**PAPER**

# Learning quantum symmetries with interactive quantum-classical variational algorithms

To cite this article: Jonathan Z Lu *et al* 2024 *J. Phys. A: Math. Theor.* **57** 315304

View the article online for updates and enhancements.

# Learning quantum symmetries with interactive quantum-classical variational algorithms

**Jonathan Z Lu**[1,5,*] ⓘ**, Rodrigo Araiza Bravo**[1,3] ⓘ**,
Kaiying Hou**[1,3]**, Gebremedhin A Dagnew**[1,4]**,
Susanne F Yelin**[1,*] **and Khadijeh Najafi**[2,*]

[1] Department of Physics, Harvard University, Cambridge, MA, United States of America
[2] IBM Quantum, IBM T.J. Watson Research Center, Yorktown Heights, NY 10598, United States of America

E-mail: lujz@mit.edu, syelin@g.harvard.edu and knajafi@ibm.com

## Abstract

A symmetry of a state $|\psi\rangle$ is a unitary operator of which $|\psi\rangle$ is an eigenvector. When $|\psi\rangle$ is an unknown state supplied by a black-box oracle, the state's symmetries provide key physical insight into the quantum system; symmetries also boost many crucial quantum learning techniques. In this paper, we develop a variational hybrid quantum–classical learning scheme to systematically probe for symmetries of $|\psi\rangle$ with no *a priori* assumptions about the state. This procedure can be used to learn various symmetries at the same time. In order to avoid re-learning already known symmetries, we introduce an interactive protocol with a classical deep neural net. The classical net thereby regularizes against repetitive findings and allows our algorithm to terminate empirically with all possible symmetries found. An iteration of the learning algorithm can be implemented efficiently with non-local SWAP gates; we also give a less efficient algorithm with only local operations, which may be more appropriate for current noisy quantum devices. We simulate our algorithm on representative families of states, including cluster states and ground states of Rydberg and

---

[3] These authors contributed equally to this work.
[4] 1QB Information Technologies, Inc.
[5] Presently at Department of Mathematics, MIT
[*] Authors to whom any correspondence should be addressed.

Ising Hamiltonians. We also find that the numerical query complexity scales well for up to moderate system sizes.

Keywords: quantum machine learning, variational quantum algorithms, interactive learning, quantum symmetries
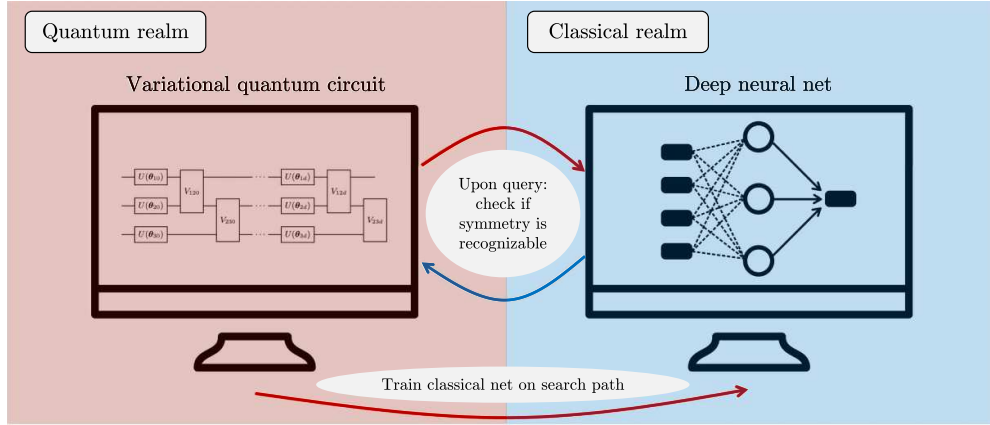
## 1. Introduction

Symmetries are of ubiquitous importance across physics, relating intimately to conservation laws and guiding the formulation of physical theories. The search for symmetries in a given physical system is a central problem in the field, and until recently, has remained a purely analytical and model-driven task. However, developments in machine learning have now opened the door for data-driven approaches. Classically, recent studies have used learning algorithms to discover conserved quantities by using the system's equation of motion [1–4] or by clustering methods [5]. In the quantum setting, classical neural networks have been successfully trained to classify states of matter [6–8], including symmetry-protected topological ones [9–11]. Quantum algorithms binarily testing for symmetries of certain Hamiltonians and for specific quantum symmetries (such as bosonic exchange) have too been constructed [12, 13]. Knowledge of a given state's symmetries significantly boosts the power of many fundamental quantum learning schemes [14–16]. Along these lines, it has been shown that knowledge of symmetries can bootstrap the quantum approximate optimization algorithm (QAOA) [17]. Symmetry-abiding neural networks can also be used as a variational ansatz for states of lattice models [18, 19].

Beyond their intrinsic importance, quantum symmetries also encode key information about the underlying quantum state. Specifically, when a state $|\psi\rangle$ is unknown, efficient extraction of information about $|\psi\rangle$ from limited data is of great interest across both physics and theoretical computer science. Full information extraction, however, is exponentially costly with the system size, requiring complete state tomography [20]. While novel approaches circumventing this issue have appeared—including basis-enhanced Born machines, classical machine learning, and shadow tomography—comparatively little attention has been paid to the direct and systematic learning of quantum symmetries, which also serve to characterize a state [8, 21–28]. In this article, we take a first step in that direction.

Define a quantum symmetry of $|\psi\rangle$ to be a unitary $U$ such that

$$U|\psi\rangle = e^{i\theta}|\psi\rangle; \tag{1}$$

that is, an operator $U$ for which $|\psi\rangle$ is an eigenvector. In line with experimental conditions, we shall assume that a black-box oracle machine prepares copies of an unknown state $|\psi\rangle$; our goal is to efficiently express the set of symmetries of $|\psi\rangle$. To find the symmetries of $|\psi\rangle$ [29], we propose a hybrid quantum–classical algorithm consisting of a variational quantum circuit interacting with a classical neural network. We assume no prior information about $|\psi\rangle$, so that our algorithm is agnostic to the input state whose symmetries it learns. Note that if $|\psi\rangle$ is known then the symmetry learning problem is in a sense trivial; see appendix A. The quantum circuit generates symmetries of $|\psi\rangle$ while simultaneously training the classical net to 'remember' the history of symmetries already found. In turn, the classical net alerts the quantum circuit when it parameterizes an operator close to a previously-found symmetry, allowing the quantum circuit to redirect its search and thereby avoid repeatedly generating similar symmetries. By both generating symmetries and keeping track of known symmetries, our algorithm continuously

**Figure 1.** Schematic overview of symmetry learning scheme. A variational quantum circuit minimizes a loss function to generate symmetries, at the same time training a classical neural network to recognize the path of potential symmetry operators it searches. The classical net then alerts the quantum circuit when it is searching for symmetries similar to those already found, so that the quantum circuit can redirect its search accordingly.

finds new symmetries. Thus, we observe empirically that the learning procedure eventually terminates. The overall structure of quantum–classical interaction is shown in figure 1.

### 1.1. Overview

Before providing a complete exposition of our contributions, we briefly discuss the main ideas permeating our results. Since our goal is to learn symmetries $U$ of $|\psi\rangle$, we require an explicit technique to model operators $U$. To do so, we encode operators into a quantum circuit ansatz $\mathcal{C}$. Moreover, since we aim to iteratively learn $U$, the circuit must be parameterized by a collection of variables $\boldsymbol{\theta}$ which, as they vary, also vary the operator that $\mathcal{C}$ represents. We can then apply optimization techniques from classical machine learning, e.g. gradient descent and variants, to vary $\boldsymbol{\theta}$ until $\mathcal{C}$ closely represents $U$. That is, we train the variational circuit by direct application of classical optimization algorithms. Our metric of 'closeness' is given by a loss function which is also used to train the circuit. An important point in this paper, however, is that the exact choice of the loss function, or equivalently the technique we use to *verify* a proposed symmetry, has subtleties. In particular, there is a tradeoff between the locality of the loss function (i.e. the furthest distance between qubits which are required by the loss function to interact) and the efficiency for which the loss function may be evaluated. Therefore, depending on the capabilities of the underlying quantum device, different choices may be more appropriate.

Another key idea in this paper is motivated by the idea that finding non-trivial symmetries necessitates finding many symmetries. As a starting example, since the identity operator is always a symmetry, there is nothing stopping the circuit $\mathcal{C}$ from converging to the identity every time. More broadly, since there are many symmetries (continuous families, in fact), it is desirable to repeatedly run the learning algorithm to find symmetries, while avoiding symmetries that are similar to those that have already been found. This goal is precisely the motivation for the introduction of a classical neural network that learns alongside the quantum circuit. Unlike

the quantum circuit, which models an underlying unitary that receives $|\psi\rangle$ as input, the classical net receives $\boldsymbol{\theta}$ as input and models a decision function that essentially outputs a single bit $b$. We construct the classical net such that $b$ serves as a flag for similarity to known symmetries, encoded by $\boldsymbol{\theta}$. That is, when $b = 1$, the classical net predicts that the current symmetry search path is similar to one already known. Hence, we may adjust the quantum learning algorithm appropriately to avoid repetitive symmetry finding. Since the $\boldsymbol{\theta}$ have a physically meaningful notion of locality, we utilize convolutional nets in the construction of the classical net.

We remark that our algorithm bears a particular resemblance to the variational quantum eigensolver (VQE) [30]. In the setting of VQE, one is given a Hamiltonian $H$ and tasked with finding its ground state energy, i.e. $E_* = \min_{|\psi\rangle}\langle\psi|H|\psi\rangle$. To do so, a VQE employs a variational quantum circuit and trains it to map $|0^n\rangle$ to a state $|\varphi\rangle$ such that $E = \langle\varphi|H|\varphi\rangle$ is close to $E_*$. Measuring the energy of the output state will then estimate the ground state energy. Learning symmetries can be seen as one way to approach the converse problem: we are instead given a state $|\psi\rangle$ and asked to find operators $U$ that are symmetries of $|\psi\rangle$. They share the trait of both being heuristic learning algorithms utilizing variational quantum circuits. However, learning symmetries appears at face value much more difficult. Rather than finding a minimum energy, we find operators themselves. Moreover, we search not for a single energy, but for a large family of operators which are all symmetries of $|\psi\rangle$.

In section 2, we discuss the full details of the symmetry learning scheme and place it within the context of noisy near-term quantum hardware. In section 3, we demonstrate the algorithm on three representative families of quantum states. We also benchmark the scalability of the algorithm and its robustness to noise. Finally, in section 4 we summarize our findings and discuss some applications of symmetry learning.
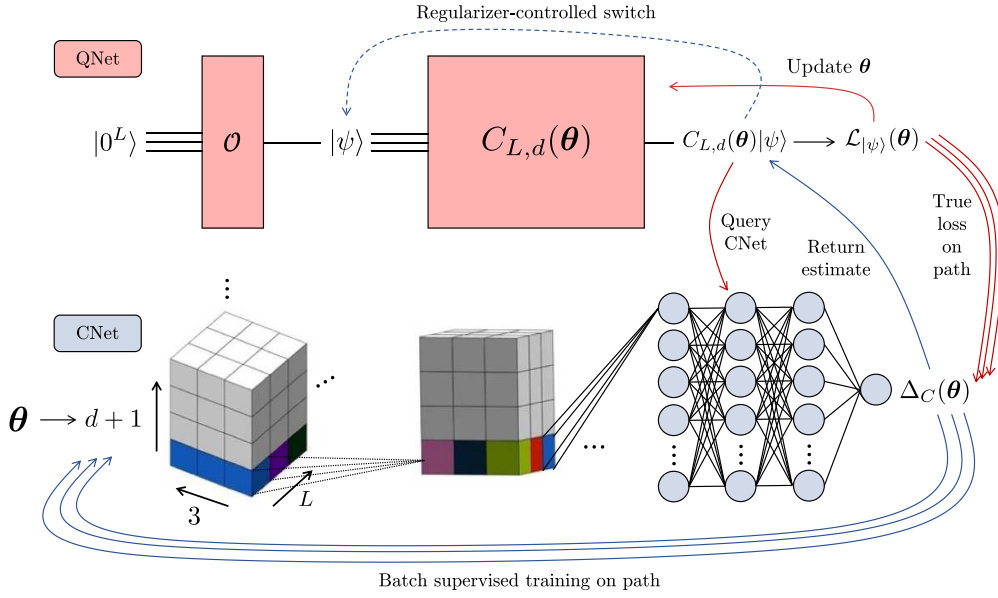
## 2. Hybrid learning scheme

By analogy to an experimental setting, we assume access to copies of an unknown state $|\psi\rangle$ and the task is to discover the set of its symmetries. Since the closeness of an operator to a symmetry can be quantified in a manner similar to metrics of state overlap, symmetry learning has a natural interpretation as an optimization problem, given as

$$S[|\psi\rangle] = \left\{ V \in \mathrm{U}\left(2^L\right) \ : \ |\langle\psi|V|\psi\rangle|^2 = 1 \right\}, \tag{2}$$

where $\mathrm{U}(2^L)$ is the unitary matrix Lie group of dimension $2^L$ (note that any mention of $\mathrm{U}(2^L)$ in this paper can be readily replaced with $\mathrm{SU}(2^L)$ the special unitary group, so long as the circuit modeling it has only gates in $\mathrm{SU}(2^L)$). Symmetries can be filtered into a collection of sets by fixing a universal parameterized quantum circuit (PQC) family $C_{L,D}(\boldsymbol{\theta})$ of depth $D$ on $L$ qubits and parameters $\boldsymbol{\theta}$. Define $S_D[|\psi\rangle]$ as the collection of symmetries of $|\psi\rangle$ representable by $C_{L,D}(\boldsymbol{\theta})$. By universality, every symmetry is contained in $S_{O(\exp L)}[|\psi\rangle]$ and

$$S_0[|\psi\rangle] \subseteq S_1[|\psi\rangle] \subseteq \cdots S_{O(\exp L)}[|\psi\rangle]. \tag{3}$$

We define the variational symmetry learning problem as the discovery and classification of symmetries of $|\psi\rangle$ for a fixed and constant depth $D$. In practice, this may neglect symmetries that require exponentially long circuit depths to be implemented in the chosen PQC architecture. A variational approach can therefore not find all symmetries in the worst case, but its utility remains in general to find symmetries that do not require impractically large circuits and time to learn.
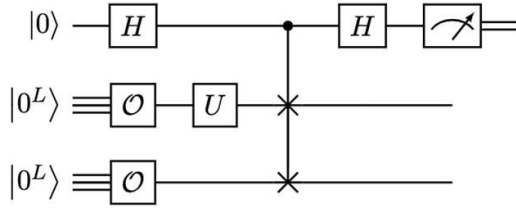
**Figure 2.** Depiction of the symmetry learning algorithm. The quantum net (red) involves a variational quantum circuit $C_{L,d}(\boldsymbol{\theta})$ on $L$ qubits and with block-depth $d$ measured by a loss function $\mathcal{L}_{|\psi\rangle}(\boldsymbol{\theta})$. $\boldsymbol{\theta}$ is varied until $C_{L,d}(\boldsymbol{\theta})$ represents a symmetry. At the end of each epoch, a classical 3-dimensional convolutional deep net (CNet; blue) learns the loss function along the path just explored by the QNet. During future epochs, the CNet informs the QNet as to whether it has already explored its current path, and hence whether it needs to leave its current path to an as-yet explored region. The CNet structure matches the parameter structure of the QNet. The first layer convolves over each set of 3 parameters per qubit, and the second layer convolves over every parameter; the result is inputted into a 3-layer fully connected network.
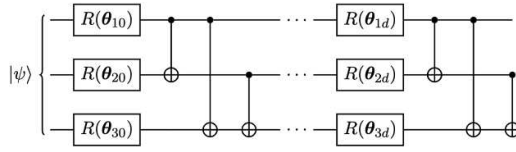
We develop the symmetry learning scheme in three stages. First, we devise a method to verify whether a given operator $U$ is a symmetry. We then show that the verification procedure can be upgraded into a learning procedure, by introducing a variational quantum algorithm (VQA) built upon a PQC family $C_{L,D}$ of depth $D$ acting on $L$ qubits. Finally, we boost the learning scheme by introducing a regularization technique based on classical deep learning that prevents the VQA from repetitively proposing similar symmetries. The full scheme is illustrated in figure 2, which we refer to throughout this section.

## 2.1. Symmetry verification

To begin, suppose that we are given an operator $U$ represented by a PQC family $C_{L,D}$. We restrict ourselves only to symmetries expressible in such a decomposition, noting that a universal quantum gate family can generally express any operator up to small error. Our first step will be to verify that $U$ is indeed a symmetry of $|\psi\rangle$. Such a problem can be decided in polynomial time by a method adapted from quantum fingerprinting, known as the SWAP$_\epsilon$ test [31]. The heart of the procedure is the controlled-SWAP gate, which takes $|0\rangle \leftrightarrow |1\rangle$ unitarily if a

**Figure 3.** Verification circuit $V$ implementing a SWAP$_\epsilon$ test. The central operation is a swap of the two $L$-qubit registers (bottom registers), controlled by the ancillary qubit (top register). By running $V$ $O(1/\epsilon^2)$ times, we can determine the overlap $|\langle\psi|U|\psi\rangle|^2$ up to error $\epsilon$. We estimate $U$ to be a symmetry of $|\psi\rangle$ if the overlap is at least $1 - O(\epsilon)$.



**Figure 4.** Example of a universal parameterized quantum circuit on $L = 3$ qubits. The cross-hairs are controlled-NOT gates and the $R$ gates are single-qubit rotations, parameterized by $\boldsymbol{\theta}$. There are $d$ layers of $\binom{L}{2}$ controlled-NOT gates between each pair of qubits, sandwiched by the rotation gates.

control qubit is $|1\rangle$. By utilization of the circuit in figure 3, measurement of the ancillary qubit yields $|0\rangle$ with probability

$$\Pr[0] = \frac{1}{2} + \frac{1}{2}|\langle\psi|U|\psi\rangle|^2. \tag{4}$$

Thus the overlap $|\langle\psi|U|\psi\rangle|^2$ is a function of the bias (probability distance from $1/2$) of a coin flip, which can be determined to error $\epsilon$ in $O(1/\epsilon^2)$ trials [31]. Since the number of swaps scales linearly with qubit size, the time complexity of the SWAP$_\epsilon$ test is polynomial: $O(L/\epsilon^2)$.

Although the SWAP$_\epsilon$ is efficient, it uses long-range SWAP gates spanning $O(L)$ qubits, which are not practical for general near-term quantum devices. In section 2.5, we give an alternative verification procedure free of long-range couplings at the expense of computational efficiency.

### 2.2. Variational quantum generative algorithm

In the process of symmetry verification we utilized the matrix element $|\langle\psi|U|\psi\rangle|^2$ as a measure of the closeness $U$ is to a symmetry. The overlap may be interpreted as a loss metric, i.e.

$$\mathcal{L}_{|\psi\rangle}(U) = \left(1 - |\langle\psi|U|\psi\rangle|^2\right)^2. \tag{5}$$

This overlap can be estimated statistically by the SWAP$_\epsilon$ test. The formulation of verification as a loss allows for a transformation from verification—a discriminative procedure—to proposition (search)—a generative model. Consider again a universal PQC; one such family includes layers of $\binom{L}{2}$ CNOT gates that are sandwiched in between single-qubit rotation gates, as depicted in figure 4 for $L = 3$ qubits. This PQC ansatz is well-established and is known to

be universal as the depth reaches exponential values [32]. In place of circuit depth, we use the *block-depth* $d \geq 0$, the number of layers of $\binom{L}{2}$ CNOT gates. Hence, $d = 0$ corresponds to a single layer of rotations, $d = 1$ gives 2 layers of rotations sandwiching 1 layer of CNOT gates, etc. The circuit on $L$ qubits is specified uniquely by either the circuit depth $D$ or the block-depth $d$; we shall use the latter, writing the circuit as $C_{L,d}$ and filtering the symmetries by block-depth as

$$S_0 \left[ |\psi\rangle \right] \subseteq \cdots \subseteq S_d \left[ |\psi\rangle \right] \subseteq \cdots S_{O(\exp L)} \left[ |\psi\rangle \right]. \tag{6}$$

Under a parameterization $\boldsymbol{\theta}$, define for convenience

$$\mathcal{L}_{|\psi\rangle} (\boldsymbol{\theta}) \equiv \mathcal{L}_{|\psi\rangle} \left( C_{L,d} (\boldsymbol{\theta}) \right). \tag{7}$$

Our VQA generates symmetries as follows. Initialize the PQC randomly by drawing $\boldsymbol{\theta}_0 \sim \text{Unif}(0, 2\pi)^{3L(d+1)}$. Next, run $C_{L,d}$ on $|\psi\rangle$ and compute $\mathcal{L}_{|\psi\rangle}(\boldsymbol{\theta}_0)$. The loss will likely be large, but $\boldsymbol{\theta}$ may be improved by an updating algorithm of classical machine learning. Our algorithm employs Nelder–Mead simplex search, which optimizes $\boldsymbol{\theta}$ locally by updating the parameter to its simplex neighbor with the minimum loss value [33]. The use of a classical updating scheme on a PQC has been adopted to considerable success in many hybrid quantum–classical algorithms such as variational quantum eigensolver (VQE) and QAOA. Here, $\boldsymbol{\theta}$ is iteratively updated until $\mathcal{L}_{|\psi\rangle}(\boldsymbol{\theta})$ falls below an error threshold $\delta$, at which point the symmetry is proposed. See appendix C for details on our choices of hyperparameters.

Since the PQC consists of layers of parameterized gates updated by a training algorithm, it has a natural interpretation as a quantum analog of a neural network, which we refer to as the QNet. The QNet VQA comprises the top (red) half of figure 2.

We remark that the choice of loss function here is global and generally may suffer from the barren plateau (BP) problem that affects VQAs nearly universally. Mitigating BPs in our algorithm is out of the scope of this work, but we expect that based on current advances in BP mitigation, it will be possible to mitigate BPs in our algorithm by modification of some components (e.g. choice of loss or ansatz) without affecting the essence of the protocol [34]. Furthermore, we will show numerically in this paper that for moderate system sizes the scaling appears to be fairly reasonable. Alternatively, it has been shown that it may be possible to avoid BPs simply by restricting the class of searched symmetries to a sufficiently local subset [35].

### 2.3. Symmetries as manifolds

The VQA proposed in the previous section above successfully generates symmetries; however, with no further assistance the PQC may propose the same symmetry repetitively. In the extreme case, the PQC could simply choose to always output the identity matrix. The prevention of repeat findings motivates the introduction of a regularization procedure. A naive method might enumerate the set of symmetries found up to the $k$th epoch of search, $\boldsymbol{\theta}_1, ..., \boldsymbol{\theta}_k$, and then adding a regularization term to the loss function of the form

$$\mathcal{L}_{|\psi\rangle}^{(\text{reg})} (\boldsymbol{\theta}) = \mathcal{L}_{|\psi\rangle} (\boldsymbol{\theta}) + \lambda \sum_{k'=0}^{k} \|\boldsymbol{\theta} - \boldsymbol{\theta}_{k'}\|^{-2}. \tag{8}$$

Yet such regularizers are doomed to fail due to the algebraic structure of $S[|\psi\rangle]$. Not only are the products of symmetries also symmetries, generating $O(\exp k)$ symmetries from a given set

of size $O(k)$, but the symmetries are generally continuous families forming a Lie subgroup $S_d[|\psi\rangle] \leqslant U(2^L)$. Consider, for example, the $L$-qubit GHZ state

$$|\text{GHZ}_L\rangle = \frac{|0^L\rangle + |1^L\rangle}{\sqrt{2}}. \tag{9}$$

For simplicity, consider $d = 0$ (factorizable) symmetries $S_0[|\text{GHZ}_L\rangle]$. Under a manifold picture of the Lie subgroup, the factorizable symmetries can be partitioned into two submanifolds

$$\left\{ \bigotimes_{k=1}^{L} \begin{pmatrix} e^{i\alpha_k} & 0 \\ 0 & e^{i\beta_k} \end{pmatrix} \Bigg| \sum_{k=1}^{L} \alpha_k = \sum_{k=1}^{L} \beta_k = 1 \right\},$$
$$\left\{ \bigotimes_{k=1}^{L} \begin{pmatrix} 0 & e^{i\alpha_k} \\ e^{i\beta_k} & 0 \end{pmatrix} \Bigg| \sum_{k=1}^{L} \alpha_k = \sum_{k=1}^{L} \beta_k = 1 \right\}. \tag{10}$$

The first submanifold is composed of diagonal matrices with phases as their entries, and includes the identity matrix. The second submanifold is composed of off-diagonal matrices with phases as their entries, and includes the bit flip symmetry $X$ where $X$ is the Pauli spin operator $\sigma_x$. Each submanifold has four degrees of freedom, forming a four-torus $T^4$. These may be equivalently expressed in circuit decomposition as

$$M_1 = \left\{ \bigotimes_{k=1}^{L} XP(\alpha_k) XP(\beta_k) \right\},$$
$$M_2 = \left\{ \bigotimes_{k=1}^{L} P(\beta_k) XP(\alpha_k) \right\}, \tag{11}$$

where $P(\varphi)$ is the phase gate

$$P(\varphi) = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\varphi} \end{pmatrix}. \tag{12}$$

Each submanifold $M_i$ is a smoothly parameterized set of symmetries. Thus, in principle we need only inspect a few points in the submanifold to deduce a closed-form expression for $M_i$.

### 2.4. Interactive classical learning

To learn efficiently, we cannot explore the same submanifold repeatedly while other manifolds remain unexplored. Rather, we must develop a procedure to push the PQC away from symmetry submanifolds already explored sufficiently. While such a task is not obvious, it is not impossible from a measure-theoretic standpoint. That is, there will never be a case in which $|\psi\rangle$ has two symmetry submanifolds $M_1, M_2$ such that $\dim(M_1) < \dim(M_2)$, so that $M_1$ would never be found. For as the manifolds are also groups, there exists automorphisms mapping $M_1$ to $M_2$ and vice versa, and hence the topological dimension of every symmetry submanifold must be identical. Nonetheless, no pointwise regularization procedure, such as that of equation (8), can recognize two symmetries being of the same submanifold and thereby push the PQC away from proposing both of them.

We therefore introduce a regularization method based on classical deep learning: rather than keeping track of specific symmetries, we train the classical net (CNet) to approximate $\mathcal{L}_{|\psi\rangle}(\boldsymbol{\theta})$

by its output $\hat{\mathcal{L}}_{|\psi\rangle}(\boldsymbol{\theta})$. The direct statistical estimation of $\mathcal{L}_{|\psi\rangle}(\boldsymbol{\theta})$ requires quantum operations and measurement handled by the QNet, so we supervise the CNet using the QNet itself. Specifically, for every batch of loss-functional evaluations the QNet performs as it searches for symmetries, we train the CNet in parallel using those evaluations as the supervisory dataset. Each batch is a transcript of the path in $\mathrm{U}(2^L)$ (henceforth abbreviated to U) walked by the QNet that terminates at a symmetry. Ultimately, the aim of the CNet is to estimate with low error the loss for $\boldsymbol{\theta}$ in a neighborhood of each $\boldsymbol{\theta}$ explored by the QNet. That is, we train the CNet by minimizing the squared error

$$\Delta_C(\boldsymbol{\theta}) = \left( \mathcal{L}_{|\psi\rangle}(\boldsymbol{\theta}) - \hat{\mathcal{L}}_{|\psi\rangle}(\boldsymbol{\theta}) \right)^2 \tag{13}$$

where $\hat{\mathcal{L}}_{|\psi\rangle}$ is the CNet estimate of the true loss.

Since the landscape of the loss function can be highly complex, we cannot hope to obtain a good classical estimate of the loss globally with only a few local training points. Instead, we design the CNet to estimate $\mathcal{L}_{|\psi\rangle}(\boldsymbol{\theta})$ locally in parameter space around points $\boldsymbol{\theta}$ explored by the QNet with very low error, but estimate regions far from points explored by the QNet with high error. By continuity, we can equivalently consider the local regions to be neighborhoods around the unitary itself in $\mathrm{U}(2^L)$. A useful intuition for the action of the CNet is to interpret the estimation of local patches as 'overfitting' the CNet, purposefully, to the areas of the unitary space that have already been explored. As the CNet trains, it will accurately predict $\mathcal{L}_{|\psi\rangle}(\boldsymbol{\theta})$ is $\boldsymbol{\theta}$ is near a parameter already explored, but will likely give a very inaccurate prediction if $\boldsymbol{\theta}$ is far from any explored parameters. Thus, by analyzing the CNet prediction accuracy on a point $\boldsymbol{\theta}$, given by $\Delta_C(\boldsymbol{\theta})$, the QNet can determine whether $C_{L,d}(\boldsymbol{\theta})$ resembles a symmetry previously explored (i.e. the absolute difference of their overlaps are small). If so, the QNet can adjust accordingly by leaving the local region of parameter space to hopefully find another region of U that has not yet been explored. We found numerically (see appendix D) that the average difference in $\Delta_C$ evaluated on a training point versus a random point is about two orders of magnitude, making for easy distinction.

Classical deep neural nets are well-suited to the task because a CNet can extrapolate relatively well locally. Consequently, as we will also show empirically, the CNet accomplishes what no pointwise regularization scheme can: it regularizes against entire continuous families of symmetries. Since we train the CNet on the entire search path, including points in U not corresponding to symmetries, the CNet regularizes against any subsets of U that the QNet has already explored, even if they are not symmetry submanifolds. Thus, the CNet minimizes the time the QNet wastes exploring parts of U already well-understood. Eventually, as the QNet explores more and more symmetries, all of the symmetry submanifolds of fixed $d$ will be learned. The CNet will then find low estimation error everywhere, at which point the learning algorithm will terminate. Thus when classical learning is added to the QNet VQA, we find empirically that the symmetry learning process converges.

As shown in the bottom half of figure 2, we develop a CNet with two three-dimensional convolutional layers. We design the convolutions to respect the structure of the parameterization of the PQC. In particular, the first layer scans over each set of 3 rotation parameters per qubit per block-depth, and the second layer scans over each individual parameter. The convolutional net layer is connected to a simple three-layer fully-connected neural net with leaky rectified linear unit (ReLU) activation functions [36].

We train the CNet in path-batches; during each symmetry-learning epoch, the CNet reports $\Delta_C$ while the QNet walks Nelder–Mead paths in parameter space, corresponding to a path in U. At the end of each epoch, when a symmetry is proposed, we use batch stochastic gradient

descent (SGD) on the transcript of points walked in the path. Note that the CNet cannot undergo training during an epoch of QNet symmetry search. Otherwise, the CNet will extrapolate the local path well and mislead the QNet into thinking that it has already traversed its current path. We elect to use SGD instead of an algorithm similar to Nelder–Mead for its superior empirical performance (see appendix D).

At Every $N$ steps on the QNet search path, the QNet pauses the search and queries the CNet. If the CNet estimation error is low, then the QNet must adjust its search path accordingly. We propose two methods by which the QNet can redirect its search. In the first, which we call the global method, the QNet simply randomly restarts. Since the parameter dimension is

$$\dim \boldsymbol{\theta} = O\left(Ld\right), \tag{14}$$

the dimensionality of search space is sufficiently large for even moderate $L, d$ that random restarts give a substantial likelihood of landing in an as-yet traversed region of U. $N$ is chosen to be much smaller than the number of steps to find a symmetry, but still reasonably large ($\sim$100), in consideration of a tradeoff between wasting too much time querying the CNet and wasting too much time searching a familiar region.

The second method, which we call the local method, alternates between finding symmetries (optimization) and maximizing the classical estimation error $\Delta_C$ (exploration). That is, instead of randomly jumping to a new region, we directly walk in the direction that appears most unfamiliar. In the exploratory phase, we use finite difference gradient descent implemented by the shift rule [37],

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \eta \left( \frac{\mathcal{L}_{|\psi\rangle}\left(\theta_{101} + \delta, \theta_{102}, ..., \theta_{3d3}\right) - \mathcal{L}_{|\psi\rangle}\left(\theta_{101} - \delta, \theta_{102}, ..., \theta_{3d3}\right)}{2\delta}, \cdots \right), \tag{15}$$

where $\eta$ is the learning rate and $\delta$ is the finite difference parameter. We favor the shift rule over Nelder–Mead because the latter fails to update $\boldsymbol{\theta}$ if all points in the neighboring simplex have similar $\Delta_C$; hence, no exploration occurs. By contrast, gradient descent will still update $\boldsymbol{\theta}$, allowing progressive movements away from known regions. The local method is more efficient for states that are highly symmetric (i.e. many symmetry submanifolds close to each other, for a reasonable metric of submanifolds on U). The global regularizer has an advantage on states with high-dimensional parameter spaces, wherein random restarts are generally faster than explicitly carving paths to unexplored regions.

The full learning scheme is given in figure 2. As Nelder–Mead has been shown to be average-case efficient, under the assumption that the regularizer prevents the same manifold from being explored more than polynomially many times, our algorithm solves the symmetry learning problem efficiently. In the absence of regularization, symmetry learning may take superexponential time or even fail to terminate at all.

### 2.5. Adjustments for near-term practical implementation

Due to the limitations of near-term quantum circuits, the symmetry learning algorithm as formulated above will be difficult to implement with current hardware. Specifically, both the $\text{SWAP}_\epsilon$ test and the PQC given in figure 4 are spatially nonlocal computations—they contain gates that span $O(L)$ qubits. In this section, we adjust the algorithm to use only local operations, thereby becoming realizable with near-term hardware, at the cost of some efficiency and universality.

A SWAP operation between two $L$-qubit registers is implemented via $L$ single-qubit SWAP gates spanning $O(L)$ qubits. While recent advancement in Rydberg atom quantum computers have made long-range SWAPs a reality [38], in more general quantum hardware the SWAP gate must be decomposed into controlled-NOT gates as

$$\text{SWAP}_{ij} = \text{CNOT}_{i \to j} \text{CNOT}_{j \to i} \text{CNOT}_{i \to j}. \tag{16}$$

Hence a controlled-SWAP can be carried out in $O(L)$ Toffoli gates. However, such gates span $O(L)$ qubits. An alternative verification procedure to the $\text{SWAP}_\epsilon$ test is via statistical comparison of the projective measurements (i.e. their distribution in a particular measurement basis) of $|\psi\rangle$ with that of $U|\psi\rangle$. For any fixed basis $b$, we may estimate the classical Kullback–Leibler (KL) divergence $\text{KL}_b(|\psi\rangle, U|\psi\rangle)$ between the measurement distributions of $|\psi\rangle$ and $U|\psi\rangle$ by constructing a sampling-based empirical cumulative distribution function (ECDF) for each state in the basis expansion of $b$. Choosing multiple bases to capture the structure of the phases, we define the quantum KL divergence as a sum over the KL divergence of each projective measurement:
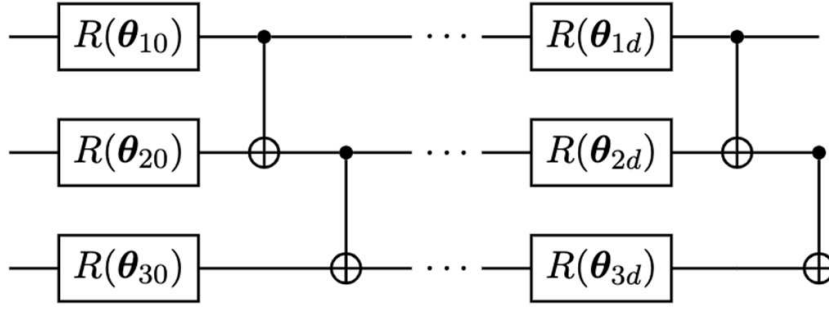
$$\text{QKL}_{|\psi\rangle}(U) = \sum_b \text{KL}_b(|\psi\rangle, U|\psi\rangle). \tag{17}$$

In practice, we find that only two bases, which need not be random, are required to capture the contributions of the phases (see appendix B). Generally, estimation of $\text{QKL}_{|\psi\rangle}(U)$ requires $O(\exp L)$ evaluations of $U$ just to sample every basis element at least once on average. However, if only a small (polynomial) subset of the basis elements have non-negligible amplitude, the QKL loss may be evaluated efficiently. Thus, although the exponential search is generally unavoidable, we can first search for a polynomially sparse basis by checking the $z$ and $x$ bases and choosing the sparser one. Further optimizations can be done via a method of binary search to find a sparse basis. A second basis can then be found by rotating the sparsest basis by a small angle (we chose $\pi/10$). Such basis optimizations should be done before beginning the learning algorithm, since checking a basis requires one sampling round, whereas the learning algorithm will use hundreds of thousands or more sampling rounds, one for each step in the search path. With preliminary basis finding, the QKL verification remains relatively efficient, and we define the near-term loss as

$$\mathcal{L}_{|\psi\rangle}(\boldsymbol{\theta}) \overset{\text{near-term}}{\longrightarrow} \text{QKL}_{|\psi\rangle}(C_{L,d}(\boldsymbol{\theta})). \tag{18}$$

The second modification is of the PQC ansatz. The form given in figure 4 is universal, but has $\binom{L}{2}$ CNOT gates in each layer, most of which span $O(L)$ qubits. Ideally, a hardware-efficient circuit family has only gates spanning $O(1)$ qubits and scales in depth at most linearly with $L$. Thus, we restrict the CNOT gates to only nearest neighbors, giving a hardware efficient circuit shown in figure 5. Although the family may no longer be universal, or at least require much longer depths to represent a general unitary, it can still represent most symmetries of practical interest, including any expressible in terms of local couplings.

For present applicability, we adopt both adjustments in this paper. Note that our algorithm is agnostic to the choice of PQC family and loss function, so these hardware efficient modifications can be readily removed when the appropriate experimental hardware is realized.

**Figure 5.** Restricted quantum circuit family, of *d* layers, appropriate for current noisy devices capable of local operations. Cross-hairs represent controlled-NOT gates, while *R* gates are single-qubit rotations, parameterized by $\boldsymbol{\theta}$.
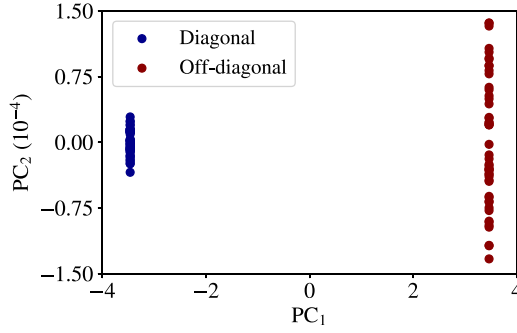
## 3. Results and discussion

We demonstrate our results on four quantum states: a Bell state, the GHZ state from equation (9) and the planar cluster state [39], which both arise from a broader family known as stabilizer states, and the transverse Field Ising Model (TFIM) which is known as one the simplest quantum spin chains. Much of our focus is on the GHZ state, for due to their sensitivity to decoherence, they are often utilized for characterization of noisy quantum hardware [40–42] and other quantum technologies, such as error correction, quantum metrology, and quantum communication [43, 44]. We will first visualize some symmetries of the states. We then benchmark the scaling of the algorithm with respect to *L* and *d* as well as its robustness against noise. We collect data by backend classical simulation in Qiskit [45], although our results on simulated noise indicate comparable results on quantum hardware.

### 3.1. Symmetries of simple circuit states

The exponential size of the matrix representation of $\boldsymbol{\theta}$ implies that there is no simple method to visualize a general symmetry aside from its circuit decomposition; we discuss further this issue at the end of the paper. However, for simple cases, we can explicitly examine the geometric or analytical structure of the symmetry, which we discuss for the *L*-qubit GHZ state, measured in the *z* basis for maximal sparseness. When $d = 0$, each symmetry factors as a tensor product of $2 \times 2$ matrices, which belong to one of the submanifolds in equation (11). Since they are either diagonal or off-diagonal, we use principal component analysis (PCA) in 2 dimensions to visualize their geometric structure, illustrated in figure 6 for $L = 3$. The diagonal and off-diagonal symmetries form distinct clusters, which is consistent with the orthogonality of their analytical representations.

Beyond $d = 0$, no such tensor product decomposition exists, so the symmetries become more difficult to visualize. (One can do so, though, by using PCA on $\boldsymbol{\theta}$ itself and then clustering the resultant dataset.) However, for $L = 2$, the unitary implemented by $C_{2,d}$ may be explicitly written. A key symmetry in many entangled states is the swap symmetry, which require at least $d = 3$ to be represented, by equation (16). We test the algorithm on a Bell basis state

$$|\Phi\rangle = \frac{|01\rangle + |10\rangle}{\sqrt{2}} \tag{19}$$

**Figure 6.** PCA projection of 100 symmetries of the 3-GHZ state, found by the QNet, into 2 dimensions. The separation between them implies that the parameterizations of the manifolds are easily distinguishable, which we know analytically to be true.

to numerically obtain the swap symmetry. On average, after ~10 trials, we find many symmetries in the family

$$
U = e^{i\varphi}\begin{pmatrix} \cdot & c_1 & -c_1 & \cdot \\ \cdot & \alpha & 1-\alpha & \cdot \\ \cdot & 1-\beta & \beta & \cdot \\ \cdot & c_2 & -c_2 & \cdot \end{pmatrix},
\tag{20}
$$

where $\alpha, \beta, c_j \in \mathbb{C}$ and the dots representing arbitrary numbers to make $U$ unitary. In the limit of $c_j, \alpha, \beta \to 0$, $U$ reduces to the familiar SWAP operation, but the algorithm finds a more general family that may be interpreted as a partial swap operation on each of the $|01\rangle$ and $|10\rangle$ basis states. More generally, we find a similar family for the $L$-GHZ state, with a copy of the $2 \times 2$ submatrix at the center of equation (20) present in each two-qubit subspace. This is a general principle of symmetry learning: one samples from a smooth symmetry submanifold, and must take a few samples to interpret the underlying family of symmetries being represented.

### 3.2. Symmetries of adiabatically prepared states

In the following section, we will investigate our algorithm on further families of states. Before doing so, we note that we have yet to use the black-box assumption (that no information is known about the state) of the states. The motivation for black-box states is that in experiment one may not have efficient access to the inverse circuit used to prepare the state because the preparation may involve, for example, unitary time evolution. This is clearly different for the GHZ family. Often states that require time evolution are both the most interesting and the most difficult to study analytically. As such, learning their symmetries sheds insight into the state itself and provides utility of our algorithm beyond the computational boosts discussed in section 1. One important example of a family of states prepared with time evolution is the Rydberg family. Specifically, consider the 1-dimensional Rydberg atom chain defined as the ground state of the Hamiltonian

$$
\frac{\mathcal{H}(t)}{\hbar} = \sum_j \frac{\Omega_j(t)}{2}\left(e^{i\phi_j(t)}|0_j\rangle\langle 1_j| + e^{-i\phi_j(t)}|1_j\rangle\langle 0_j|\right) \quad - \sum_j \Delta_j(t)\hat{n}_j + \sum_{j<k}V_{jk}\hat{n}_j\hat{n}_k,
\tag{21}
$$

where $\hat{n}_j = |1_j\rangle\langle 1_j|$, $\Omega_j$ is the Rabi frequency for qubit $j$, $\phi_j$ is the laser phase, and $\Delta_j$ is the laser detuning frequency. $V_{jk}$ is the potential, which follows the van der Waals power law $V_{jk} = C_6/|\mathbf{r}_i - \mathbf{r}_j|^6$, where $C_6 = 862 \times 2\pi$ MHz $\mu$m$^6$. In recent years interest in Rydberg atoms has exploded due to their potential as scalable neutral-atom quantum computers and programmable simulators, as well as their interesting quantum dynamics [28, 46–50]. When the parameters are time-independent and uniform, i.e. $\Omega_i = \Omega_j$ and $\Delta_i = \Delta_j$ for all $i$ and $j$, the (matter) phase of a Rydberg atom array is determined by two parameters, $\Delta/\Omega$ and $R_b/a$, where $a$ is the lattice constant and $R_b$ is the blockade radius defined by $C_6/R_b^6 = \Omega$. We thus label states as $|\Delta/\Omega, R_b/a\rangle$. Physically, the blockade radius is the minimum distance between two Rydberg atoms for them to be simultaneously excitable, and is closely related to the NP-complete problem of Maximum Independent Set [46, 51].

For concreteness, we study the 7-qubit Rydberg chain with open boundary conditions, setting $\phi_j = 0$. Under these conditions, the Rydberg chain has two ordered phases, denoted $\mathbb{Z}_2$ and $\mathbb{Z}_3$, which are nearly classical because they look like $|\mathbb{Z}_2\rangle = (1-\epsilon)|1010101\rangle + \epsilon|\text{other}\rangle$ and $|\mathbb{Z}_3\rangle = (1-\epsilon)|1001001\rangle + \epsilon|\text{other}\rangle$. In the literature, these phases are also said to obey a discrete permutation symmetry because the $\mathbb{Z}_n$ phase is invariant, up to a small error, when cyclically permuted $n$ times [52]. Symmetry learning provides an alternative way to probe ordered phases numerically. Although there are much easier ways to deduce this symmetry, such as just ansatzing some possible spatial translation symmetries and checking them explicitly, this example provides evidence that symmetry learning can become useful to learn much more complicated symmetries of physically interesting ground states in the future.
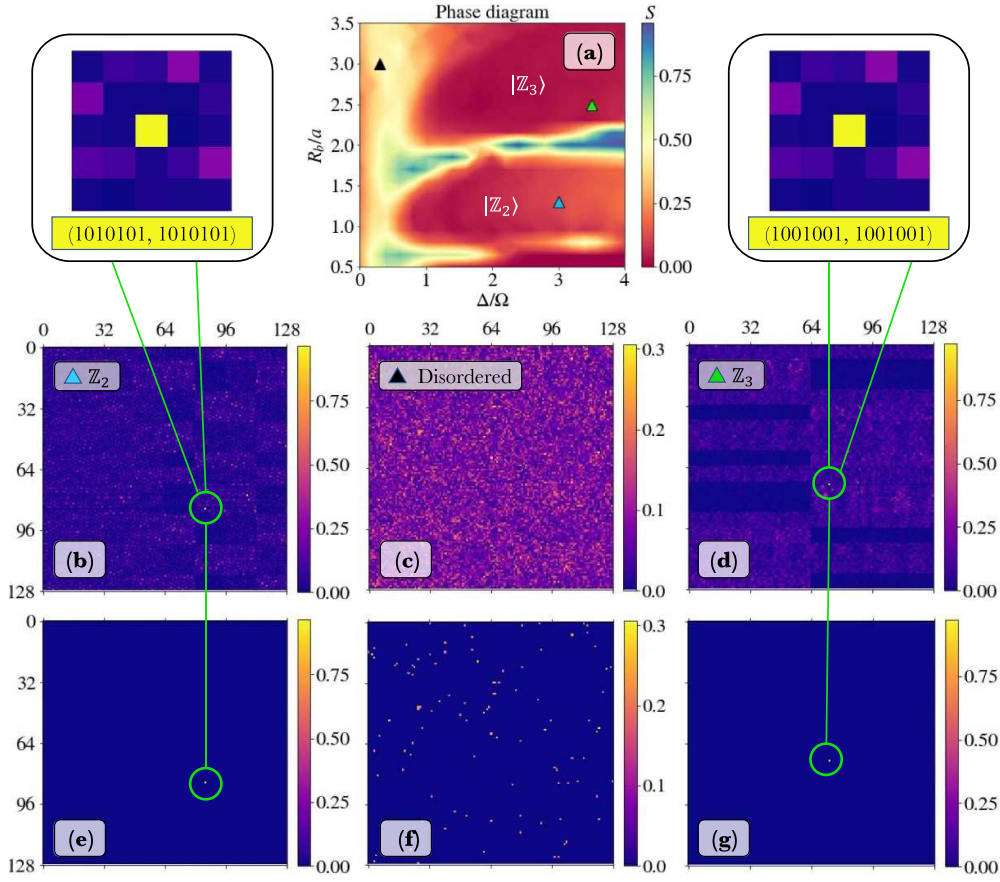
To simulate the Rydberg arrays, we use the technique of adiabatic evolution. Adiabatic evolution achieves the ground state of a target Hamiltonian by starting with an easily preparable ground state of an initial Hamiltonian, and then slowly modifying the Hamiltonian to reach the target. In practice, the time scale of the parameter variation is $\sim 3\,\mu$s. This variation can be thought of as iteratively time-evolving the state according to a smoothly parameterized Hamiltonian, resulting in the state

$$|\psi(t)\rangle = e^{-i\int dt\,\mathcal{H}(t)}|\psi(0)\rangle. \tag{22}$$

Adiabatic evolution is a key technique to Rydberg-atom quantum computing [47, 53]. We set $|\psi(0)\rangle = |0\rangle$ for simplicity, though any easily preparable state works equivalently. We prepare Rydberg states in classical simulation using Hamiltonians generated from Bloqade [54]. Having simulated experimental conditions, we proceed to examine their symmetries. Figure 7(a) shows the phase diagram of the 7-qubit Rydberg chain, calculated by the entanglement entropy

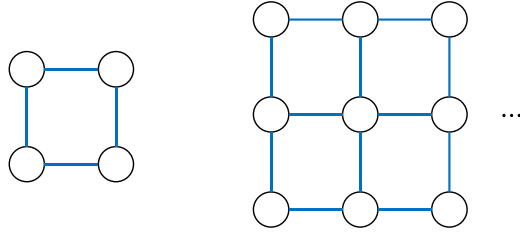$$S[|\Delta/\Omega, R_b/a\rangle] = -\operatorname{Tr}[\rho_A \log \rho_A] \tag{23}$$

where $\rho_A$ is the reduced density matrix of $|\Delta/\Omega, R_b/a\rangle$ calculated by tracing over half of the chain. The two lobes correspond to the ordered phases. From the diagram we examine three points: $|3.0, 1.3\rangle$ in the $\mathbb{Z}_2$ phase, $|3.5, 2.5\rangle$ in the $\mathbb{Z}_3$ phase, and $|0.3, 3.0\rangle$ in neither (the disordered phase). Figures 7(b)–(d) respectively visualize a representative symmetry from these phases, by plotting $|U_{ij}|^2$ for each [55]. Similar results are found by repeatedly running the algorithm until the CNet stops it. In (b) and (d), almost all elements look like random noise except for a single bright point, which we zoom in on in the insets. In binary, the indices correspond to exactly the $\mathbb{Z}_2$ and $\mathbb{Z}_3$ phases, since they are respectively close to $|1010101\rangle$ and $|1001001\rangle$. As such, the corresponding matrix element in the symmetries are close to 1, while all others are random noise far below 1. This visual is even more clear when we choose a cutoff $\alpha$ and set each element $|U_{ij}|^2$ to 0 unless $|U_{ij}|^2 \geq \max_{ij}|U_{ij}|^2 - \alpha$; this is shown in

**Figure 7.** Visualization of the symmetries of the 7-qubit Rydberg chain $|\Delta/\Omega, R_b/a\rangle$, whose ground state is prepared by adiabatic evolution. (a): the phase diagram based on the entanglement entropy $S$ of $|\Delta/\Omega, R_b/a\rangle$ with two ordered lobes, (b)–(d): matrix visualizations $|U_{ij}|^2$ of a representative symmetry $U \in \mathbb{R}^{2^7 \times 2^7}$ for three Rydberg ground states. Most of the matrix elements look like random noise, but on the $\mathbb{Z}_2$ and $\mathbb{Z}_3$ there is a clear single element that is close to 1, which is zoomed in on the inset. The 'signal' matrix elements are more easily seen in (e)–(g), which send all elements where at least $\alpha = 0.1$ smaller than the maximum element to 0.

(e)–(g) for $\alpha = 0.1$. On the other hand, the 'symmetry' for the disordered phase, which has no special structure, resembles random noise, both in (c) and (f). This lack of structure reflects the disordered nature of the state.

Three observations are of importance in the Rydberg example. First, although we used adiabatic preparation which produced a much more imperfect phase diagram as compared to exact diagonalization (which has smooth and almost entropyless ordered phase lobes), we nonetheless managed to learn symmetries. This illustrates the robustness of our algorithm to the more noisy and imperfect experimental settings and will be explored further in the next section via precise noise models. Second, we deduced from figure 7 that the symmetries of the $\mathbb{Z}_2$ phase were matrices $U$ of the form $U_{85,85} = 1 - \alpha_{85,85}$ ($85 = 1010101$ in binary) and all other $U_{ij} = \alpha_{ij}$, where the $\alpha$'s are small. This large 'signal' matrix element indicates that the state itself is nearly classical, hinting that the phase is highly ordered. The same analysis

**Figure 8.** Graph representation of planar cluster states for $L = \ell^2$ qubits. The nodes are qubits and the edges are controlled-$Z$ gates. The cluster state is thus defined as $\Pi_{(a,b) \in \text{Edges}} C_a Z_b |+\rangle^{\otimes L}$ where $C_a Z_b$ is a control-$Z$ gate between qubits $a.b$ and $|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$.

follows for the $\mathbb{Z}_3$ phase, and the opposite for the disordered phase, which allow us to conclude that there is no simple structure for disordered states. Thus, from symmetry learning, we have deduced most of the salient aspects of the Rydberg phase diagram, which in turn shed light on the physics. Symmetry learning, therefore, provides a new way to probe physics described by Hamiltonians. Lastly, while figure 7 shows that there is no simple structure to the symmetries of the disordered phase, little further insight can be obtained by inspecting the matrix alone. We discuss this problem further in section 4.

As we conduct numerical scaling analyses in the next section, we continue to consider ground states of Hamiltonians that require the black-box assumption, but turn to an Ising model for simplicity.
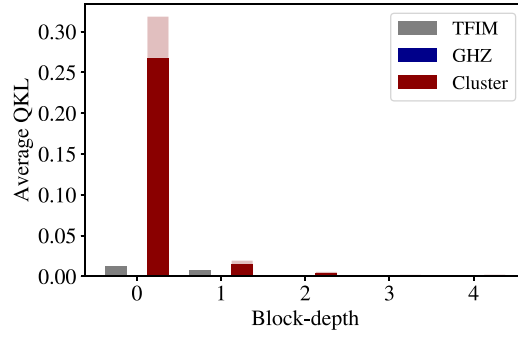
### 3.3. Larger degrees of freedom

As shown in equation (6), more symmetries are representable by $C_{L,d}$ with increasing $d$. Inversely, the minimum $d$ required for the learning algorithm to find symmetries is a state complexity measure. For example, although every $S_0[|\psi\rangle]$ is nonempty for any $|\psi\rangle$ (due to the identity matrix), the rotation symmetries among all possible single-qubit rotation operations may be sparse and thus difficult to find. Thus, relatively few epochs will successfully find a symmetry. For a given $d$, the average loss over many (say, 100) epochs on a given state measures the difficulty of learning symmetries at that block-depth. (We assume implicitly here that trainability is not a problem, which breaks down for $d$ larger than what we probe here, due to the general vanishing gradient problem in machine learning.) We can examine the average loss on various states to compare the complexity of their symmetries—that is, how large $d$ must be before most epochs can find symmetries. Importantly, this simple measure can be done without knowing the actual symmetries themselves; we do not even need to look at the symmetries to understand the relation between the loss and the block-depth.

We demonstrate the above procedure on the GHZ state, the planar cluster state, and the ground state of the TFIM Hamiltonian. The planar cluster state is represented as a graph state for which nodes are qubits and edges are controlled-$Z$ gates, where $Z$ is the Pauli $\sigma_z$ operator. The planar cluster states are defined only for $L = \ell^2$ qubits, and are shown in figure 8.

The TFIM Hamiltonian is given by

$$\mathcal{H} = 0.5 \sum_i S_i^z - J \sum_i S_i^x S_{i+1}^x, \tag{24}$$

**Figure 9.** Average loss for 4-qubit GHZ, TFIM, and cluster states as a function of $d$. The GHZ losses are all negligibly small and thus invisible. The transparent bar indicates one standard deviation.

where $J = 1$ is the interaction strength and the $\sigma_i$ are the Pauli spin operators on the $i$th qubit. While such variational methods as VQE and QAOA can be used to prepare the TFIM ground state, we do so numerically via exact diagonalization.
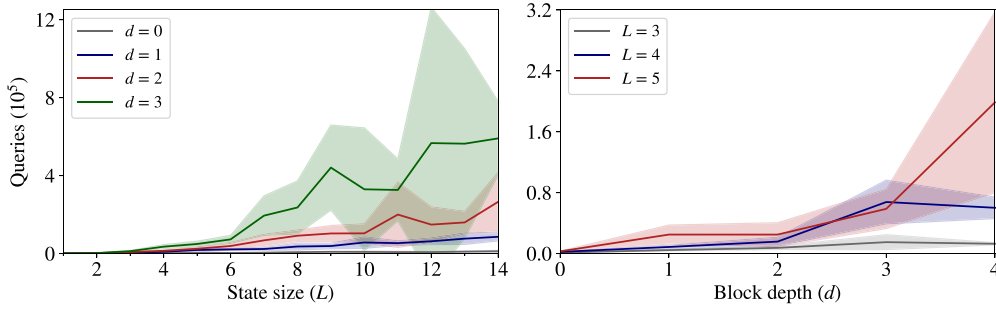
Our preliminary analysis shows that the TFIM state is maximally sparse in the $x$ basis and cluster/GHZ states in the $z$ basis. Finding 100 symmetries for each depth with $L = 4$ yields an average loss given in figure 9. As we might expect, the cluster state has the most complex symmetries, requiring $d \approx 2$ to easily find symmetries with every epoch. The GHZ state is the simplest due to its well-studied $d = 0$ symmetries from figure 6. In practice, we can resolve the issue of hard-to-find symmetries for small $d$ by postselecting on epochs that do find symmetries, using regularization to prevent traversing known paths. However, figure 9 shows that for relatively small $d$ even complicated states like the cluster state permit most epochs to learn symmetries, evidencing that our hardware-efficient PQC ansatz remains sufficiently rich to efficiently parameterize non-trivial symmetries.

### 3.4. Query complexity

We benchmark the numerical scalability of the learning algorithm both with respect to $d$ and $L$, using the $L$-GHZ state as a concrete example. We measure scalability in terms of the search query complexity; that is, the average number of Nelder–Mead iterations per epoch of symmetry learning. The advantage of query complexity is that it is agnostic to the verification procedure, so our results hold even without the near-term related modifications. With current devices, short-depth implementations are of the most significance. Demonstrated on the GHZ state, figure 10 shows that the learning algorithm scales reasonably with small $d$. The query complexity also scales well with respect to $L$ on average. Such findings give numerical credence to our previous claims about the general efficiency of the learning algorithm, at least for moderately sized systems.

### 3.5. Regularization performance

The heart of the CNet is a classical estimation of $\mathcal{L}_{|\psi\rangle}(\boldsymbol{\theta})$. The more rapidly the local landscape of a symmetry submanifold can be fully learned (i.e. estimated to low error) by the CNet, the more efficiently our scheme finds new symmetries. Measurement of classical learning efficiency manifests differently in the global and local methods. For each, we define a

**Figure 10.** Average query complexity (number of search iterations to find a symmetry) of the GHZ state as a function of state size $L$ (left) and block-depth $d$ (right), estimated with 10 trials per $L$ per $d$. Shading represents standard deviation.

characteristic *learning time*; for the global scheme,

$$\tau_G = \frac{1}{N} \sum_{n=1}^{N} \mathrm{argmin}_{t \geqslant 1} \left[ M\left(\boldsymbol{\theta}_0^{(n)}\right) \neq M\left(\boldsymbol{\theta}_t^{(n)}\right) \right] \tag{25}$$
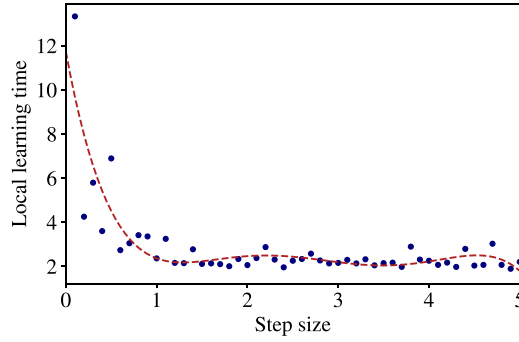
where $\boldsymbol{\theta}_t^{(n)}$ is the symmetry proposed on the $t$th epoch of the $n$th fully-reset search (i.e. the CNet is re-initialized to random weights) starting from a random $\boldsymbol{\theta}$, and $M(\boldsymbol{\theta})$ is the symmetry submanifold containing $\boldsymbol{\theta}$. For large $N$, $\tau_G$ represents the expected number of epochs of training from Nelder–Mead paths the CNet requires to learn a randomly chosen symmetry manifold. For an ideal regularizer, $\tau_G = 1$; for a useless regularizer, $\tau_G \to \infty$.

On the other hand, for the local scheme. As the QNet alternates between optimization and exploration with no random restarts, the key metric of its performance is thus the frequency with which the QNet crosses to a different manifold and back. Hence the learning time may be defined implicitly by

$$\tau_L(T)^{-1} = \frac{1}{T} \sum_{t=1}^{T} \mathbb{1}\left[ M(\boldsymbol{\theta}_{t-1}) \neq M(\boldsymbol{\theta}_t) \right], \tag{26}$$

where $\boldsymbol{\theta}_t$ is the symmetry on the $t$th optimization epoch. For large $T$, $\tau_L$ has the same interpretation as $\tau_G$.

Explicit measurement of the learning time requires analytical knowledge of the set of all symmetry submanifolds. Since we have written that set for $S_0[|\mathrm{GHZ}\rangle_L]$, we will measure the learning times for both methods on the 3-GHZ state. In the global scheme, over 100 runs we found $\tau \approx 1.2$ regardless of the steps $N$ per CNet query, in a range of $N = 100$ to $N = 5000$. The learning time of the local method depends on the distance walked in the exploration phase, which can be measured either by the step size in the gradient descent procedure or in the number of iterations; we fix the latter and use the former, and plot the learning time (which can be interpreted as a scoring function since bigger is better) in figure 11 for $T = 400$, fitted by standard methods in the dashed line to guide the eye. The shift rule also requires a finite difference parameter $h$, but we found that the learning time does not depend significantly on it. Figure 11 illustrates the intuition that, for small step sizes/time spent exploring, the algorithm requires much more time to learn each manifold because it cannot walk sufficiently far away from a symmetry submanifold before the exploratory phase ends. For sufficiently
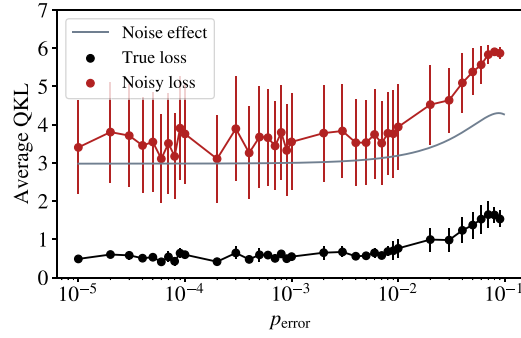
**Figure 11.** Local learning time $\tau_L(400)$ (blue dots) fitted with a trend line (red, dashed) for ease of visualization. $\tau_L$ represents the average number of points on a symmetry submanifold needed to be learned before the CNet recognizes the entire submanifold—the smaller $\tau_L$ is, the more efficiently the algorithm regularizes. For larger gradient descent step sizes, the learning time saturates.

large step size, however, ($\approx 5$ for 3-GHZ), the exploration has taken the QNet sufficiently far away from the known symmetry submanifold that more exploration is unnecessary. Although figure 11 benchmarks a simple state, the lessons taught by the results hold more generally in that a sufficiently large choice of step size is crucial to build an effective local regularizer.

### 3.6. Learning in the presence of noise

A final consideration for implementation in near-term devices is the robustness of the learning algorithm in the presence of noise. For simplicity, we consider a noise model, simulated classically, for which each gate (both in the state preparation and $C_{L,d}$) can incur a bit flip (erroneous $X$ error) or reset (qubit resets to $|0\rangle$), each independently with probability $p_{\text{error}}$. The measurement of each qubit may also be erroneous with the same probability. The presence of such noise may be physically interpreted as thermal fluctuations that jiggle the path traversed by the QNet in U. Because our QNet requires many iterations to propose a symmetry—$10^5$ or more for some systems, as shown in figure 10—we expect that such small thermal fluctuations average out in the process, at least up to a certain threshold of $p_{\text{error}}$.

We examine the loss metric $\mathcal{L}_{|\psi\rangle}(\boldsymbol{\theta})$ as a function of $p_{\text{error}}$ on the 3-GHZ state for $d = 1$ to demonstrate this effect numerically. The black curve in figure 12 represents the true average loss of the symmetry; that is, the QKL value between the shadows of $U|\text{GHZ}_3\rangle$ and $|\text{GHZ}_3\rangle$, where $U$ is the symmetry proposed by the noisy PQC. Until $p_{\text{error}} \sim 10^{-2}$, the loss remains relatively low and constant. However, in practice we cannot calculate the true loss due to noisiness of state preparation. When $|\text{GHZ}_3\rangle$ is itself is prepared with noise, the resultant average QKL value is given in red. As expected, they are much higher, but the difference between the two, plotted with standard smoothening methods in grey, also remains relatively constant until $p_{\text{error}} \sim 10^2$. Thus, when the error is below a reasonably high threshold, the learning algorithm continues to propose true symmetries, and the amount by which the loss increases due to the noisiness of state preparation varies little. By subtracting out this constant gap, we can account for the effect of the noise and thereby continue to learn symmetries in the presence of noise. Above the threshold, the loss increases roughly linearly and slowly, and symmetries may still be classified, albeit slightly noisily.

**Figure 12.** Average loss over 20 symmetries proposed by a noisy circuit with respect to a noisily prepared GHZ state (red) and a noiseless GHZ state (black), which represents the true loss of the noisy symmetry. Error bars show log-standard deviation. The grey line is a fitted curve of the difference in losses evaluated with respect to a noisy state and a noiseless state.

## 4. Conclusion and outlook

We developed an interactive hybrid quantum–classical learning algorithm that efficiently solves the symmetry learning problem—to discover and classify every symmetry of an unknown state $|\psi\rangle$ that can be represented by a certain quantum circuit family with a fixed block-depth $d$. We first showed that known symmetries could be verified efficiently via a quantum algorithm, and then devised a method to upgrade the verification procedure into a variational quantum learning algorithm that generated symmetries. To regularize against repetitive propositions of similar symmetries, we introduced an interactive protocol with a classical deep neural network that guided the variational quantum algorithm away from areas of unitary space $U(2^L)$ explored in previous iterations of learning. Thus, for each state we demonstrated it upon, the algorithm converged. For purposes of near-term implementation, we showed that aspects of the algorithm difficult to implement in current hardware could be replaced by presently realizable methods at the cost of some efficiency. We benchmarked our algorithm in simple cases and showed that it scales well with respect to $d$ and $L$. We further showed that the learning scheme is robust against noise.

Although the parameterized quantum circuit families we discussed are universal or near-universal, a general symmetry may require a very deep circuit in the family to be represented. In practice, under a restriction of depth, we can detect more symmetries by choosing various families of circuits and running the learning algorithm on each. Since some symmetries are more easily represented by certain circuit architectures than others, such an approach maximizes the number of symmetries found empirically on near-term devices.

An important next step in the utilization of our algorithm is more robust benchmarking on various quantum hardware, such as IBM digital quantum devices or Rydberg machines. More analysis on a broader set of states is also of interest. As for regularization, a more complex protocol may allow for even more efficiency. For example, a procedure based on the Metropolis-Hastings algorithm, which stochastically combines the global and local approaches proposed in this paper, is a natural next step of study. Indeed, recent work has shown that application of similar approaches based on Markov chain Monte Carlo methods significantly improve the efficiency of many variational quantum algorithms [56].

Beyond experimental considerations, we consider three points of inquiry stemming from our work. The first is physical: while we address the computational problem of symmetry learning, it remains unclear as to how the symmetries are to be physically interpreted. Historically, searching the symmetries of a system generally begins with some intuitive interpretation from which the equations are obtained. If, on the other hand, one has a general matrix or circuit which is known to be a symmetry, it is unclear as to how one would extract the physics from it. The second question is complexity-theoretic: although our work is manifestly computational, it should be possible to derive analytical complexity bounds on interactive quantum–classical search. These bounds should be considered in the context of local symmetries, or in other manners that do not enable barren plateaus to become an inhibitory problem. Numerous investigations of variational search complexities have been conducted in recent years, such as that by [57]. Lastly, perhaps the most difficult structure for the CNet to capture is the composition of unitaries. That is, if the QNet has learned $U_1$ and $U_2$, we do not want it to learn $U_1 U_2$ or $U_2 U_1$ since it is already clear that a composition of symmetries is a symmetry. Since the composition of a symmetry family $\mathcal{V}$ by another symmetry may not be close in any norm sense to $\mathcal{V}$. Hence, it is important that the CNet (or some other method) can regularize against compositions. The simulations in this paper indicated that the CNet was able to regularize well. However, more isolated analysis of the CNet alone is necessary to understand precisely what forms of similarity the CNet learns to regularize against.

In a broader context, symmetry learning can serve as a data-driven guide to learning about the physical phenomena of quantum systems and their relation to symmetries. For example, symmetry breaking is closely related to phase transitions and other emergent properties. Symmetry learning also aligns with the goals of partial tomography in providing a characterization of a state without completely reconstructing it classically. While recent methods such as shadow tomography [58] offer an efficient and convenient way to understand properties of a state from a computational perspective, symmetry learning may be of more relevance from a physics perspective, guiding researchers in quantum machine learning and shedding light on the physics of complex, empirically created quantum systems such as spin liquids [59] in addition to boosting other quantum learning algorithms that benefit from knowing symmetries.

## Data availability statement

The code generating the data in this paper is freely available for use [60].

## Acknowledgments

## Appendix A. Further discussion on symmetry learning

We provide some further discussion about the definition of symmetry learning. Given a state $|\psi\rangle$, if it is known in full, then one may write down every symmetry of $|\psi\rangle$ as

$$
U = \begin{pmatrix}
e^{i\theta} & u_{12} & \cdots & u_{12^N} \\
0 & u_{22} & \cdots & u_{22^N} \\
\vdots & \vdots & \ddots & \vdots \\
0 & u_{2^N 2} & \cdots & u_{2^N 2^N}
\end{pmatrix}
\tag{A1}
$$

where $U$ is expressed in a basis $\mathcal{B} = \{|\psi\rangle, v_2, ..., v_{2^N}\}$ where the vectors $v_k$ are arbitrary and the $u_{ij}$ can be anything that preserves unitarity. In this sense writing down the symmetries of a state is mathematically trivial if $|\psi\rangle$ is a known state. The existence of a closed-form expression of symmetries for a known quantum state $|\psi\rangle$ is, however, not inherently useful for two distinct reasons that both stem from the need to understand the symmetry physically. First, a general state prepared by quench dynamics according to a possibly time-dependent Hamiltonian in the experimental setting is generally unknown, so using $|\psi\rangle$ as a basis results in circular logic. Therefore, using the technique above to deduce symmetries is at least as hard as a full tomography, which is generally exponentially hard. We impose the black-box/unknown condition for $|\psi\rangle$ for this reason. Second, even if the state is known, it would be difficult to extract an efficient representation of the symmetry from the complete matrix form above. A variational technique enables the approximation of certain symmetries into an efficient representation, which takes a step towards garnering useful information from these symmetries. Future potential upgrades that make this VQA robust against barren plateaus may also provide a further advantage in making the entire learning protocol efficient on average.
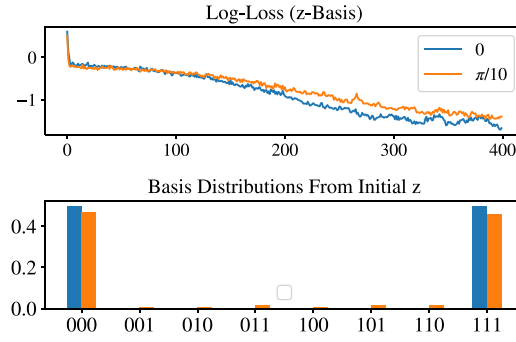
## Appendix B. QKL basis requirement

The definition of the QKL divergence requires a choice of a number of bases. For every family of states we considered, only 2 constant bases were necessary. We show this by defining a QKL cross validation loss and showing that for 2 constant bases, the cross validation is sufficiently small.

The methodology is as follows. Choose 2 bases to train on. These may be chosen randomly. However, if one knows that $|\psi\rangle$ has most of its amplitude concentrated in the small proportion of the basis elements in some basis, that basis will be optimal to train on. The second basis may be obtained by perturbing the optimal basis by a small rotation angle. We find that a rotation about an arbitrary axis of $\pi/10$ suffices to capture all of the phase information of the state.

To verify that 2 bases suffices, choose $n$ random bases after training and evaluate the QKL value over the random set. If it is under the desired error threshold, the two bases suffice. In our case, we found that the QKL over 3 random bases was of order $\sim 10^{-8}$ consistently.

## Appendix C. Hyperparameters

We use a error tolerance of $10^{-12}$ for Nelder–Mead search, with a maximum allowed iterations of $10^4$ for $L = 3$ qubits and up to $10^6$ for $L = 15$ qubits. The prediction error threshold for the global regularization scheme is $\delta = 10^{-2}$. The CNet uses 100 nodes per layer in the fully

**Figure 13.** Top: $\log \Delta_C$ of CNets under each basis (0 [blue] corresponds to the $z$-basis). Bottom: visualization of the shadows of the 3-GHZ state.

connected neural network component. We chose these parameters for their good numerical results and rapid convergence in the systems studied in this paper, but different states or those with sizes larger than the sizes considered in our study may require different choices.

## Appendix D. CNet demonstration

It is instructive to examine the classical deep net independently to verify its ability to learn the loss metric landscape. As an example, we consider the CNet on the 3-qubit GHZ state with a block-depth of $d = 2$. We generate a simple dataset of 3000 random parameters on two bases, one the computational (z) basis and the other a perturbation by a rotation of $\pi/10$ radians about a fixed axis. For each basis, the respective CNet learns the projected classical KL value on the training set, then attempt to estimate it on a testing set of 500 values. The resulting learning curve is given in figure 13.

The computational basis learns to a training error of 0.024 and incurs a cross-validation (test) error of 1.372, while the perturbed basis has a training/validation error of 0.038/1.224. The gap between the training and cross-validation error spans two orders of magnitude. Such effects, already visible in this toy example, are dramatically enhanced in practice wherein the training set is a local path rather than a random global sample, for the correlation between points is much higher, leading to a lower training loss and a higher cross-validation error. This is precisely the desired property of the CNet.

## ORCID iDs

Jonathan Z Lu ⬤ https://orcid.org/0000-0002-2806-8248
Rodrigo Araiza Bravo ⬤ https://orcid.org/0000-0002-5550-7576

## References

[1] Mototake Y-I 2021 Interpretable conservation law estimation by deriving the symmetries of dynamics from trained deep neural networks *Phys. Rev.* E **103** 033303
[2] Wetzel S J, Melko R G, Scott J, Panju M and Ganesh V 2020 Discovering symmetry invariants and conserved quantities by interpreting Siamese neural networks *Phys. Rev. Res.* **2** 033499

[3] Ha S and Jeong H 2021 Discovering conservation laws from trajectories via machine learning (arXiv:2102.04008)

[4] Liu Z, Madhavan V and Tegmark M 2022 Machine learning conservation laws from differential equations *Phys. Rev.* E **106** 045307

[5] Krippendorf S and Syvaeri M 2020 Detecting symmetries with neural networks *Mach. Learn.: Sci. Technol.* **2** 015010

[6] Broecker P, Carrasquilla J, Melko R G and Trebst S 2017 Machine learning quantum phases of matter beyond the fermion sign problem *Sci. Rep.* **7** 1

[7] Dong X-Y, Pollmann F and Zhang X-F *et al* 2019 Machine learning of quantum phase transitions *Phys. Rev.* B **99** 121104

[8] Carleo G and Troyer M 2017a Solving the quantum many-body problem with artificial neural networks *Science* **355** 602

[9] Zhang Y, Ginsparg P and Kim E-A 2020 Interpreting machine learning of topological quantum phase transitions *Phys. Rev. Res.* **2** 023283

[10] Deng D-L, Li X and Sarma S D 2017 Machine learning topological states *Phys. Rev.* B **96** 195145

[11] Käming N, Dawid A, Kottmann K, Lewenstein M, Sengstock K, Dauphin A and Weitenberg C 2021 Unsupervised machine learning of topological phase transitions from experimental data *Mach. Learn.: Sci. Technol.* **2** 035037

[12] LaBorde M L and Wilde M M 2022 Quantum algorithms for testing Hamiltonian symmetry *Phys. Rev. Lett.* **129** 160503

[13] LaBorde M L, Rethinasamy S and Wilde M M 2023 Testing symmetry on quantum computers *Quantum* **7** 1120

[14] Meyer J J, Mularski M, Gil-Fuster E, Mele A A, Arzani F, Wilms A and Eisert J 2023 Exploiting symmetry in variational quantum machine learning *PRX Quantum* **4** 010328

[15] Larocca M, Sauvage F, Sbahi F M, Verdon G, Coles P J and Cerezo M 2022 Group-invariant quantum machine learning *PRX Quantum* **3** 030341

[16] Zheng H, Li Z, Liu J, Strelchuk S and Kondor R 2023 Speeding up learning quantum states through group equivariant convolutional quantum ansätze *PRX Quantum* **4** 020327

[17] Zhou Z, Du Y, Tian X and Tao D 2023 QAOA-in-QAOA: solving large-scale MaxCut problems on small quantum machines *Phys. Rev. Appl.* **19** 024027

[18] Luo D, Chen Z, Hu K, Zhao Z, Hur V M and Clark B K 2023 Gauge-invariant and anyonic-symmetric autoregressive neural network for quantum lattice models *Phys. Rev. Res.* **5** 013216

[19] Luo D, Carleo G, Clark B K and Stokes J 2021 Gauge equivariant neural networks for quantum lattice gauge theories *Phys. Rev. Lett.* **127** 276402

[20] Paris M and Rehacek J 2004 *Quantum State Estimation* vol 649 (Springer Science & Business Media)

[21] Torlai G *et al* 2019 Integrating neural networks with a quantum simulator for state reconstruction *Phys. Rev. Lett.* **123** 230504

[22] Torlai G, Mazzola G, Carrasquilla J, Troyer M, Melko R and Carleo G 2018 Neural-network quantum state tomography *Nat. Phys.* **14** 447

[23] Carrasquilla J, Torlai G, Melko R G and Aolita L 2019 Reconstructing quantum states with generative models *Nat. Mach. Intell.* **1** 155

[24] Wang J, Han Z-Y, Wang S-B, Li Z, Mu L-Z, Fan H and Wang L 2020 Scalable quantum tomography with fidelity estimation *Phys. Rev.* A **101** 032321

[25] Carrasquilla J and Torlai G 2021 How to use neural networks to investigate quantum many-body physics *PRX Quantum* **2** 040201

[26] Cha P, Ginsparg P, Wu F, Carrasquilla J, McMahon P L and Kim E-A 2022 Attention-based quantum tomography *Mach. Learn.: Sci. Technol.* **3** 01LT01

[27] Huang H-Y, Kueng R, Torlai G, Albert V V and Preskill J 2022 Provably efficient machine learning for quantum many-body problems *Science* **377** eabk3333

[28] Gomez A M, Yelin S F and Najafi K 2022 Reconstructing quantum states using basis-enhanced Born machines (arXiv:2206.01273)

[29] We restrict ourselves to pure states $|\psi\rangle$ and system-isolated operators (i.e. acting only on $|\psi\rangle$) for simplicity. However, The algorithm generalizes readily to a density matrix formalism.

[30] Cerezo M, Sharma K, Arrasmith A and Coles P J 2022 Variational quantum state eigensolver *npj Quantum Inf.* **8** 113

[31]  Buhrman H, Cleve R, Watrous J and De Wolf R 2001 Quantum fingerprinting *Phys. Rev. Lett.* **87** 167902

[32]  Shende V V, Markov I L and Bullock S S 2004 Minimal universal two-qubit controlled-not-based circuits *Phys. Rev.* A **69** 062321

[33]  Nelder J A and Mead R 1965 A simplex method for function minimization *Comput. J.* **7** 308

[34]  Cerezo M *et al* 2021a Variational quantum algorithms *Nat. Rev. Phys.* **3** 625

[35]  Cerezo M, Sone A, Volkoff T, Cincio L and Coles P J 2021b Cost function dependent barren plateaus in shallow parametrized quantum circuits *Nat. Commun.* **12** 1791

[36]  Agarap A F 2018 Deep learning using rectified linear units (ReLU) (arXiv:1803.08375)

[37]  Grossmann C, Roos H-G and Stynes M 2007 *Numerical Treatment of Partial Differential Equations* vol 154 (Springer)

[38]  Bluvstein D, Levine H, Semeghini G, Wang T T, Ebadi S, Kalinowski M, Keesling A, Maskara N, Pichler H and Greiner M 2021 A quantum processor based on coherent transport of entangled atom arrays (arXiv:2112.03923)

[39]  Gottesman D and Chuang I L 1999 Demonstrating the viability of universal quantum computation using teleportation and single-qubit operations *Nature* **402** 390

[40]  Omran A *et al* 2019 Generation and manipulation of Schrödinger cat states in Rydberg atom arrays *Science* **365** 570

[41]  Song C *et al* 2019 Generation of multicomponent atomic Schrödinger cat states of up to 20 qubits *Science* **365** 574–7

[42]  Wei K X, Lauer I, Srinivasan S, Sundaresan N, McClure D T, Toyli D, McKay D C, Gambetta J M and Sheldon S 2020 Verifying multipartite entangled Greenberger-Horne-Zeilinger states via multiple quantum coherences *Phys. Rev.* A **101** 032343

[43]  Zhao Z, Chen Y-A, Zhang A-N, Yang T, Briegel H J and Pan J-W 2004 Experimental demonstration of five-photon entanglement and open-destination teleportation *Nature* **430** 54

[44]  Pezzè L, Smerzi A, Oberthaler M K, Schmied R and Treutlein P 2018 Quantum metrology with nonclassical states of atomic ensembles *Rev. Mod. Phys.* **90** 035005

[45]  M S A *et al* 2021 Qiskit: an open-source framework for quantum computing

[46]  Ebadi S, Keesling A, Cain M, Wang T T, Levine H, Bluvstein D, Semeghini G, Omran A, Liu J-G and Samajdar R 2022 Quantum optimization of maximum independent set using Rydberg atom arrays *Science* **376** eabo6587

[47]  Bernien H *et al* 2017 Probing many-body dynamics on a 51-atom quantum simulator *Nature* **551** 579

[48]  Ebadi S *et al* 2021 Quantum phases of matter on a 256-atom programmable quantum simulator *Nature* **595** 227

[49]  Levine H, Keesling A, Semeghini G, Omran A, Wang T T, Ebadi S, Bernien H, Greiner M, Vuletić V and Pichler H 2019 Parallel implementation of high-fidelity multiqubit gates with neutral atoms *Phys. Rev. Lett.* **123** 170503

[50]  Keesling A *et al* 2019 Quantum Kibble–Zurek mechanism and critical dynamics on a programmable Rydberg simulator *Nature* **568** 207

[51]  Pichler H, Wang S-T, Zhou L, Choi S and Lukin M D 2018 Quantum optimization for maximum independent set using Rydberg atom arrays (arXiv:1808.10816)

[52]  This definition, while useful in our analysis, is imprecise in general. A more accurate formulation of ordered phases is through the construction of an order parameter that takes on a certain range of values in each phase, with sharp changes at the boundary. While the order parameter unambiguously defines the ordered phase for any system size, the closeness-to-a product-state property does not hold for large systems because of quantum fluctuations. As each qubit in the chain has a small error, the total error from a product state scales exponentially with system size, and has been observed in experiment by [48]. We study this example as a simple system—small enough to be classically simulated—with a physically meaningful and important symmetry.

[53]  Samajdar R, Ho W W, Pichler H, Lukin M D and Sachdev S 2020 Complex density wave orders and quantum phase transitions in a model of square-lattice Rydberg atom arrays *Phys. Rev. Lett.* **124** 103601

[54]  2023 Bloqade.jl: Package for the quantum computation and quantum simulation based on the neutral-atom architecture (https://github.com/QuEraComputing/Bloqade.jl)

[55]  Our algorithm found these symmetries and optimized to losses of 0.062 for $\mathbb{Z}_2$, 0.0428 for $\mathbb{Z}_3$, and 1.416 for disordered, over 2000 iterations. Further details are in the appendix.

[56] Patti T L, Shehab O, Najafi K and Yelin S F 2022 Markov chain Monte Carlo enhanced variational quantum algorithms *Quantum Sci. Technol.* **8** 015019
[57] Bittel L and Kliesch M 2021 Training variational quantum algorithms is NP-hard *Phys. Rev. Lett.* **127** 120502
[58] Aaronson S 2019 Shadow tomography of quantum states *SIAM J. Comput.* **49** STOC18
[59] Semeghini G *et al* 2021 Probing topological spin liquids on a programmable quantum simulator *Science* **374** 1242
[60] Lu J Z, Bravo R A, Hou K, Dagnew G, Yelin S and Najafi K HQSLN (available at: https://github.com/jz-lu/symmetry-learning)