth d are denoted as $U_{\mathrm{CP}}^{(d)}$, as shown in panel (b). Panel (c) illustrates the equality $U_{\mathrm{CP}}^{(d)\dagger} Z_i U_{\mathrm{CP}}^{(d)} = \frac{1}{2}(Z_i + Z_{i+1} + Z_{i-2}Z_{i-1}Z_i - Z_{i-2}Z_{i-1}Z_{i+1})$ obtained using Pauli propagation. The blue two-qubit gates represent controlled–X gates, the blue three-qubit gates represent Toffoli gates, the green two-qubit gates represent controlled–Z gates, and the ZZ measurement indicates a projective measurement in the computational basis.

where the string order parameter O_{SPT} in the main text is redefined as $\mathscr{S}_{ab} = Z_a X_{a+1} X_{a+3} \dots X_{b-3} X_{b-1} Z_b$. The observable in the Heisenberg picture for the Exact QCNN proposed in Ref.[11] (Fig. 7(b)) matches this expression.

We consider an Exact QCNN for the FM phase characterized by the order parameter that is the linear combination of local observables, as O_{FM} in the main text. Since this order parameter is not a string, we instead consider the long-range order parameter $\mathscr{S}_{ab} = Z_a Z_b$ and construct a circuit such that the observable in the Heisenberg picture matches Eq. (39). The Exact QCNN circuit for the FM phase is shown in Fig. 12(a). The unitary operation combining the convolution and pooling layers at a certain depth d, $U_{\mathrm{CP}}^{(d)}$, is defined as shown in Fig. 12(b). Using the relations for Pauli propagation illustrated in Fig. 12(c), the observable in the Heisenberg picture for this Exact QCNN becomes

$$\begin{aligned} &(U_{\mathrm{CP}}^{(depth)} \cdots U_{\mathrm{CP}}^{(1)})^\dagger (I \otimes \cdots \otimes I \otimes Z \otimes I \otimes \cdots \otimes I \otimes Z)(U_{\mathrm{CP}}^{(depth)} \cdots U_{\mathrm{CP}}^{(1)}) \\ =&\sum_{ab} C_{ab}^{(1)} Z_a Z_b + \sum_{a_1 b_1 a_2 b_2} C_{a_1 b_1 a_2 b_2}^{(2)} Z_{a_1} Z_{b_1} Z_{a_2} Z_{b_2} + \cdots \\ =&\sum_{ab} C_{ab}^{(1)} \mathscr{S}_{ab} + \sum_{a_1 b_1 a_2 b_2} C_{a_1 b_1 a_2 b_2}^{(2)} \mathscr{S}_{a_1 b_1} \mathscr{S}_{a_2 b_2} + \cdots, \end{aligned} \tag{40}$$

which matches the MSO in Eq. (39).

In the main text (the subsection on error probabilities), it is evident that under a few copies of test data, the Exact QCNN for the FM phase performs worse than the order parameter. This suggests that Exact QCNNs may not be advantageous for quantum phases characterized by order parameters expressible as linear combinations of local observables.

Classical Neyman-Pearson test using QCNNs and Exact QCNNs The QCNN and the Exact QCNN classify quantum phases based on the expectation values of Pauli operators or Pauli strings with respect to the output quantum states (e.g., ZXZ Pauli string: the Exact QCNN for the Trivial vs. SPT case). By performing the projective measurement described by a Pauli operator or Pauli string n times, we test whether the output expectation value $f(\rho_{\text{test}}^{(i)})$ for input test data $\rho_{\text{test}}^{(i)}$ exceeds 0.5 or not. The hypothesis corresponding to the trivial phase (labeled $y^{(i)} = 0$) is $H_0 : f(\rho_{\text{test}}^{(i)}) \leq 0.5$, and the hypothesis corresponding to the non-trivial phase (labeled $y^{(i)} = 1$) is $H_1 : f(\rho_{\text{test}}^{(i)}) > 0.5$. The test described in case (ii) of Thm. 1 with $\langle O \rangle_{\text{th}} = 0.5$ is therefore UMP for these hypotheses. We refer to this test as the classical Neyman-Pearson test for QCNNs and Exact QCNNs.

The Type-I error probability for test data labeled $y^{(i)} = 0$ is given by

$$\alpha_n^{(i)} = \sum_{x>c} \binom{n}{x} \left(\frac{1+f(\rho_{\text{test}}^{(i)})}{2} \right)^x \left(\frac{1-f(\rho_{\text{test}}^{(i)})}{2} \right)^{n-x} + \gamma \binom{n}{c} \left(\frac{1+f(\rho_{\text{test}}^{(i)})}{2} \right)^c \left(\frac{1-f(\rho_{\text{test}}^{(i)})}{2} \right)^{n-c}, \tag{41}$$

and the Type-II error probability for test data labeled $y^{(i)} = 1$ is given by

$$\beta_n^{(i)} = \sum_{x<c} \binom{n}{x} \left(\frac{1+f(\rho_{\text{test}}^{(i)})}{2} \right)^x \left(\frac{1-f(\rho_{\text{test}}^{(i)})}{2} \right)^{n-x} + (1-\gamma) \binom{n}{c} \left(\frac{1+f(\rho_{\text{test}}^{(i)})}{2} \right)^c \left(\frac{1-f(\rho_{\text{test}}^{(i)})}{2} \right)^{n-c}. \tag{42}$$

The error probabilities α_n and β_n computed in the numerical results are the averages of $\alpha_n^{(i)}$ and $\beta_n^{(i)}$ over the test data for each phase, respectively.