ARTICLE OPEN

Check for updates

Theoretical guarantees for permutation-equivariant quantum neural networks

Louis Schatzki[1,2,6]✉, Martín Larocca[3,4,6], Quynh T. Nguyen[3,5], Frédéric Sauvage[3] and M. Cerezo[1]✉

Despite the great promise of quantum machine learning models, there are several challenges one must overcome before unlocking their full potential. For instance, models based on quantum neural networks (QNNs) can suffer from excessive local minima and barren plateaus in their training landscapes. Recently, the nascent field of geometric quantum machine learning (GQML) has emerged as a potential solution to some of those issues. The key insight of GQML is that one should design architectures, such as equivariant QNNs, encoding the symmetries of the problem at hand. Here, we focus on problems with permutation symmetry (i.e., symmetry group S_n), and show how to build S_n-equivariant QNNs We provide an analytical study of their performance, proving that they do not suffer from barren plateaus, quickly reach overparametrization, and generalize well from small amounts of data. To verify our results, we perform numerical simulations for a graph state classification task. Our work provides theoretical guarantees for equivariant QNNs, thus indicating the power and potential of GQML.

npj Quantum Information (2024)10:12 ; https://doi.org/10.1038/s41534-024-00804-1

INTRODUCTION

Symmetry studies and formalizes the invariance of objects under some set of operations. A wealth of theory has gone into describing symmetries as mathematical entities through the concept of groups and representations. While the analysis of symmetries in nature has greatly improved our understanding of the laws of physics, the study of symmetries in data has just recently gained momentum within the framework of learning theory. In the past few years, classical machine learning practitioners realized that models tend to perform better when constrained to respect the underlying symmetries of the data. This has led to the blossoming field of geometric deep learning[1–5], where symmetries are incorporated as geometric priors into the learning architectures, improving trainability and generalization performance[6–13].

The tremendous success of geometric deep learning has recently inspired researchers to import these ideas to the realm of quantum machine learning (QML)[14–16]. QML is a new and exciting field at the intersection of classical machine learning, and quantum computing. By running routines in quantum hardware, and thus exploiting the exponentially large dimension of the Hilbert space, the hope is that QML algorithms can outperform their classical counterparts when learning from data[17].

The infusion of ideas from geometric deep learning to QML has been termed 'geometric quantum machine learning' (GQML)[18–24]. GQML leverages the machinery of group and representation theory[25] to build quantum architectures that encode symmetry information about the problem at hand. For instance, when the model is parametrized through a quantum neural network (QNN)[16,26–28], GQML indicates that the layers of the QNN should be equivariant under the action of the symmetry group associated to the dataset. That is, applying a symmetry transformation on the input to the QNN layers should be the same as applying it to its output.

One of the main goals of GQML is to create architectures that solve, or at least significantly mitigate, some of the known issues of standard symmetry non-preserving QML models[16]. For instance, it has been shown that the optimization landscapes of generic QNNs can exhibit a large number of local minima[29–32], or be prone to the barren plateau phenomenon[33–45] whereby the loss function gradients vanish exponentially with the problem size. Crucially, it is known that barren plateaus and excessive local minima are connected to the expressibility[30,32,37,43,46] of the QNN, so that problem-agnostic architectures are more likely to exhibit trainability issues. In this sense, it is expected that following the GQML program of baking symmetry directly into the algorithm, will lead to models with sharp inductive biases that suitably limit their expressibility and search space.

In this work, we leverage the GQML toolbox to create models that are permutation invariant, i.e., models whose outputs remain invariant under the action of the symmetric group S_n (see Fig. 1). We focus on this particular symmetry as learning problems with permutation symmetries abound. Examples include learning over sets of elements[47,48], modeling relations between pairs (graphs)[49–54] or multiplets (hypergraphs) of entities[55–57], problems defined on grids (such as condensed matter systems)[58–61], molecular systems[62–64], evaluating genuine multipartite entanglement[65–68], or working with distributed quantum sensors[69–71].

Our first contribution is to provide guidelines to build unitary S_n-equivariant QNNs. We then derive rigorous theoretical guarantees for these architectures in terms of their trainability and generalization capabilities. Specifically, we prove that S_n-equivariant QNNs do not lead to barren plateaus, can be overparametrized with polynomially deep circuits, and generalize well with only a polynomial number of training points. We also identify problems (i.e., datasets) for which the model is trainable, but also datasets leading to untrainability. All these appealing properties are also demonstrated in numerical simulations of a graph classification

[1]Information Sciences, Los Alamos National Laboratory, Los Alamos, NM 87545, USA. [2]Electrical and Computer Engineering, University of Illinois Urbana-Champaign, Urbana, IL 61801, USA. [3]Theoretical Division, Los Alamos National Laboratory, Los Alamos, NM 87545, USA. [4]Center for Nonlinear Studies, Los Alamos National Laboratory, Los Alamos, NM 87545, USA. [5]Harvard Quantum Initiative, Harvard University, Cambridge, ML 02138, USA. [6]These authors contributed equally: Louis Schatzki, Martín Larocca.
✉email: louisms2@illinois.edu; cerezo@lanl.gov

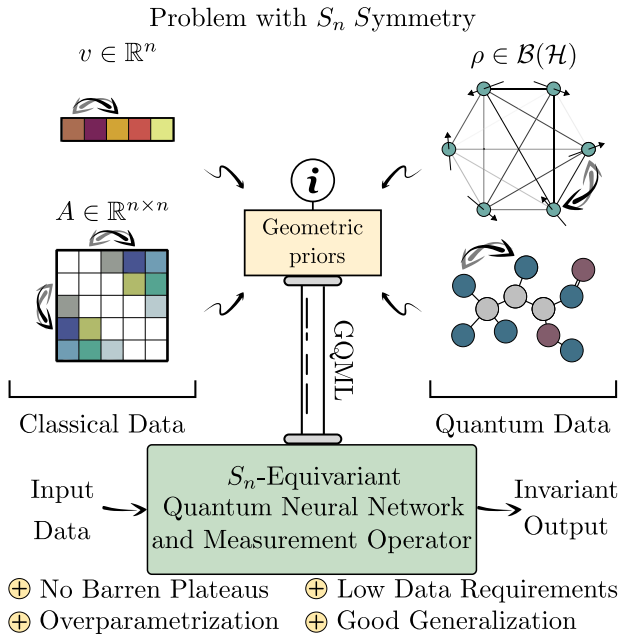


Fig. 1 GQML embeds geometric priors into a QML model. Incorporating prior knowledge through S_n-equivariance heavily restricts the search space of the model. We show that such inductive biases lead to models that do not exhibit barren plateaus, can be efficiently overparametrized, and require small amounts of data to generalize well.

task. Our empirical results verify our theoretical ones, and even show that the performance of S_n-equivariant QNNs can, in practice, be better than that guaranteed by our theorems.

RESULTS

Preliminaries

While the formalism of GQML can be readily applied to a wide range of tasks with S_n symmetry, here we will focus on supervised learning problems. We note, however that our results can be readily extended to more general scenarios such as unsupervised learning[72,73], reinforced learning[74,75], generative modeling[76–79], or to the more task-oriented computational paradigm of variational quantum algorithms[63,80].

Generally, a supervised quantum machine learning task can be phrased in terms of a data space $\mathcal{R}$ -a set of quantum states on some Hilbert space $\mathcal{H}$- and a real-valued label space $\mathcal{Y}$. We will assume $\mathcal{H}$ to be a tensor product of n two-dimensional subsystems (qubits) and thus of dimension $d = 2^n$. We are given repeated access to a training dataset $\mathcal{S} = \{(\rho_i, y_i)\}_{i=1}^M$, where ρ_i is sampled from $\mathcal{R}$ according to some probability P, and where $y_i \in \mathcal{Y}$. We further assume that the labels are assigned by some underlying (but unknown) function $f : \mathcal{R} \mapsto \mathcal{Y}$, that is, $y_i = f(\rho_i)$. We make no assumptions regarding the origins of ρ_i, meaning that these can correspond to classical data embedded in quantum states[81,82], or to quantum data obtained from some quantum mechanical process[60,61,83].

The goal is to produce a parametrized function $h_{\boldsymbol{\theta}} : \mathcal{R} \mapsto \mathcal{Y}$ closely modeling the outputs of the unknown target f, where $\boldsymbol{\theta}$ are trainable parameters. That is, we want $h_{\boldsymbol{\theta}}$ to accurately predict labels for the data in the training set $\mathcal{S}$ (low training error), as well as to predict the labels for new and previously unseen states (small generalization error). We will focus on QML models that are parametrized through a QNN, a unitary channel $\mathcal{U}_{\boldsymbol{\theta}} : \mathcal{B}(\mathcal{H}) \to \mathcal{B}(\mathcal{H})$ such that $\mathcal{U}_{\boldsymbol{\theta}}(\rho) = U(\boldsymbol{\theta})\rho U(\boldsymbol{\theta})^\dagger$. Here, $\mathcal{B}(\mathcal{H})$ denotes the space of bounded linear operators in $\mathcal{H}$. Throughout this work we will restrict to L-layered QNNs

$$\mathcal{U}_{\boldsymbol{\theta}} = \mathcal{U}^L_{\theta_L} \circ \cdots \circ \mathcal{U}^1_{\theta_1}, \quad \text{where} \quad \mathcal{U}^l_{\theta_l}(\rho) = \mathrm{e}^{-\mathrm{i}\theta_l H_l} \rho \mathrm{e}^{\mathrm{i}\theta_l H_l}, \tag{1}$$

for some Hermitian generators $\{H_l\}$, so that $U(\boldsymbol{\theta}) = \prod_{l=1}^L \mathrm{e}^{-\mathrm{i}\theta_l H_l}$. Moreover, we consider models that depend on a loss function of the form

$$\ell_{\boldsymbol{\theta}}(\rho_i) = \mathrm{Tr}[\mathcal{U}_{\boldsymbol{\theta}}(\rho_i) O], \tag{2}$$

where O is a Hermitian observable. We quantify the training error via the so-called empirical loss, or training error, which is defined as

$$\widehat{\mathcal{L}}(\boldsymbol{\theta}) = \sum_{i=1}^M c_i \ell_{\boldsymbol{\theta}}(\rho_i). \tag{3}$$

The model is trained by solving the optimization task $\arg\min_{\boldsymbol{\theta}} \widehat{\mathcal{L}}(\boldsymbol{\theta})$[63]. Once a desired convergence in the optimization is achieved, the optimal parameters, along with the loss function $\ell_{\boldsymbol{\theta}}$, are used to predict labels. For the case of binary classification, where $\mathcal{Y} = \{+1, -1\}$, one can choose $c_i := -\frac{y_i}{M}$. Then, if the measurement operator is normalized such that $\ell_{\boldsymbol{\theta}}(\rho_i) \in [-1, 1]$, this corresponds to the hinge loss, a standard loss function but not the only relevant one[84]) in machine learning.

We further remark that while Eq. (3) approximates the error of the learned model, the true loss is defined as

$$\mathcal{L}(\boldsymbol{\theta}) = \mathbb{E}_{\rho \sim P}[c(y)\ell_{\boldsymbol{\theta}}(\rho)]. \tag{4}$$

Here, we have denoted the weights as $c(y)$ to make their dependency on the labels y explicit. The difference between the true loss and the empirical one, known as the generalization error, is given by

$$\mathrm{gen}(\boldsymbol{\theta}) = |\mathcal{L}(\boldsymbol{\theta}) - \hat{\mathcal{L}}(\boldsymbol{\theta})|. \tag{5}$$

We now turn to GQML, where the first step is identifying the underlying symmetries of the dataset, as this allows us to create suitable inductive biases for $h_{\boldsymbol{\theta}}$. In particular, many problems of interest exhibit so-called label symmetry, i.e., the function f produces labels that remain invariant under a set of operations on the inputs. Concretely, one can verify that such set of operations forms a group[18], which leads to the following definition.

Definition 1. (Label symmetries and G-invariance). Given a compact group G and some unitary representation R acting on quantum states ρ, we say f has a label symmetry if it is G-invariant, i.e., if

$$f(R(g)\rho R(g)^\dagger) = f(\rho), \ \forall g \in G. \tag{6}$$

Here, we recall that a representation is a mapping of a group into the space of invertible linear operators on some vector space (in this case the space of quantum states) that preserves the structure of the group[25]. Also, we note that some problems may have functions f whose outputs change (rather than being invariant) in a way entirely determined by the action of G on their inputs. While still captured by general GQML theory, these do not pertain to Definition 1 and are not discussed further. Label invariance captures the scenario where the relevant information in ρ is unchanged under the action of G.

Evidently, when searching for models $h_{\boldsymbol{\theta}}$ that accurately predict outputs of f, it is natural to restrict our search to the space of models that respect the label symmetries of f. In this context, the theory of GQML provides a constructive approach to create G-invariant models, resting on the concept of equivariance[23].

Definition 2. (Equivariance). We say that an observable O is G-equivariant iff for all elements $g \in G$, $[O, R(g)] = 0$. We say that a

layer $\mathcal{U}^l_{\theta_l}$ of a QNN is G-equivariant iff it is generated by a G-equivariant Hermitian operator.

By the previous definition, G-equivariant layers are maps that commute with the action of the group

$$\mathcal{U}^l_{\theta_l}(R(g)\rho R(g)^\dagger) = R(g)\mathcal{U}^l_{\theta_l}(\rho)R(g)^\dagger. \tag{7}$$

Definition 2 can be naturally extended to QNNs.

Definition 3. (Equivariant QNN). We say that a L-layered QNN is G-equivariant iff each of its layers is G-equivariant.

Altogether, equivariant QNNs and measurement operators provide a recipe to design invariant models, i.e., models that respect the label symmetries. Akin to their classical machine learning counterparts[1–5], such GQML models consist in a composition of many equivariant operations (realized by the L layers of the equivariant QNN) and an invariant one (realized by the measurement of the equivariant observable)[23]. Furthermore, model invariance extends to the loss function itself, as captured by the following Lemma.

Lemma 1. (Invariance from equivariance). A loss function of the form in Eq. (2) is G-invariant if its composed of a G-equivariant QNN and measurement.

A proof of this Lemma along with that of the following Lemmas and Theorems are presented in Supplementary Methods 2 and 3.

S_n-Equivariant QNNs and measurements

In the previous section we have described how to build generic G-invariant models. We now specialize to the case where G is the symmetric group S_n, and where R is the qubit-defining representation of S_n, i.e., the one permuting qubits which for any $\pi \in S_n$ acts as

$$R(\pi)\bigotimes_{i=1}^n |\psi_i\rangle = \bigotimes_{i=1}^n \left|\psi_{\pi^{-1}(i)}\right\rangle. \tag{8}$$

Following Definitions 2 and 3, the first step towards building S_n-equivariant QNNs is defining S_n-equivariant generators for each layer. In the Methods section we describe how such operators can be obtained, but here we will restrict our attention to the following set of generators

$$\mathcal{G} = \left\{\frac{1}{n}\sum_{j=1}^n X_j, \frac{1}{n}\sum_{j=1}^n Y_j, \frac{2}{n(n-1)}\sum_{k<j} Z_jZ_k\right\}. \tag{9}$$

Note that there is some freedom in the choice of generators. Any two sums over two distinct single qubit Pauli operators (the first two generators) plus a sum over pairs of the remaining Pauli operator (the third generator) suffices and we choose the above set without loss of generality. In Fig. 2 we show an example of an $L = 3$ layered S_n-equivariant QNN acting on $n = 4$ qubits. While the single-qubit rotations generated by $\mathcal{G}$ are readily achievable in most quantum computing platforms, the collective ZZ interactions are best suited to architectures allowing for reconfigurable connectivity[85–87] or platforms that implement mediated all-to-all interactions[88,89]. In fact, such interactions are referred to as one-axis twisting[90] in the context of spin squeezing[91] and form the basis of many quantum sensing protocols.

In addition, we will consider observables of the following form

$$\mathcal{M} = \left\{\frac{1}{n}\sum_{j=1}^n X_j, \frac{2}{n(n-1)}\sum_{k<j;j=1}^n X_jX_k, \prod_{j=1}^n X_j\right\}, \tag{10}$$

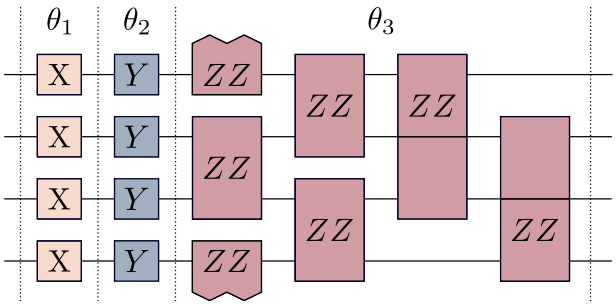


Fig. 2 Quantum circuit for an S_n-equivariant QNN. Each layer of the QNN is obtained by exponentiation of a generator from the set $\mathcal{G}$ in Eq. (9). Here we show a circuit with $L = 3$ layers acting on $n = 4$ qubits. Single-qubit blocks indicate a rotation about the x or y axis, while two-qubit blocks denote entangling gates generated by a ZZ interaction. All colored gates between dashed horizontal lines share the same trainable parameter θ_l.

where χ is a (fixed) Pauli matrix. It is straightforward to see that any $H_l \in \mathcal{G}$ and $O \in \mathcal{M}$ will commute with $R(\pi)$ for any $\pi \in S_n$. We note that one could certainly consider other observables as well.

We now leverage tools from representation theory to understand and unravel the underlying structure of S_n-equivariant QNNs and measurement operators. The previous will allow us to derive, in the next section, theoretical guarantees for these GQML models.

One of the most notable results from representation theory is that a given finite dimensional representation of a group decomposes into an orthogonal direct sum of fundamental building-blocks known as irreducible representations (irreps). As further explained in the Methods, the qubit-defining representation takes, under some appropriate global change of basis (which we denote with $\cong$), the block-diagonal form

$$R(\pi \in S_n) \cong \bigoplus_\lambda \bigoplus_{\mu=1}^{d_\lambda} r_\lambda(\pi) = \bigoplus_\lambda r_\lambda(\pi)\otimes \mathbb{1}_{d_\lambda}. \tag{11}$$

Here λ labels the irreps of S_n and r_λ is the corresponding irrep itself, which appears d_λ times. The collection of these repeated irreps is called an isotypic component. Crucially, the only irreps appearing in R correspond to two-row Young diagrams (see Methods) and can be parametrized by a single non-negative integer m, as $\lambda \equiv \lambda(m) = (n-m, m)$, where $m = 0, 1, \ldots, \lfloor \frac{n}{2} \rfloor$. It can be shown that

$$\begin{aligned} d_\lambda &= n - 2m + 1, \quad \text{and} \\ m_\lambda &= \tfrac{n!(n-2m+1)!}{(n-m+1)!m!(n-2m)!} \end{aligned} \tag{12}$$

where again d_λ is the number of times the irrep appears and m_λ is the dimension of the irrep itself. Note that every d_λ is in $\mathcal{O}(n)$, whereas some m_λ can grow exponentially with the number of qubits. For instance, if n is even and $m = n/2$, one finds that $m_\lambda = \Omega(4^n/n^2)$. We finally note that Eq. (11) implies $\sum_\lambda m_\lambda d_\lambda = 2^n$.

Given the block-diagonal structure of R, S_n-equivariant unitaries and measurements must necessarily take the form

$$U(\boldsymbol{\theta}) \cong \bigoplus_\lambda \mathbb{1}_{m_\lambda}\otimes U_\lambda(\boldsymbol{\theta}), \quad \text{and} \quad O \cong \bigoplus_\lambda \mathbb{1}_{m_\lambda}\otimes O_\lambda. \tag{13}$$

That is, both $U(\boldsymbol{\theta})$ and O decompose into a direct sum of d_λ-dimensional blocks repeated m_λ times (with m_λ called the multiplicity) on each isotypic component λ. This decomposition is illustrated in Fig. 3.

Let us highlight several crucial implications of the block diagonal structure arising from S_n-equivariance. First and foremost, we note that, under the action of an S_n-equivariant QNN, the

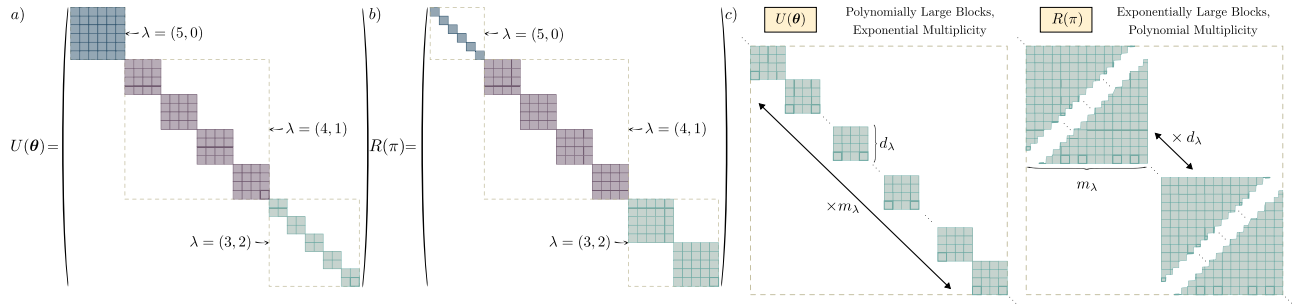


Fig. 3 Representation theory and S_n-equivariance. Using tools from representation theory we find that the S_n-equivariant QNN $U(\boldsymbol{\theta})$ and the representation of the group elements $R(\pi)$ -for any $\pi \in S_n$- admit an irrep block decomposition as in Eq. (13) and Eq. (11), respectively. The irreps can be labeled with a single parameter $\lambda = (n-m, m)$ where $m = 0, 1, \ldots, \lfloor \frac{n}{2} \rfloor$. For a system of $n = 5$ qubits, we show in **a**) the block diagonal decomposition for $U(\boldsymbol{\theta})$ and in **b**) the decomposition of $R(\pi)$ as a representation of S_5. The dashed boxes denote the isotypic components labeled by λ. **c** As n increases, $U(\boldsymbol{\theta})$ has a block diagonal decomposition which contains polynomially large blocks repeated a (potentially) exponential number of times. In contrast, the block decomposition of $R(\pi)$ (for any $\pi \in S_n$) contains blocks that can be exponentially large but that are only repeated a polynomial number of times.

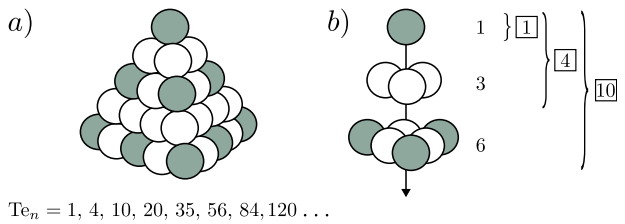


Fig. 4 Tetrahedral numbers. a The Tetrahedral numbers Te_n are obtained by counting how many spheres can be stacked in the configuration of a tetrahedron (triangular base pyramid) of height n. **b** One can also compute Te_n as the sum of consecutive triangular numbers, which count how many objects (e.g., spheres) can be arranged in an equilateral triangle.

Hilbert space decomposes as

$$\mathcal{H} \cong \bigoplus_{\lambda} \bigoplus_{v=1}^{m_\lambda} \mathcal{H}_\lambda^v, \tag{14}$$

where each $\mathcal{H}_\lambda^v$ denotes a d_λ-dimensional invariant subspace. Moreover, one can also see that when the QNN acts on an input quantum state as $\mathcal{U}_{\boldsymbol{\theta}}(\rho) = U(\boldsymbol{\theta})\rho U(\boldsymbol{\theta}^\dagger)$, it can only access the information in ρ which is contained in the invariant subspaces $\mathcal{H}_\lambda^v$ (see also ref. [23]). This means that to solve the learning task, we require two ingredients: i) the data must encode the relevant information required for classification into these subspaces[23,25], and ii) the QNN must be able to accurately process the information within each $\mathcal{H}_\lambda^v$. As discussed in the Methods, we can guarantee that the second condition will not be an issue, as the set of generators in Eq. (9) is universal within each invariant subspace, i.e., the QNN can map any state in $\mathcal{H}_\lambda^v$ to any other state in $\mathcal{H}_\lambda^v$ (see also ref. [92]).

A second fundamental implication of Eq. (13) is that the manifold of equivariant unitaries is of low dimension. We make this explicit in the following lemma.

Lemma 2. (Dimension of S_n-equivariant unitaries). The submanifold of S_n-equivariant unitaries is of dimension equal to the Tetrahedral numbers $\mathrm{Te}_{n+1} = (0.0ptn + 33)$ (see Fig. 4), and therefore on the order of $\Theta(n^3)$.

Crucially, Lemma 2 shows that the equivariance constraint limits the degrees of freedom in the QNN (and concomitantly in any observable) from 4^n to only polynomially many.

Absence of barren plateaus in S_n-equivariant QNNs

Barren plateaus have been recognized as one of the main challenges to overcome in order to guarantee the success of QML models using QNNs[16]. When a model exhibits a barren plateau, the loss landscape becomes, on average, exponentially flat and featureless as the problem size increases[33–45]. This severely impedes its trainability, as one needs to spend an exponentially large amount of resources to correctly estimate a loss-minimizing direction. Issues of barren plateaus arise primarily due to the structure of the models (including the choice of QNN, the input state and the observables) employed[33–43,45] but can also be caused solely by effects of noise[44]. In the rest of this section, we will only be concerned with the former type of barren plateaus, that is the most studied.

Recently, a great deal of effort has been put forward towards creating strategies capable of mitigating the effect of barren plateaus[78,93–105]. While these are promising and have shown moderate success, the 'holy grail' is identifying architectures which are immune to barren plateaus altogether, and thus enjoy trainability guarantees. Examples of such architectures are shallow hardware efficient ansatzes[34], quantum convolutional neural networks[106], or the transverse field Ising model Hamiltonian variational ansatz[43,45]. Here, we prove that another architecture can be added to this list: S_n-equivariant QNNs.

When studying barren plateaus, one typically analyzes the variance of the empirical loss function partial derivatives, $\partial_\mu \hat{\mathcal{L}}(\boldsymbol{\theta}) = \partial \hat{\mathcal{L}}(\boldsymbol{\theta})/\partial\theta_\mu$, where $\theta_\mu \in \boldsymbol{\theta}$. We say that there is a barren plateau in the θ_μ direction if $\mathbb{E}_{\boldsymbol{\theta}}[\partial_\mu \hat{\mathcal{L}}(\boldsymbol{\theta})] = 0$ and $\mathrm{Var}_{\boldsymbol{\theta}}[\partial_\mu \hat{\mathcal{L}}(\boldsymbol{\theta})]$ is exponentially vanishing.

Before stating our main results, we introduce a bit of notation. Let us define Q_λ^v to be the operator that maps vectors from $\mathcal{H}$ to $\mathcal{H}_\lambda^v$, such that $(Q_\lambda^v)^\dagger Q_\lambda^v$ realizes a projection onto $\mathcal{H}_\lambda^v$ (see Supplementary Methods 4 for additional details). Given a matrix $B \in \mathbb{C}^{d\times d}$, we will denote its restriction to $\mathcal{H}_\lambda^v$ as

$$B_\lambda^v = Q_\lambda^v B (Q_\lambda^v)^\dagger, \tag{15}$$

with $B_\lambda^v \in \mathbb{C}^{d_\lambda \times d_\lambda}$. We remark that the restriction of S_n-equivariant generators is independent of the v multiplicity index (see Eq. (13)). On the other hand, the restriction of non-equivariant operators (such as the input states ρ_1) are not independent of v, meaning that that the set composed of all the restrictions ρ_λ^v contain an exponentially large amount of non-redundant information that the QNN can act on (see also ref. [23]).

Denoting the weighted average of the input states as $\sigma = \sum_{i=1}^M c_i \rho_i$, we find:

Theorem 1. (Variance of partial derivatives). Let $\mathcal{U}_{\boldsymbol{\theta}}$ be an S_n-equivariant QNN, with generators in $\mathcal{G}$, and O an S_n-equivariant

measurement operator from $\mathcal{M}$. Consider an empirical loss $\hat{\mathcal{L}}(\boldsymbol{\theta})$ as in Eq. (3). Assuming a circuit depth L such that the QNN forms independent 2-designs on each isotypic block, we have $\langle \partial_\mu \hat{\mathcal{L}}(\boldsymbol{\theta}) \rangle_{\boldsymbol{\theta}} = 0$, and

$$\mathrm{Var}_{\boldsymbol{\theta}}[\partial_\mu \hat{\mathcal{L}}(\boldsymbol{\theta})] = \sum_\lambda \frac{2d_\lambda}{(d_\lambda^2-1)^2} \Delta(H_{\mu,\lambda}) \Delta(O_\lambda) \Delta\left(\sum_{\nu=1}^{m_\lambda} \sigma_\lambda^\nu\right). \tag{16}$$

Here, $\Delta(B) = \mathrm{Tr}[B^2] - \frac{\mathrm{Tr}[B]^2}{\dim(B)}$.

In the "Methods", we present a sketch of the proof for Theorem 1, as well as its underlying assumptions.

We remark that while we have derived Theorem 1 for S_n-equivariant QNNs and measurement operators, given some general finite-dimensional compact group G, the form of Eq. (16) is valid provided that one uses a G-equivariant QNN that is universal with each invariant subspace. In this case, the summation over λ will run over the irreps of the representation of G.

Let us now analyze each term in Eq. (16) to identify potential sources of untrainability. First, let us consider the prefactors $\frac{2d_\lambda}{(d_\lambda^2-1)^2}$. From Eq. (12) we can readily see that $\frac{2d_\lambda}{(d_\lambda^2-1)^2} \in \Omega(\frac{1}{n^3})$ for any λ. Next, it is convenient to separate the two remaining potential sources of barren plateaus into two categories: i) those that are QNN or measurement dependent, $\Delta(H_{\mu,\lambda})$ and $\Delta(O_\lambda)$, and ii) those that are dataset-dependent, $\Delta(\sum_\nu \sigma_\lambda^\nu)$. This identification commonly appears when analyzing the absence of barren plateaus (see refs. [34,42,43,106,107]) and allows one to study how the architecture and dataset individually affect the trainability. In what follows, we will say that some architecture does not induce barren plateaus if the terms that are QNN or measurement dependent are not exponentially vanishing.

Using tools from representation theory we can obtain the following exact expressions for S_n-equivariant operators.

Theorem 2. Let A be a S_n-equivariant operator.

$$\begin{cases} \text{If } A = \sum_{j=1}^n \chi_j, & \text{then } \Delta(A_\lambda) = 2\binom{d_\lambda+1}{3}, \\ \text{If } A = \sum_{k<j} \chi_j \chi, & \text{then } \Delta(A_\lambda) = \frac{8}{3}\binom{d_\lambda+2}{5}, \\ \text{If } A = \prod_{j=1}^n \chi_j, & \text{then } \Delta(A_\lambda) = \frac{d_\lambda^2-1+n \bmod 2}{d_\lambda}, \end{cases} \tag{17}$$

where $\chi \in \{X, Y, Z\}$.

In Supplementary Methods 6, we also derive formulas for the case of A being k-body operators.

Let us review the implications of Theorem 2. First, note that all elements of our gate-set $\mathcal{G}$ and measurement-set $\mathcal{M}$ are of the form in Theorem 2, and therefore belong in $\Omega(d_\lambda)$. This follows from the fact that the binomial coefficient $\binom{0.0ptn+ab}{}$ scales as a polynomial of degree b in n. Since d_λ itself is in $\Theta(n)$ (see Eq. (12)), for all λ and μ

$$\Delta(O_\lambda) \text{ and } \Delta(H_{\mu,\lambda}) \in \Omega(n). \tag{18}$$

Hence, combining this result with Theorem 1 allows us to argue that S_n-equivariant QNNs do not induce barren plateaus.

Corollary 1. Under the same assumptions as Theorem 1, it follows that, if $\Delta(\sum_{\nu=1}^{m_\lambda} \sigma_\lambda^\nu) \in \Omega(1/\mathrm{poly}(n))$, then the empirical loss function satisfies

$$\mathrm{Var}_{\boldsymbol{\theta}}[\partial_\mu \hat{\mathcal{L}}] \in \Omega\left(\frac{1}{\mathrm{poly}(n)}\right). \tag{19}$$

We note that a crucial requirement for Corollary 1 to hold is that $\Delta(\sum_\nu \sigma_\lambda^\nu)$ needs to be, at most, polynomially vanishing. In Sec., we identify cases of datasets leading to trainability but also to untrainability. Finally, we note that as discussed in Supplementary Methods 9, Corollary 1 is sufficient to guarantee that the loss function does not exhibit the narrow gorge phenomenon, whereby the minima of the loss occupy an exponentially small volume of parameter space[108]. In other words, we show that absence of barren plateau implies absence of narrow gorges and loss function anti-concentration.

Efficient overparametrization

Absence of barren plateaus is a necessary, but not sufficient, condition for trainability, as there could be other issues compromising the parameter optimization. In particular, it has been shown that quantum landscapes can exhibit a large number of local minima[29–31]. As such, here we consider a different aspect of the trainability of S_n-equivariant QNNs: their ability to converge to global minima. For this purpose, we find it convenient to recall the concept of overparametrization.

Overparametrization denotes a regime in machine learning where models have a capacity much larger than that necessary to represent the distribution of the training data. For example, when the number of parameters is greater than the number of training points. Models operating in the overparametrized regime have seen tremendous success in classical deep learning, as they closely fit the training data but still generalize well when presented with new data instances[109–112]. Recently, ref. [32] studied overparametrization in the context of QML models. A clear phase transition in the trainability of under- and overparametrized QNNs was evidenced: Below some critical number of parameters (underparametrized) the optimizer greatly struggles to minimize the loss function, whereas beyond that number of parameters (overparametrized) it converges exponentially fast to solutions (see Methods for further details).

Given the desirable features of overparametrization, it is important to estimate how many parameters are needed to achieve this regime. Here, we can derive the following theorem.

Theorem 3. Let $\mathcal{U}_{\boldsymbol{\theta}}$ be a S_n-equivariant QNN with generators in $\mathcal{G}$. Then, $\mathcal{U}_{\boldsymbol{\theta}}$ can be overparametrized with $\mathcal{O}(n^3)$ parameters.

Theorem 3 guarantees that S_n-equivariant QNNs only require a polynomial number of parameters to reach overparametrization.

Generalization from few data points

Thus far, we have seen that S_n-equivariant QNNs can be efficiently trained, as they exhibit no barren plateaus and can be overparametrized. However, in QML we are not only interested in achieving a small training error, we also aim at low generalization error[26,61,113–116].

Computing the generalization error in Eq. (4) is usually not possible, as the probability distribution P over which the data is sampled is generally unknown. However, one can still derive bounds for gen($\boldsymbol{\theta}$) which guarantee a certain performance when the model sees new data. Here, we obtain an upper bound for the generalization error via the covering numbers (see Methods)[61,117], and prove that the following theorem holds.

Theorem 4. Consider a QML problem with loss function as described in Eq. (4). Suppose that an n-qubit S_n-equivariant QNN $\mathcal{U}(\boldsymbol{\theta})$ is trained on M samples to obtain some trained parameters $\boldsymbol{\theta}^*$. Then the following inequality holds with probability at least $1-\delta$

$$\text{gen}(\boldsymbol{\theta}^*) \leqslant \mathcal{O}\left(\sqrt{\frac{\text{Te}_{n+1}}{M}} + \sqrt{\frac{\log(1/\delta)}{M}}\right). \tag{20}$$

The crucial implication of Theorem 4 is that we can guarantee $\text{gen}(\boldsymbol{\theta}^*) \leqslant \epsilon$ with high probability, if $M \in \mathcal{O}\left(\frac{\text{Te}_{n+1}+\log(1/\delta)}{\epsilon^2}\right)$. For fixed δ and ϵ, this implies $M \in \mathcal{O}(n^3)$, i.e., we only need a polynomial number of training points. Also note that this results shows that minimizing the empirical loss closely minimizes the true loss with high probability. Say that $\hat{\mathcal{L}}^* = \inf_{\boldsymbol{\theta}} \hat{\mathcal{L}}(\boldsymbol{\theta})$ is the minimal empirical loss and $\mathcal{L}^* = \inf_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta})$ the minimal true loss. Then, with $M \in \mathcal{O}\left(\frac{\text{Te}_{n+1}+\log(1/\delta)}{\epsilon^2}\right)$ training data point the inequality $|\hat{\mathcal{L}}^* - \mathcal{L}^*| \leqslant \epsilon$ holds with probability at least $1-\delta$.

Lastly, we remark that Theorem 4 can be readily adapted to other GQML models. As shown in Methods, this theorem stems from the fact that the equivariant unitary submanifold, in its block-diagonal form in Eq. (13), can be covered[117] by ε-balls in a block-wise manner. In Supplementary Methods 8, we also show that the VC dimension[118] of equivariant QNNs (and also more general parameterized channels) can be upper bounded by the dimension of the commutant of the symmetry group, a fact which could be of independent interest.

Trainable states

As discussed in the previous section, S_n-equivariant QNNs and measurement operators cannot induce barren plateaus. Thus, the trainability of the model hinges on the behavior of $\Delta(\sum_\nu \sigma_\lambda^\nu)$. We note that this dataset-dependent trainability is not unique to S_n-equivariant QNNs, but is rather present in all absence of barren plateaus results (see refs. [34,42,43,106,107,119]) as there always exist datasets for which an otherwise trainable model can be rendered untrainable.

To understand the conditions that lead to an exponentially vanishing of $\Delta(\sum_\nu \sigma_\lambda^\nu)$ we note that for a Hermitian operator B, we have $\Delta(B) = D_{\text{HS}}\left(B, \frac{\text{Tr}[B]}{\dim(B)}\mathbb{1}\right)$, where $D_{\text{HS}}(A,B) = \|A-B\|_2^2$ is the Hilbert-Schmidt distance. Alternatively, we can interpret $\Delta(B)$ as the variance of the eigenvalues of B. From here, we can see that one will obtain trainability if at least one σ_λ is not exponentially close to a multiple of the identity in some subspace $\mathcal{H}_\lambda^\nu$.

In Table 1, we present examples of states for which $\Delta(\sum_\nu \sigma_\lambda^\nu)$ vanishes polynomially, leading to a trainable model, but also cases where the input state leads to exponentially vanishing $\Delta(\sum_\nu \sigma_\lambda^\nu)$ and thus to a barren plateau. While we leave the details of how each type of input state is generated for the Methods section, we note that the results in Table 1 demonstrate the critical role that the input states play in determining the trainability of a model (this will be further elucidated in numerical results below). Such insight is particularly important as one can create adversarial datasets yielding barren plateaus (see Supplementary Methods 10). Moreover, it indicates that care must be taken when encoding classical data into quantum states as the embedding scheme can induce trainability issues[42,119].

Numerical results

Here, we consider the task of classifying connected graph states from disconnected graph states, which are prepared as follows. First, we generate n-node random graphs from the Erdös-Rényi distribution[120], with an edge probability of 40%. The ensuing graphs are binned into two categories: connected and disconnected. We then embed these graphs into quantum graph states via the canonical scheme of[121,122] (see Methods section). We highlight that such encoding preserves symmetries in the input data, in the sense that a permutation of the underlying graph yields a permutation of the qubits constituting its graph state (i.e., of the form Eq. (8)). The previous allows us to create a dataset where half of the states encodes connected graphs (label $y_i = +1$), and the other half encodes disconnected graphs (label $y_i = -1$). To analyze the data, we use an S_n-equivariant QNN with generators in Eq. (9) (see also Fig. 2), and measure the operator $O = \frac{2}{n(n-1)}\sum_{k<j;j=1}^{n} X_j X_k$.

In the following, we characterize the trainability and generalization properties of S_n-equivariant QNNs for this classification task, but we note that further aspects of the problem are discussed in the Supplementary Note. These include analyzing the effect of the graph encoding scheme in the trainability, the irrep contributions to the gradient variance, and comparing S_n-equivariant QNNs against problem-agnostic ones. In particular, the latter shows that for the present graph classification task, problem-agnostic models are hard to train and tend to greatly overfit the data, i.e., they have large generalization errors despite performing well on the training data.

Table 1. Input pure states and their effect on the trainability of S_n-equivariant QNNs.

Input state	Trainable?	Method
Symmetric	Yes	Analytical
Fixed Hamming-weight encoding	Yes	Analytical
Local Haar random	Yes	Numerical
Fixed depth random circuit	Yes	Numerical
Disconnected graph state	Yes	Numerical
3-regular graph state	Yes	Numerical
$n/2$-regular graph state	Yes	Numerical
Global Haar Random	No	Analytical
Linear depth random circuit	No	Numerical
Erdös-Rényi random graph state	No	Numerical

Trainable means that $\Delta(\sum_\nu \sigma_\lambda^\nu) \in \Omega(1/\text{poly}(n))$, whereas untrainable means $\Delta(\sum_\nu \sigma_\lambda^\nu) \in \mathcal{O}(1/2^n)$. Analytical method indicates that we can exactly compute the scaling of $\Delta(\sum_\nu \sigma_\lambda^\nu)$, whereas numerical one means that we evaluate it numerically. The analytical proofs and details of the simulations can be found in Supplementary Methods 7. We note that, these results are obtained by computing the loss with a single data instance (i.e., for $M=1$ in Eq. (3)).

Numerics on barren plateaus

In Fig. 5a we show the variance of the cost function partial derivatives for a parameter θ_μ in the middle of the QNN. Each point is evaluated for a total of 50 random input states, and with 20 random sets of parameters $\boldsymbol{\theta}$ per input. We can see that when the variance is evaluated for states randomly drawn from the whole dataset—with an equal number of connected and disconnected graphs—then $\text{Var}_{\boldsymbol{\theta}}[\partial_\mu \hat{\mathcal{L}}]$ only decreases polynomially with the system size (as evidenced by the curved line in the log-linear scale), meaning that the model does not exhibit a barren plateau. We note that, as shown in Fig. 5a, when the input to the QNN is a disconnected graph state, then the variance vanishes polynomially, whereas if we input a connected graph state it vanishes exponentially. This illustrates a key fact of QML: when trained over a dataset, the data from different classes can contribute very differently to the model's trainability (see ref. [18] for

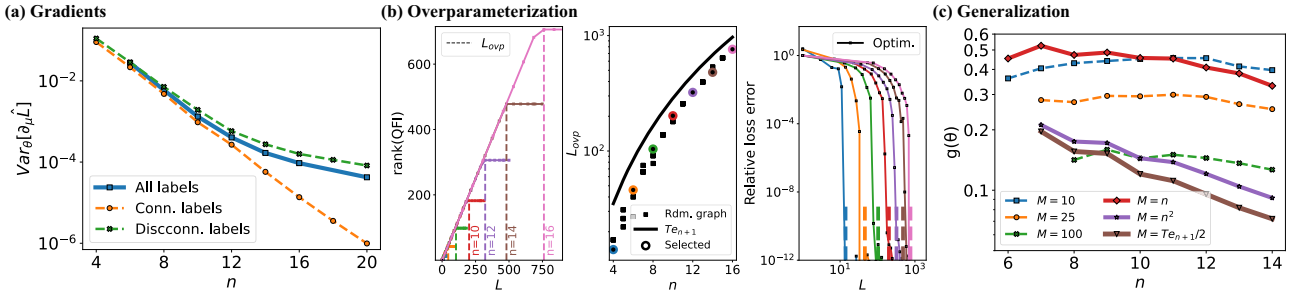


Fig. 5 Task of distinguishing connected from disconnect graphs with an S_n-equivariant QNN. a Variance of the loss function partial derivatives versus the number of qubits n (in log-linear scale). The square blue line depicts the variance for inputs of the QNN drawn from a dataset composed of connected and disconnected graph states. To visualize how the data with different labels contributes to this variance, we also plot in green crosses (orange circles) the variances when the QNN is only fed connected (disconnected) graph states. **b** In the left panel, we show representative results for the rank of the QFIM (defined in the main text) versus the number of layers L for different number of qubits n. The critical value of layers at which this rank saturates, denoted L_{ovp} (vertical dashed lines), corresponds to the onset of overparametrization. In the middle panel, we report the scaling of L_{ovp} versus the number of qubits (log-linear scale). For each problem size, we present results for ten random input graph states and, as a comparison, also report the Tetrahedral numbers Te_{n+1} (solid line). In the right panel, we report the relative loss error of optimized QNNs at given number of layers L (in log-linear scale). These are obtained for different system sizes, with the dashed vertical lines indicating the corresponding values of L_{ovp}. **c** Normalized generalization error versus number of qubits n (in log-linear scale) for different training dataset sizes M. Here, we consider an overparametrized QNN with $L = \text{Te}_{n+1}$.

a discussion on how this result enables new forms of classification).

Numerics on overparametrization

Following the results in ref. [32], let us analyze the overparametrization phenomenon by studying the rank of the quantum fisher information matrix (QFIM)[123,124], denoted $F(\boldsymbol{\theta})$ and whose entries are given by

$$[F(\boldsymbol{\theta})]_{jk} = 4\text{Re}[\langle \partial_j\psi(\boldsymbol{\theta})|\partial_k\psi(\boldsymbol{\theta})\rangle - \langle \partial_j\psi(\boldsymbol{\theta})|\psi(\boldsymbol{\theta})\rangle\langle\psi(\boldsymbol{\theta})|\partial_k\psi(\boldsymbol{\theta})\rangle],$$

with $|\psi(\boldsymbol{\theta})\rangle = U(\boldsymbol{\theta})|\psi\rangle$, and $|\partial_i\psi(\boldsymbol{\theta})\rangle = \partial|\psi(\boldsymbol{\theta})\rangle/\partial\theta_i = \partial_i|\psi(\boldsymbol{\theta})\rangle$ for $\theta_i \in \boldsymbol{\theta}$. The rank of the QFIM quantifies the number of potentially accessible directions in state space. In this sense, the model is overparametrized if the QFIM rank is saturated, i.e., if adding more parameters (or layers) to the QNN does not further increase the QFIM rank. When this occurs, one can access all possible directions in state space and efficiently reach the solution manifold[32,125,126]. On the other hand, the model is underparametrized if the QFIM rank is not maximal. In this case, there exists inaccessible directions in state space, leading to false local minima, that is, local minima that are not actual minima of the loss function.

In Fig. 5(b, left panel) we report representative results of the QFIM rank versus the number of layers L for problems with even numbers $n \in [4, 16]$ of qubits. These results correspond to random connected graphs and random values of $\boldsymbol{\theta}$. Here we can see that, for a given n, as the number of layers increases, the rank of the QFIM also increases until it reaches a saturation point. Once this critical number of layers (denoted as L_{ovp}) is reached, the model is considered to be overparametrized[32]. In Fig. 5(b, middle panel) we plot the scaling of L_{ovp} (for 10 random connected or disconnected graphs per system size) versus n, as well as the Tetrahedral numbers Te_{n+1}. As can be seen, in all cases, the overparametrization onset occurs for a number of layers $L_{\text{ovp}} < \text{Te}_{n+1}$, indicating efficient overparametrization.

To appreciate the practical effects of overparametrization, we report in Fig. 5(b, right panel) optimization performances of S_n-equivariant QNNs as a function of the number L of layers employed. All the optimizations are performed using the hinge loss function, with the L-BFGS-B optimization algorithm[127]. The system sizes are in $n \in [4, 16]$ qubits, and correspond to the graphs that were studied in the left panel and highlighted in the middle one. The relative loss error reported indicates how close an optimized QNN is from the best achievable model. Explicitly, it is defined as $|\hat{\mathcal{L}}_L - \hat{\mathcal{L}}_{\text{min}}|/|\hat{\mathcal{L}}_{\text{min}}|$, where $\hat{\mathcal{L}}_L$ is the loss achieved after optimization of a QNN with a given L, and where $\hat{\mathcal{L}}_{\text{min}}$ is the minimum loss achieved for any of the values L considered, i.e., $\hat{\mathcal{L}}_{\text{min}} = \arg\min_L \hat{\mathcal{L}}_L$ (we systematically verify that for sufficient large L all optimizations reliably converge to this same loss $\hat{\mathcal{L}}_{\text{min}}$). For every value of n studied, we see that for a small number of layers the optimizer struggles to significantly minimize the loss. However, as L increases, there exists a computational phase transition whereby the optimizer is able to easily identify optimal parameters and reach much smaller loss values. Notably, such computational phase transition occurs slightly before L_{ovp} (indicated by a dashed vertical line), meaning that even before the QFIM rank saturates, the model has sufficient directions to efficiently reach the solution manifold. Overall, we see that for number of layers growing at most polynomially with n, one can ensure convergence to solution of the model.

Numerics on generalization error

In Fig. 5c, we study the generalization error of an overparametrized S_n-equivariant QNN (with $L = \text{Te}_{n+1}$) for different training dataset sizes M and with respect to test sets of size $M_{test} = 2 \times \text{Te}_{n+1}$ that are independently drawn from the training ones. Generalization errors are evaluated for random QNNs parameters $\boldsymbol{\theta}$ and we report the 90-th percentile of the errors obtained, i.e., for $\delta = 90\%$ in Eq. (20). In the plot, we show the normalized generalization error $g(\boldsymbol{\theta}) = \frac{\text{gen}(\boldsymbol{\theta})}{\text{V ar}_{\boldsymbol{\theta},\rho}^{1/2}[\ell(\boldsymbol{\theta},\rho)]}$.

We stress that such normalization can only increase the generalization errors obtained, and is only used in order to compare generalization errors across different values of n without artifacts resulting from loss concentration effects as the system sizes grow. As seen in Fig. 5(c), when the size of the training set is constant, the generalization error is also approximately constant across problem sizes. However, when the training set size scales with n, the generalization error decreases with n, with this even occurring for $M = n$. Notably, if $M = \text{Te}_{n+1} \in \Theta(n^3)$, we can see that the generalization error significantly decreases with problem size. That is, for this problem, we found generalization errors to be better than the scaling of the bounds derived in Eq. (20).

DISCUSSION

GQML has recently been proposed as a framework for systematically creating models with sharp geometric priors arising from the symmetries of the task at hand[18–22]. Despite its great promise, this nascent field has only seen heuristic success as no true performance guarantees have been proved for its models. In this

work we provide the first theoretical guarantees for GQML models aimed at problems with permutation invariance. Our first contribution is the introduction of the S_n-equivariant QNN architecture. Using tools from representation theory, we rigorously find that these QNNs present salient features such as absence of barren plateaus (and narrow gorges), generalization from very few data points, and a capability of being efficiently overparametrized. All these favorable properties can be viewed as being direct consequences of the inductive biases embedded in the model, which greatly limits their expressibility[37,46,128]. Namely, these S_n-equivariant QNNs act only on the –polynomially large– multiplicity spaces of the qubit-defining representation of S_n. To complete our analysis, we performed numerical simulations for a graph classification task and heuristically found that the model's performance is even better than that predicted by our theoretical results.

Taken together, our results provide the first rigorous guarantees for equivariant QNNs, and demonstrate that GQML may be a powerful tool in the QML repertoire. We highlight that while we focus on problems with S_n symmetry, many of our proof techniques hold for general finite-dimensional compact groups. Hence, we hope that the representation-theory-based techniques used here can serve as blueprints to analyze the performance of other models. We envision that in the near future, GQML models with provable guarantees will be widely spread among the QML literature.

Finally, we note that while our results were derived in the absence of noise, it would be interesting to account for hardware imperfections. Clearly, the presence of noise would change our analysis, and most likely weaken our trainability guarantees. As such, while we can guarantee that S_n-equivariant QNNs will be useful on fault-tolerant quantum devices, we do not abandon hope that they can be used in the near-term era provided that noise levels are small enough.

Note added: In light of the recent preprint[129], we have added a detailed discussion in the Supplementary Note regarding the possibility of classically simulating S_n-equivariant QNNs. As we argue there, for most relevant cases in QML, the algorithm in[129] is not fully classical, as it require access to a quantum computer to obtain a "classical description" of the input data. Moreover, even if one is given such "classical description", the ensuing algorithm that replaces the use of a QNN scales extremely poorly with the number of qubits. Taken together these results indicate that if one has access to a quantum computer, it is not entirely obvious whether one should use it to obtain a classical description of the data followed by expensive post-processing, or if one should run the QNN on the quantum device and exploit its favorable properties like efficient overparametrization and absence of barren plateaus. We will save such comparison for future work.

Now, we will briefly compare S_n-equivariant QNNs to other barren-plateau-avoiding architectures.

First, let us consider the shallow hardware efficient ansatz (HEA)[34,130] and the quantum convolutional neural network (QCNN)[60,106]. While our goal is not to provide a comprehensive description of these models, we recall the three key properties leading to their trainability: locality of the gates, shallowness of the circuit, locality of the measurement operator. Both the HEA and QCNN are composed of parametrized gates acting in a brick-like fashion on alternating pairs of neighboring qubits (local gates), and are composed of only a few—logarithmically many—layers of such gates (shallowness of the circuit). The combination of these two factors leads to a low scrambling power and greatly limited expressibility of the QNN. Then, the final ingredient for their trainability requires measuring a local operators (i.e., an operator acting non-trivially on a small number of qubits). While this assumption is guaranteed for QCNNs—due to their feature-space reduction property—, the HEA can be shown to be untrainable for global measurement (i.e., operators acting non-trivially on all qubits). Here we can already see that S_n-equivariant QNNs do not share the properties leading to trainability in HEAs and QCNNs. To begin, we can see from the set of generators $\mathcal{G}$ in Eq. (9) that the S_n-equivariant architecture allows for all long-range interactions in each layer, breaking the locality of gates assumption. Moreover, and in stark contrast to HEAs, one can train the S_n-equivariant QNN even when measuring global observables (for instance, we allow for the $O = \prod_{j=1}^{n} X_j$ in Eq. (10)). Finally, we remark that HEAs and QCNNs cannot be efficiently overparametrized, as they require an exponentially large number of parameters to reach overparametrization[43]. On the other hand, according to Theorem 3 the S_n-equivariant QNN can be overparametrized with polynomially many layers.

Next, let us consider the transverse field Ising model Hamiltonian variational ansatz (TFIM-HVA)[43,45]. The mechanism leading to absence of barren plateaus in this architectures is more closely related to that of the S_n-equivariant model, although there are still some crucial differences. On the one hand, it can be shown that the TFIM-HVA has an extremely limited expressibility, having only a maximum number of free parameters in $\mathcal{O}(n^2)$, and being able to reach overparametrization with polynomially many layers. While this is similar to the case of S_n-equivariant architectures (see Lemma 2 and Theorem 3), the block diagonal structure of the TFIM-HVA is fundamentally different than that arising from S_n-equivariant: The TFIM-HVA unitary has four exponentially large blocks repeated a single time each, while S_n-equivariant unitaries have polynomially small blocks repeated exponentially many times. This subtle, albeit important, distinction makes it such that S_n-equivariant QNNs enjoy generalization guarantees (from Theorem 4) which are not directly applicable to TFIM-HVA architectures.

The previous shows that S_n-equivariant QNNs stand-out amid the other trainable architectures, exhibit many favorable properties that other models only partially enjoy.

Lastly, we now consider future directions and possible extensions of our work. We recall that Definition 3 requires *every* layer of the QNN to be equivariant. This is evidently not general, as one could have several consecutive layers which are not individually equivariant, but compose to an equivariant unitary for certain $\boldsymbol{\theta}$[18,131]. While in this manuscript we do not consider this scenario, it is worth exploring how less strict equivariance conditions affect the performance and the trainability guarantees here derived. Second, we note that as indicated in this work, the block diagonal structure of the S_n-equivariant QNN restricts the information in the input data that the model can access. This could lead to conditions where the model cannot solve the learning task as it cannot 'see' the relevant information in the input states. Such issue can be in principle solved by allowing the model to simultaneously act on multiple copies of the data, and even to change the representation of S_n throughout the circuit[23]. We also leave this exploration for future work.

Another potentially interesting research direction would be equivariant embeddings and re-uploading of classical data. For the purposes of this work, we make no assumptions to the source or form of the data, such as whether it is quantum or classical. However, when considering analyzing classical data on quantum computer, embeddings become important. We give one such example, which we call a 'fixed Hamming-weight encoding'. Another example is the standard encoding of a graph into a graph state, which we considered in our numerics. This is far from exhaustive and more sophisticated methods exist, including trainable encoding[54]. Similarly, we have not studied how our results change in the presence of data re-uploading[132]. We know that if the data is re-uploaded via equivariant generators (e.g., if the data re-uploading unitary takes the form $V(\boldsymbol{x}) = \prod_{l'} e^{-ix_l H_l}$, with H_l being S_n-equivariant), then our theoretical guarantees results do not change. This follows from the fact that the DLA of the circuit will remain the same, and hence our results follow.

We leave the study of more general encoding and re-uploading schemes for future work.

METHODS

This section provides an overview of the different tools used in the main text. Here we also present a sketch of the proof of our main results. Full details can be found in the Supplementary Methods.

Building S_n-equivariant operators

Here, we briefly describe how to build S_n-equivariant operators that can be used as generators of the QNN, or as measurement operators. In particular, we will focus on the so-called twirling method[19,23]. Take a unitary representation R of a discrete group G over a vector space V. Then the twirl operator is the linear map $\mathcal{T}_G : GL(V) \to GL(V)$, defined as

$$\mathcal{T}_G(A) = \frac{1}{|G|}\sum_{g\in G} R(g)AR(g)^\dagger. \tag{21}$$

It can be readily verified that the twirling of any operator A yields a G-equivariant operator, i.e., we have $[\mathcal{T}_G(A), R(g)] = 0$ for any $g \in G$.

The previous allows us to obtain a G-equivariant operator from any operator $A \in GL(V)$. For instance, let us consider the case in the case of $G = S_n$, R the qubit-defining representation and $A = X_1$. Then, we have $\mathcal{T}_G(X_1) = \frac{1}{n!}\sum_{\pi\in S_n} R(\pi)X_1R(\pi)^\dagger = \frac{1}{n}\sum_{i=1}^n X_i = \mathcal{T}_G(X_j)$ for any $1 \leqslant j \leqslant n$. Note that twirling over S_n cannot change the locality of an operator. That is, twirling a k-body operator leads to a sum of k-body operators.

Representation theory of S_n

In this section we review a few basic notions from representation theory. For a more thorough treatment we refer the reader to refs. [133–136], and more specifically to the tutorial in ref. [25] which provides an introduction to representation theory from the perspective of QML. We recall that we are interested in the qubit-defining representation of S_n, i.e., the one permuting qubits

$$R(\pi \in S_n)\bigotimes_{i=1}^n |\psi_i\rangle = \bigotimes_{i=1}^n \left|\psi_{\pi^{-1}(i)}\right\rangle.$$

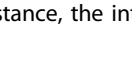
As mentioned in the main text, representations break down into fundamental building blocks called irreducible representations (irreps).

Definition 4. (Irrep decomposition). Given some unitary representation R of a compact group G, there exists a basis under which it takes a block diagonal form

$$R(g \in G) \cong \bigoplus_\lambda \bigoplus_{\mu=1}^{m_{r_\lambda}} r_\lambda(\pi) = \bigoplus_\lambda r_\lambda(\pi)\otimes \mathbb{1}_{m_{r_\lambda}}, \tag{22}$$

with $r_\lambda(\pi)$ irreps of G appearing m_{r_λ} times.

The irreps of the symmetric group are commonly labeled by the set of partitions of the integer n. A partition of a positive integer $n \in \mathbb{N}$ is a non-decreasing sequence of positive integers $\lambda = (\lambda_1, \cdots, \lambda_k)$ satisfying $\sum_i \lambda_i = n$. Partitions are typically visualized using young diagrams, a set of empty, left-justified boxes arranged in rows such that there are λ_i boxes in the i-th row. For instance, the integer $n = 3$ can split into

$(3,0) =$  , $(2,1) =$  , $(1,1,1) =$  . (23)

We note that in the case of the qubit-defining representation, the only λ appearing in Eq. (22) have at most two rows (e.g., would not include the last partition in Eq. (23)).

The dimension of an S_n irrep r_λ can be computed from the hook length formula

$$\dim(r_\lambda) = \frac{n!}{\prod_{b\in\lambda} h_\lambda(b)}, \tag{24}$$

where each $h_\lambda(b)$ is the hook length for box b in λ, which is the total number of boxes in a 'hook' (or 'l' shape) composed of box b and every box beneath (in the same column) and to its right (in the same row).

Given the block-diagonal structure of R in Eq. (22), one can see that a general G-equivariant operator has to be of the form

$$A \cong \bigoplus_\lambda \mathbb{1}_{\dim(r_\lambda)} \otimes A_\lambda, \tag{25}$$

where A_λ are m_{r_λ}-dimensional matrices repeated $\dim(r_\lambda)$ times. In general, the number of times an irrep appears in an arbitrary representation R (i.e., m_{r_λ} in Eq. (22)) can be determined through character theory. Instead, in our case, we will take a shortcut and exploit one of the most remarkable results in representation theory, called the Schur-Weyl duality[137].

Consider the representation Q of the unitary group $\mathbb{U}(2)$ acting on $\mathcal{H} = (\mathbb{C}^2)^{\otimes n}$ through the n-fold tensor product $Q(W \in \mathbb{U}(2)) = W^{\otimes n}$. Evidently, according to Eq. (22), Q will also have an isotypic decomposition

$$Q(W \in \mathbb{U}(2)) = \bigoplus_s \mathbb{1}_{m_{q_s}} \otimes q_s(W), \tag{26}$$

where s labels the different (spin) irreps of $\mathbb{U}(2)$. The Schur-Weyl duality, states that the matrix algebras $\mathbb{C}[R]$ and $\mathbb{C}[Q]$ mutually centralize each other, meaning that $\mathbb{C}[R]$ is the space of $\mathbb{U}(2)$-equivariant linear operators, and similarly $\mathbb{C}[Q]$ is the space of S_n-equivariant ones. As a consequence of this duality, $\mathcal{H}$ can be decomposed as $\mathcal{H} \cong \bigoplus_\lambda V_\lambda \otimes W_\lambda$, where λ simultaneously labels irrep spaces V_λ and W_λ for S_n and $\mathbb{U}(2)$, respectively. That is, $\mathcal{H}$ supports a simultaneous action of S_n and $\mathbb{U}(2)$, where the irreps of each appear exactly once and are correlated: Each of the two-row Young diagrams $\lambda = (n-m, m)$ labeling the irreps in R can be associated unequivocally with a spin label $s(\lambda)$ for an $\mathbb{U}(2)$ irrep appearing in Q

$$s(\lambda) = \frac{\lambda_1 - \lambda_2}{2} = \frac{n-2m}{2}. \tag{27}$$

Moreover, since under the joint action of $S_n \times \mathbb{U}(2)$ the multiplicities are one, one can assert that the irrep q_λ of $\mathbb{U}(2)$ appears $\dim(r_\lambda)$-times in Q, and conversely, the irrep r_λ of S_n appears $\dim(q_\lambda)$-times in R. Using the well-known dimension of spin irreps $\dim(q_s) = 2s+1$, we can derive an expression for the multiplicity of S_n irreps

$$m_{r_\lambda} = \dim(q_{s(\lambda)}) = 2s(\lambda) + 1 = n - 2m + 1. \tag{28}$$

Also, it is straightforwards to adapt the formula in Eq. (24) to two-row diagrams $\lambda = (n-m, m)$

$$\dim(r_\lambda) = \frac{n!(n-2m+1)!}{(n-m+1)!m!(n-2m)!}. \tag{29}$$

We finally note that, since we are ultimately interested in S_n-equivariant operators, in the main text we have defined $d_\lambda \equiv m_{r_\lambda}$ and $m_\lambda \equiv \dim(r_\lambda)$. That is, the dimension and multiplicity of an irrep in the main text are for the representations of $\mathbb{U}$.

Universality, expressibility, and dynamical Lie algebra

In the main text we have argued that the set of generators in Eq. (9) is universal within each invariant subspace. Here we will formalize this statement.

First, let us recall that we say that a parametrized unitary is universal if it can generate any unitary (up to a global phase) in the space over which it acts. One can quantify the capacity of being able to create different unitaries through the so-called measures of expressibility[37,43,46,128]. Here we will focus on the notion of potential expressibility of a given QNN, which is formalized via the dynamical Lie algebra of the architecture[138].

Definition 5. (Dynamical Lie algebra). Given a set of generators $\mathcal{G}$ defining a QNN, its dynamical Lie algebra $\mathfrak{g}$ is the span of the Lie closure $\langle\cdot\rangle_{\text{Lie}}$ of $\mathcal{G}$. That is, $\mathfrak{g} = \text{span}_{\mathbb{R}}\langle\mathcal{G}\rangle_{\text{Lie}}$, where $\langle\mathcal{G}\rangle_{\text{Lie}}$ is defined as the set of all the nested commutators generated by the elements of $\mathcal{G}$.

In particular, the dynamical Lie algebra (DLA) fully characterizes the group of unitaries that can be ultimately expressed by the circuit: for any unitary U realized by a QNN with generators in $\mathcal{G}$ there exists an anti-hermitian operator $\eta \in \mathfrak{g} = \langle\mathcal{G}\rangle_{\text{Lie}}$ such that $U = e^{\eta}$. Evidently, $\mathfrak{g} \subseteq \mathfrak{u}(d)$, that is, it is a subalgebra of the space of anti-hermitian operators. When $\mathfrak{g}$ is $\mathfrak{su}(d)$ or $\mathfrak{u}(d)$ we say that the QNN is controllable or universal since for any pair of states $|\psi\rangle$ and $|\phi\rangle$, there exists a unitary $U = e^{\eta}$ with $\eta \in \mathfrak{g}$ such that $|\langle\phi|U|\psi\rangle|^2 = 1$.

In the framework of GQML one designs symmetry-respecting QNNs by using group-equivariant generators. This implies that the corresponding DLA is constrained and necessarily takes the form

$$\mathfrak{g} = \bigoplus_{\lambda} \mathbb{1}_{m_\lambda} \otimes \mathfrak{g}_\lambda, \tag{30}$$

where $\mathfrak{g}_\lambda \subseteq \mathfrak{u}(d_\lambda)$. For this scenario, we provide a notion of controllability restricted to each of the invariant subspaces: We say that a QNN is subspace-controllable in the isotypic component λ if $\mathfrak{g}_\lambda$ is $\mathfrak{su}(d_\lambda)$ or $\mathfrak{u}(d_\lambda)$. This means that the QNN can map between any pair of states in every $\mathcal{H}_\lambda^v$. Notably, the following result follows from Refs. [92,139].

Lemma 3. (Subspace controllability). The set of S_n-equivariant generators in Eq. (9) is subspace-controllable in every λ.

As shown below, this result will be crucial for the proof of Theorem 1.

Proof of absence of barren plateaus

Here we sketch our proof of Theorem 1. Our goal is to calculate $\text{Var}_{\boldsymbol{\theta}}[\partial_\mu\widehat{\mathcal{L}}(\boldsymbol{\theta})] = \mathbb{E}_{\boldsymbol{\theta}}[(\partial_\mu\widehat{\mathcal{L}}(\boldsymbol{\theta}))^2] - \mathbb{E}_{\boldsymbol{\theta}}[\partial_\mu\widehat{\mathcal{L}}(\boldsymbol{\theta})]^2$. In general, we will have to deal with integrals of the form $\int_{\mathcal{D}_{\boldsymbol{\theta}}} f(U(\boldsymbol{\theta}))$ where f is some parametrized function—for example the cost function or its partial derivatives– and $\mathcal{D}_{\boldsymbol{\theta}} : [0, 2\pi]^M \to [0, 1]$ is some distribution over parameter space—typically the uniform distribution. The first step is to transform the integration over parameter space to an integration over the resulting QNN unitary distribution $\mathcal{D}$. Since $\mathcal{D}$ is known to converge (given enough depth) to ϵ-approximate 2-designs over the Lie group $e^{\mathfrak{g}}$[43,140], assuming f is a polynomial of degree $\leqslant 2$ in the entries of U (as is the case of interest), we can replace the integration over $\mathcal{D}$ with an integration over the Haar measure over $e^{\mathfrak{g}}$. In general, $\mathfrak{g}$ is a reductive Lie algebra consisting of multiple orthogonal ideals $\mathfrak{g} = \bigoplus_\lambda \mathfrak{g}_\lambda$, where $\mathfrak{g}_i$ is either simple or abelian, and the Lie group $e^{\mathfrak{g}}$ is the product group $\bigotimes_\lambda e^{\mathfrak{g}_\lambda}$. It can be shown (see Supplementary Methods 4) that the Haar measure over such a product group is the product of the Haar measures over the normal subgroups $e^{\mathfrak{g}_\lambda}$. Finally, the ansatz with generators in Eq. (9) has a DLA $\mathfrak{g}$ that is subspace controllable, meaning that each simple $\mathfrak{g}_\lambda$ is either $\mathfrak{su}(d_\lambda)$ or $\mathfrak{u}(d_\lambda)$[92,139]. Summarizing, we have

$$\begin{aligned}\int_{\mathcal{D}_{\boldsymbol{\theta}}} d\boldsymbol{\theta} f(U(\boldsymbol{\theta})) &= \int_{\mathcal{D}} dU f(U) \\ &\to \int_{e^{\mathfrak{g}}} d\mu(U) f(U) \\ &= \prod_\lambda \int_{\mathbb{U}(d_\lambda)} d\mu_\lambda(U_\lambda) f(\{U_\lambda\}).\end{aligned} \tag{31}$$

The main advantage of Eq. (31) is that we can use tools from Weingarten calculus to perform symbolic integration over the Haar measure of unitary groups[141]. Explicitly, we care for the variance of $\partial_\mu\widehat{\mathcal{L}}(\boldsymbol{\theta}) = \sum_{i=1}^M c_i \partial_\mu \ell_{\boldsymbol{\theta}}(\rho_i)$ where

$$\partial_\mu \ell_{\boldsymbol{\theta}}(\rho_i) = i\text{Tr}[U_B \rho_i U_B^\dagger [H_\mu, U_A^\dagger O U_A]],$$

where U_B and U_A denote the unitary circuits before and after the parametrized gate we are differentiating. Assuming that the depth L of the QNN is enough to guarantee that both U_A and U_B form independent 2-designs on $e^{\mathfrak{g}}$, we can use Weingarten calculus to evaluate the terms in $\mathbb{E}_{\boldsymbol{\theta}}[(\partial_\mu\widehat{\mathcal{L}}(\boldsymbol{\theta}))^2]\,\mathbb{E}_{\boldsymbol{\theta}}[\partial_\mu\widehat{\mathcal{L}}(\boldsymbol{\theta})]^2$, and obtain Eq. (16) in Theorem 1. The details of this calculation are presented in Supplementary Methods 4.

While the previous, along with the results in Theorem 2, allow to prove by direct construction that S_n-equivariant QNNs do not lead to barren plateaus, we here provide further intuition for this result in terms of the expressibility reduction induced by the equivariance inductive biases. As shown in ref. [37], QNNs that are too expressible exhibit exponentially vanishing gradients, whereas models whose expressibility is restricted can exhibit large gradients. Hence, we can expect the result in Corollary 1 to be a direct consequence of the reduced expressibility of the model. We can further formalize this statement using the results of ref. [43]. Therein, it was found that there exists a link between the presence or absence of barren plateaus and the dimension of the DLA. In particular, the authors conjecture, and prove for several examples (see also ref. [142] for an independent verification of the conjecture), that deep QNNs have gradients that scale inversely with the size of the DLA, that is, $\text{Var}_{\boldsymbol{\theta}}[\partial\hat{\mathcal{L}}(\boldsymbol{\theta})] \sim \frac{1}{\text{poly}(\dim(\mathfrak{g}))}$. For the case of S_n-equivariant QNNs we know from Lemma 3 that $\dim(\mathfrak{g}) \in \Theta(n^3)$ thus indicating that the variance should only vanish polynomially with n (for an appropriate dataset). We note this conjecture was recently proven[140,143].

Intuition behind the overparametrization phenomenon

Recently, ref. [32] studied the overparametrization of QNNs from the perspective of a complexity phase transition in the loss landscape. In the underparametrized regime, we experience rough loss landscapes, which in turn can be traced back to a lack of control in parametrized state space. When the number of parameters is below the number of directions in state space, the parameter update can only access a subset of those potential directions. This constraint can be shown to introduce false local minima, that is, local minima that are not actual minima of the loss function (as a function of state space) but instead artifacts of a poor parametrization. Instead, upon introduction of more parameters the parametrized state starts accessing these previously unavailable directions, and false minima disappear as we transition into the overparametrized regime. Because in the overparametrized regime the number of parameters is greater than the number of ever accessible directions, solutions in the control landscape are degenerate and form multidimensional submanifolds, allowing the optimizer to reach them more easily[125,126].

The main contribution in ref. [32] is the realization that, under standard assumptions, one needs one parameter per potentially accessible direction in state space, and that the latter can be formalized as the dimension of the orbit of the initial state under the Lie group $e^{\mathfrak{g}}$ resulting from the exponential of the DLA $\mathfrak{g}$. In particular, this means that exponential DLA architectures require an exponential number of parameters to be overparametrized,

whereas polynomial DLA architectures only need a polynomial number of them.

With these definitions, the proof of Theorem 3 is immediate. Since the ansatz is subspace controllable (Lemma 3), the dimension of the DLA is equal to the dimension of the commutant, which is $\Theta(n^3)$ (Lemma 2).

To finish, we note that the definition of overparametrization employed here (in terms of saturating the number of available directions) might differ from some definitions of overparametrization in the classical neural network community. Namely, in classical machine learning researchers have studied overparametrization through the optics of generalization[109,144–147], while others have investigated the effect of overparametrization on the training processes. In particular, it has been proposed that the onset of overparametrization can be detected using metrics such as parameter redundancy which is captured by the rank of the classical Fisher information matrix[148–150]. It is precisely this notion of overparametrization that ref. [32] ported to quantum, and the one used in the present work.

Generalization

We consider the QML setting in this paper where the empirical loss function is of the form $\widehat{\mathcal{L}}(\boldsymbol{\theta}) = \sum_{i=1}^{M} c_i \mathrm{Tr}[U_{\boldsymbol{\theta}}(\rho_i)O]$. We assume that the operator norm of O is bounded by a constant and also $|c_i| \leqslant 1/M$. We follow closely the covering number-based generalization bound in ref. [61]. First recall that a set V is ε-covered by a subset $K \subseteq V$ with respect to a distance metric d if $\forall\, x \in V, \exists\, y \in K$ such that $d(x,y) \leqslant \varepsilon$. The ε-covering number (w.r.t. metric d) of V, denoted as $\mathcal{N}(V,d,\varepsilon)$, is the cardinality of the smallest such subset[117]. The following theorem bounds the ε-covering number of S_n-equivariant QNNs.

Theorem 5. The ε-covering number of the set $\mathcal{V}_n$ of n-qubit unitary S_n-equivariant QNNs w.r.t. the operator norm $\|\cdot\|$ can be bounded as $\mathcal{N}(\mathcal{V}_n, \|\cdot\|, \varepsilon) \leqslant \left(\frac{6}{\varepsilon}\right)^{2\mathrm{Te}_{n+1}}$.

Proof. Recall that an S_n-EQNN U can be block-diagonalized as $U \cong \bigoplus_\lambda \mathbb{1}_{m_\lambda} \otimes U_\lambda$, where each U_λ is a unitary for U to be unitary. Let $\mathbb{U}(d_\lambda)$ denote the set of all unitaries of dimension d_λ. Following Lemma 6 in ref. [61] and Section 4.2 in ref. [151] we can bound the ε-covering number of $\mathbb{U}_{d_\lambda}$ as follows

$$\mathcal{N}(\mathbb{U}(d_\lambda), \|\cdot\|, \varepsilon) \leqslant \left(\frac{6}{\varepsilon}\right)^{2d_\lambda^2}. \tag{32}$$

Next, we construct an ε-covering subset of the S_n-equivariant unitary set, $\mathcal{V}_n$, from the ε-covering subsets, K_λ, of the blocks λ. Indeed, given any $U \cong \bigoplus_\lambda \mathbb{1}_{m_\lambda} \otimes U_\lambda$, we can identify unitaries $\tilde{U}_\lambda$ from K_λ such that $\|U_\lambda - \tilde{U}_\lambda\| \leqslant \varepsilon, \forall \lambda$. The unitary $\tilde{U} \cong \bigoplus_\lambda \mathbb{1}_{m_\lambda} \otimes \tilde{U}_\lambda$ then satisfies

$$\|U - \tilde{U}\| \leqslant \max_\lambda \|U_\lambda - \tilde{U}_\lambda\| \leqslant \varepsilon. \tag{33}$$

Therefore, there exists an ε-covering net of $\mathcal{V}_n$ of size $\prod_\lambda \left(\frac{6}{\varepsilon}\right)^{2d_\lambda^2} = \left(\frac{6}{\varepsilon}\right)^{2\mathrm{Te}_{n+1}}$, concluding the proof. □

Having established this bound on the ε-covering numbers of S_n-EQNN, we apply a known result from ref. [61] (with some extra care) to obtain Theorem 4.

Proof. (Proof of Theorem 4). We assume knowledge of Theorem 6 in ref. [61]. In step two of the proof where the authors use the chaining argument[152] to bound the generalization error, notice that the covering number $\mathcal{N}_j$ in their Eq. (64) is replaced by $\left(\frac{6}{\varepsilon}\right)^{2\mathrm{Te}_{n+1}}$ in our case. In other words, there is no architecture-dependence (the number of gates T in their case) inside the logarithm in the resulting Eq. (65). Applying this change to the rest of their proof leads to our claimed generalization bound. □

We note that in the previous derivation, we have used knowledge of the isotypic decomposition of the S_n-equivariant QNN, which allows us to obtain a specialized generalization error bound that does not follow from a direct application of the results in ref. [61].

Trainable and untrainable states

Here, we describe how the states in Table 1 are obtained. The "symmetric states" are obtained from the symmetric subspace[153], i.e., the set of states $\{|\psi\rangle \in \mathcal{H} \,|\, R(\pi)|\psi\rangle = |\psi\rangle, \forall \pi \in S_n\}$. The so-called "fixed Hamming-weight encoded" states correspond to states representing classical data: Given an array of real values $\{x_i\}$, such that $\sum_i x_i^2 = 1$, each x_i is encoded as the weight of a unique bitstring $\mathbf{z}$ of Hamming weight k, where k is some fixed constant. That is, prepare the state $|\mathbf{x}\rangle = \sum_{\mathbf{z}\,\text{s.t.}\, w(\mathbf{z})=k} x_{\mathbf{z}}|\mathbf{z}\rangle$, where we are now indexing x_i with a bitstring $\mathbf{z}$. "Local Haar random" states are obtained by preparing the state $|0\rangle^{\otimes n}$ and applying a Haar random single-qubit unitary to each qubit. "Global Haar random" states are obtained by preparing the state $|0\rangle^{\otimes n}$ and applying a random n-qubit unitary sampled from the Haar measure over $\mathbb{U}(d)$. The "fixed and linear depth random circuit" states correspond to the states obtained by preparing the state $|0\rangle^{\otimes n}$ and respectively applying a constant-depth, or linear-depth layered hardware-efficient quantum circuit[34,130] with random parameters. For the "graph states", we use a canonical encoding to embed a graph into a quantum state[121,122]. Specifically, to create a graph state, one starts with the state $|+\rangle^{\otimes n}$, and applies a controlled-Z rotation for every edge in the graph. We consider 3-regular and $n/2$-regular graphs, as well as random graphs generated according to the Erdös-Rényi model[120].

DATA AVAILABILITY

Data generated and analyzed during current study are available from the corresponding author upon reasonable request.

CODE AVAILABILITY

Code used to generate data in this study are available from the corresponding author upon reasonable request.

Received: 10 January 2023; Accepted: 6 January 2024;
Published online: 22 January 2024

REFERENCES

1. Cohen, T. & Welling, M. Group equivariant convolutional networks. In: *Proc. International Conference on Machine Learning*. **33** (2016).
2. Bronstein, M. M., Bruna, J., Cohen, T. & Veličković, P. Geometric deep learning: Grids, groups, graphs, geodesics, and gauges. Preprint at https://arxiv.org/abs/2104.13478 (2021).
3. Kondor, R. & Trivedi, S. On the generalization of equivariance and convolution in neural networks to the action of compact groups. In: *Proc. International Conference on Machine Learning* **35** (2018).
4. Bogatskiy, A. et al. Symmetry group equivariant architectures for physics. In: *Proc. 2021 US Community Study on the Future of Particle Physics* (2021).
5. Bekkers, E. J. et al. Roto-translation covariant convolutional networks for medical image analysis. In: *Proc. International Conference on Medical Image Computing and Computer-Assisted Intervention*, 440–448 (2018).
6. Schütt, K. T. et al. Schnet: A continuous-filter convolutional neural network for modeling quantum interactions. *Adv. Neural Inf. Process. Sys.* **31**, 992–1002 (2017).
7. Boyda, D. et al. Sampling using su (n) gauge equivariant flows. *Phys. Rev. D* **103**, 074504 (2021).

8. Rezende, D. J., Racanière, S., Higgins, I. & Toth, P. Equivariant Hamiltonian flows. Preprint at https://arxiv.org/abs/1909.13739 (2019).
9. Thomas, N. et al. Tensor field networks: Rotation-and translation-equivariant neural networks for 3d point clouds. Preprint at https://arxiv.org/abs/1802.08219 (2018).
10. Toth, P. et al. Hamiltonian generative networks. In: *Proc. International Conference on Learning Representations* **8** (2020).
11. Köhler, J., Klein, L. & Noé, F. Equivariant flows: Exact likelihood generative learning for symmetric densities. In: *Proc. International Conference on Machine Learning* **37** (2020).
12. Anderson, B., Hy, T. S. & Kondor, R. Cormorant: covariant molecular neural networks. *Adv. Neural Inf. Process. Syst.* **32**, 14537–14546 (2019).
13. Bogatskiy, A. et al. Lorentz group equivariant neural network for particle physics. In: *Proc. International Conference on Machine Learning* **37** (2020).
14. Schuld, M., Sinayskiy, I. & Petruccione, F. An introduction to quantum machine learning. *Contemp. Phys.* **56**, 172–185 (2015).
15. Biamonte, J. et al. Quantum machine learning. *Nature* **549**, 195–202 (2017).
16. Cerezo, M., Verdon, G., Huang, H.-Y., Cincio, L. & Coles, P. J. Challenges and opportunities in quantum machine learning. *Nat. Comput. Sci.* **2**, 567–576 (2022).
17. 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).
18. Larocca, M. et al. Group-invariant quantum machine learning. *PRX Quantum* **3**, 030341 (2022).
19. Meyer, J. J. et al. Exploiting symmetry in variational quantum machine learning. *PRX Quantum* **4**, 010328 (2023).
20. Sauvage, F., Larocca, M., Coles, P. J. & Cerezo, M. Building spatial symmetries into parameterized quantum circuits for faster training. *Quantum Sci. Technol.* **9**, 01509 (2024).
21. Zheng, H., Li, Z., Liu, J., Strelchuk, S. & Kondor, R. On the super-exponential quantum speedup of equivariant quantum machine learning algorithms with su(*d*) symmetry. Preprint at https://arxiv.org/abs/2207.07250 (2022).
22. Zheng, H., Li, Z., Liu, J., Strelchuk, S. & Kondor, R. Speeding up learning quantum states through group equivariant convolutional quantum ansätze. *PRX Quantum* **4**, 020327 (2023).
23. Nguyen, Q. T. et al. A theory for equivariant quantum neural networks. Preprint at https://arxiv.org/abs/2210.08566 (2022).
24. Wang, X. et al. Symmetric pruning in quantum neural networks. In: *Proc. International Conference on Learning Representations* **11** (2023).
25. Ragone, M. et al. Representation theory for geometric quantum machine learning. Preprint at https://arxiv.org/abs/2210.07980 (2022).
26. Abbas, A. et al. The power of quantum neural networks. *Nat. Comput. Sci.* **1**, 403–409 (2021).
27. Liu, J. et al. An analytic theory for the dynamics of wide quantum neural network. *Phys. Rev. Lett.* **130**, 150601 (2023).
28. Liu, J., Tacchino, F., Glick, J. R., Jiang, L. & Mezzacapo, A. Representation learning via quantum neural tangent kernels. *PRX Quantum* **3**, 030323 (2022).
29. Bittel, L. & Kliesch, M. Training variational quantum algorithms is NP-hard. *Phys. Rev. Lett.* **127**, 120502 (2021).
30. Anschuetz, E. R. & Kiani, B. T. Beyond barren plateaus: Quantum variational algorithms are swamped with traps. *Nat. Commun.* **13**, 7760 (2022).
31. Fontana, E., Cerezo, M., Arrasmith, A., Rungger, I. & Coles, P. J. Non-trivial symmetries in quantum landscapes and their resilience to quantum noise. *Quantum* **6**, 804 (2022).
32. Larocca, M., Ju, N., García-Martín, D., Coles, P. J. & Cerezo, M. Theory of overparametrization in quantum neural networks. *Nat. Comput. Sci.* **3**, 542—551 (2023).
33. McClean, J. R., Boixo, S., Smelyanskiy, V. N., Babbush, R. & Neven, H. Barren plateaus in quantum neural network training landscapes. *Nat. Commun.* **9**, 1–6 (2018).
34. Cerezo, M., Sone, A., Volkoff, T., Cincio, L. & Coles, P. J. Cost function dependent barren plateaus in shallow parametrized quantum circuits. *Nat. Commun.* **12**, 1–12 (2021).
35. Sharma, K., Cerezo, M., Cincio, L. & Coles, P. J. Trainability of dissipative perceptron-based quantum neural networks. *Phys. Rev. Lett.* **128**, 180505 (2022).
36. Holmes, Z. et al. Barren plateaus preclude learning scramblers. *Phys. Rev. Lett.* **126**, 190501 (2021).
37. Holmes, Z., Sharma, K., Cerezo, M. & Coles, P. J. Connecting ansatz expressibility to gradient magnitudes and barren plateaus. *PRX Quantum* **3**, 010313 (2022).
38. Cerezo, M. & Coles, P. J. Higher order derivatives of quantum neural networks with barren plateaus. *Quantum Sci. Technol.* **6**, 035006 (2021).
39. Marrero, C. O., Kieferová, M. & Wiebe, N. Entanglement-induced barren plateaus. *PRX Quantum* **2**, 040316 (2021).
40. Patti, T. L., Najafi, K., Gao, X. & Yelin, S. F. Entanglement devised barren plateau mitigation. *Phys. Rev. Res.* **3**, 033090 (2021).
41. Uvarov, A. & Biamonte, J. D. On barren plateaus and cost function locality in variational quantum algorithms. *Journal of Physics A: Mathematical and Theoretical* **54**, 245301 (2021).
42. Thanasilp, S., Wang, S., Nghiem, N. A., Coles, P. J. & Cerezo, M. Subtleties in the trainability of quantum machine learning models. *Quantum Mach. Intell.* **5**, 21 (2023).
43. Larocca, M. et al. Diagnosing barren plateaus with tools from quantum optimal control. *Quantum* **6**, 824 (2022).
44. Wang, S. et al. Noise-induced barren plateaus in variational quantum algorithms. *Nat. Commun.* **12**, 1–11 (2021).
45. Wiersema, R. et al. Exploring entanglement and optimization within the Hamiltonian variational ansatz. *PRX Quantum* **1**, 020319 (2020).
46. Sim, S., Johnson, P. D. & Aspuru-Guzik, A. Expressibility and entangling capability of parameterized quantum circuits for hybrid quantum-classical algorithms. *Adv. Quantum Technol.* **2**, 1900070 (2019).
47. Zaheer, M. et al. Deep sets. Guyon, I.et al. (eds.) *Adv. Neural Inf. Process. Sys.* **30** (2017).
48. Maron, H., Litany, O., Chechik, G. & Fetaya, E. On learning sets of symmetric elements. In: *Proc. International Conference on Machine Learning* **37** (2020).
49. Maron, H., Ben-Hamu, H., Shamir, N. & Lipman, Y. Invariant and equivariant graph networks. In: *Proc. International Conference on Learning Representations* (2019).
50. Keriven, N. & Peyré, G. Universal invariant and equivariant graph neural networks. In *Proc. Adv. Neural Inf. Process. Syst.* **32** (2019).
51. Maron, H., Ben-Hamu, H., Serviansky, H. & Lipman, Y. Provably powerful graph networks. In: *Proc. Adv. Neural Inf. Process. Sys.* **32** (2019).
52. Verdon, G. et al. Quantum graph neural networks. Preprint at https://arxiv.org/abs/1909.12264 (2019).
53. Mernyei, P., Meichanetzidis, K. & Ceylan, I. I. Equivariant quantum graph circuits. In *Proc. International Conference on Machine Learning* **39** (2022).
54. Skolik, A., Cattelan, M., Yarkoni, S., Bäck, T. & Dunjko, V. Equivariant quantum circuits for learning on weighted graphs. *Npj Quantum Inf.* **9**, 47 (2023).
55. Maron, H., Fetaya, E., Segol, N. & Lipman, Y. On the universality of invariant networks. In: *Proc. International Conference on Machine Learning* **36** (2019).
56. Thiede, E. H., Hy, T. S. & Kondor, R. The general theory of permutation equivariant neural networks and higher order graph variational encoders. Preprint at https://arxiv.org/abs/2004.03990 (2020).
57. Pan, H. & Kondor, R. Permutation equivariant layers for higher order interactions. In Camps-Valls, G., Ruiz, F. J. R. & Valera, I. (eds.) *Proceedings of the 25th International Conference on Artificial Intelligence and Statistics*, vol. 151 of *Proceedings of Machine Learning Research*, 5987–6001 (PMLR, 2022).
58. Farhi, E., Goldstone, J. & Gutmann, S. A quantum approximate optimization algorithm. Preprint at https://arxiv.org/abs/1411.4028 (2014).
59. Hadfield, S. et al. From the quantum approximate optimization algorithm to a quantum alternating operator ansatz. *Algorithms* **12**, 34 (2019).
60. Cong, I., Choi, S. & Lukin, M. D. Quantum convolutional neural networks. *Nat. Phys.* **15**, 1273–1278 (2019).
61. Caro, M. C. et al. Generalization in quantum machine learning from few training data. *Nat. Commun.* **13**, 4919 (2022).
62. Peruzzo, A. et al. A variational eigenvalue solver on a photonic quantum processor. *Nat. Commun.* **5**, 1–7 (2014).
63. Cerezo, M. et al. Variational quantum algorithms. *Nat. Rev. Phys.* **3**, 625–644 (2021).
64. Tang, H. L. et al. qubit-adapt-vqe: an adaptive algorithm for constructing hardware-efficient ansätze on a quantum processor. *PRX Quantum* **2**, 020310 (2021).
65. Horodecki, R., Horodecki, P., Horodecki, M. & Horodecki, K. Quantum entanglement. *Rev. Mod. Phys.* **81**, 865 (2009).
66. Walter, M., Gross, D. & Eisert, J. Multipartite entanglement. *Quantum Information: From Foundations to Quantum Technology Applications* 293–330 (John Wiley and & Sons, Inc., 2016).
67. Beckey, J. L., Gigena, N., Coles, P. J. & Cerezo, M. Computable and operationally meaningful multipartite entanglement measures. *Phys. Rev. Lett.* **127**, 140501 (2021).
68. Schatzki, L., Liu, G., Cerezo, M. & Chitambar, E. A hierarchy of multipartite correlations based on concentratable entanglement. Preprint at https://arxiv.org/pdf/2209.07607.pdf (2022).
69. Guo, X. et al. Distributed quantum sensing in a continuous-variable entangled network. *Nat. Phys.* **16**, 281–284 (2020).
70. Zhang, Z. & Zhuang, Q. Distributed quantum sensing. *Quantum Sci. Technol.* **6**, 043001 (2021).
71. Huerta Alderete, C. et al. Inference-based quantum sensing. *Phys. Rev. Lett.* **129**, 190501 (2022).

72. Otterbach, J. S. et al. Unsupervised machine learning on a hybrid quantum computer. Preprint at https://arxiv.org/abs/1712.05771 (2017).
73. Kerenidis, I., Landman, J., Luongo, A. & Prakash, A. q-means: a quantum algorithm for unsupervised machine learning. *Adv. Neural Inf. Process. Sys.* **32**, 4134–4144 (2019).
74. Saggio, V. et al. Experimental quantum speed-up in reinforcement learning agents. *Nature* **591**, 229–233 (2021).
75. Skolik, A., Jerbi, S. & Dunjko, V. Quantum agents in the gym: a variational quantum algorithm for deep q-learning. *Quantum* **6**, 270 (2022).
76. Dallaire-Demers, P.-L. & Killoran, N. Quantum generative adversarial networks. *Phys. Rev. A* **98**, 012324 (2018).
77. Benedetti, M. et al. A generative modeling approach for benchmarking and training shallow quantum circuits. *Npj Quantum Inf.* **5**, 45 (2019).
78. Kieferova, M., Carlos, O. M. & Wiebe, N. Quantum generative training using rényi divergences. Preprint at https://arxiv.org/abs/2106.09567 (2021).
79. Romero, J. & Aspuru-Guzik, A. Variational quantum generators: generative adversarial quantum machine learning for continuous distributions. *Adv. Quantum Technol.* **4**, 2000003 (2021).
80. Bharti, K. et al. Noisy intermediate-scale quantum algorithms. *Rev. Mod. Phys.* **94**, 015004 (2022).
81. Havlíček, V. et al. Supervised learning with quantum-enhanced feature spaces. *Nature* **567**, 209–212 (2019).
82. Schuld, M. & Petruccione, F. *Supervised Learning with Quantum Computers*, vol. 17 (Springer, 2018).
83. Schatzki, L., Arrasmith, A., Coles, P. J. & Cerezo, M. Entangled datasets for quantum machine learning. Preprint at https://arxiv.org/abs/2109.03400 (2021).
84. Janocha, K. & Czarnecki, W. M. On loss functions for deep neural networks in classification. In: *Proc. Theoretical Foundations of Machine Learning* **25** (2017).
85. Grzesiak, N. et al. Efficient arbitrary simultaneously entangling gates on a trapped-ion quantum computer. *Nat. Commun.* **11**, 2963 (2020).
86. Pino, J. M. et al. Demonstration of the trapped-ion quantum ccd computer architecture. *Nature* **592**, 209–213 (2021).
87. Bluvstein, D. et al. A quantum processor based on coherent transport of entangled atom arrays. *Nature* **604**, 451–456 (2022).
88. Pedrozo-Peñafiel, E. et al. Entanglement on an optical atomic-clock transition. *Nature* **588**, 414–418 (2020).
89. Marciniak, C. D. et al. Optimal metrology with programmable quantum sensors. *Nature* **603**, 604–609 (2022).
90. Kitagawa, M. & Ueda, M. Squeezed spin states. *Phys. Rev. A* **47**, 5138–5143 (1993).
91. Wineland, D. J., Bollinger, J. J., Itano, W. M., Moore, F. & Heinzen, D. J. Spin squeezing and reduced quantum noise in spectroscopy. *Phys. Rev. A* **46**, R6797 (1992).
92. Albertini, F. & D'Alessandro, D. Controllability of symmetric spin networks. *J. Math. Phys.* **59**, 052102 (2018).
93. Grant, E., Wossnig, L., Ostaszewski, M. & Benedetti, M. An initialization strategy for addressing barren plateaus in parametrized quantum circuits. *Quantum* **3**, 214 (2019).
94. Skolik, A., McClean, J. R., Mohseni, M., van der Smagt, P. & Leib, M. Layerwise learning for quantum neural networks. *Quantum Mach. Intell.* **3**, 1–11 (2021).
95. Sauvage, F. et al. Flip: A flexible initializer for arbitrarily-sized parametrized quantum circuits. Preprint at https://arxiv.org/abs/2103.08572 (2021).
96. Sack, S. H., Medina, R. A., Michailidis, A. A., Kueng, R. & Serbyn, M. Avoiding barren plateaus using classical shadows. *PRX Quantum* **3**, 020365 (2022).
97. Rad, A., Seif, A. & Linke, N. M. Surviving the barren plateau in variational quantum circuits with bayesian learning initialization. Preprint at https://arxiv.org/abs/2203.02464 (2022).
98. Broers, L. & Mathey, L. Optimization of quantum algorithm protocols without barren plateaus. Preprint at https://arxiv.org/abs/2111.08085 (2021).
99. Liu, H.-Y., Sun, T.-P., Wu, Y.-C., Han, Y.-J. & Guo, G.-P. A parameter initialization method for variational quantum algorithms to mitigate barren plateaus based on transfer learning. *New J. Phys.* **25**, 013039 (2023).
100. Friedrich, L. & Maziero, J. Avoiding barren plateaus with classical deep neural networks. *Phys. Rev. A* **106**, 042433 (2022).
101. Kulshrestha, A. & Safro, I. Beinit: Avoiding barren plateaus in variational quantum algorithms. In *Proc. IEEE International Conference on Quantum Computing and Engineering (QCE)*, 197–203 (IEEE, 2022).
102. Mele, A. A., Mbeng, G. B., Santoro, G. E., Collura, M. & Torta, P. Avoiding barren plateaus via transferability of smooth solutions in Hamiltonian variational ansatz. *Phys. Rev. A* **106**, 060401 (2022).
103. Zhang, K., Hsieh, M.-H., Liu, L. & Tao, D. Escaping from the Barren Plateau via Gaussian Initializations in Deep Variational Quantum Circuits. *Adv. Neural Inf. Process Syst.* **36**, 18612–18627 (2022).
104. Grimsley, H. R., Mayhall, N. J., Barron, G. S., Barnes, E. & Economou, S. E. Adaptive, problem-tailored variational quantum eigensolver mitigates rough parameter landscapes and barren plateaus. *Npj Quantum Inf.* **9**, 19 (2023).
105. Cerezo, M., Sharma, K., Arrasmith, A. & Coles, P. J. Variational quantum state eigensolver. *Npj Quantum Inf.* **8**, 1–11 (2022).
106. Pesah, A. et al. Absence of barren plateaus in quantum convolutional neural networks. *Phys. Rev. X* **11**, 041011 (2021).
107. Liu, Z., Yu, L.-W., Duan, L.-M. & Deng, D.-L. The presence and absence of barren plateaus in tensor-network based machine learning. *Phys. Rev. Lett.* **129**, 270501 (2022).
108. Arrasmith, A., Holmes, Z., Cerezo, M. & Coles, P. J. Equivalence of quantum barren plateaus to cost concentration and narrow gorges. *Quantum Sci. Technol.* **7**, 045015 (2022).
109. Zhang, C., Bengio, S., Hardt, M., Recht, B. & Vinyals, O. Understanding deep learning (still) requires rethinking generalization. *Commun. ACM* **64,** 107–115 (2021).
110. Allen-Zhu, Z., Li, Y. & Song, Z. A convergence theory for deep learning via over-parameterization. In: *Proc. International Conference on Machine Learning* **36** (2019).
111. Allen-Zhu, Z., Li, Y. & Liang, Y. Learning and generalization in overparameterized neural networks, going beyond two layers. *Adv. Neural Inf. Process. Syst.* **33**, 6158–6169 (2019).
112. Buhai, R.-D., Halpern, Y., Kim, Y., Risteski, A. & Sontag, D. Empirical study of the benefits of overparameterization in learning latent variable models. In: *Proc. International Conference on Machine Learning* **37** (2020).
113. Banchi, L., Pereira, J. & Pirandola, S. Generalization in quantum machine learning: a quantum information standpoint. *PRX Quantum* **2**, 040321 (2021).
114. Caro, M. C. et al. Out-of-distribution generalization for learning quantum dynamics. *Nat. Commun.* **14**, 3751 (2023).
115. Du, Y., Tu, Z., Yuan, X. & Tao, D. Efficient measure for the expressivity of variational quantum algorithms. *Phys. Rev. Lett.* **128**, 080506 (2022).
116. Huang, H.-Y. et al. Power of data in quantum machine learning. *Nat. Commun.* **12**, 1–9 (2021).
117. Shalev-Shwartz, S. & Ben-David, S. Understanding Machine Learning: From Theory to Algorithms (Cambridge University Press, 2014).
118. Hajek, B. & Raginsky, M. Ece 543: Statistical learning theory. http://maxim.ece.illinois.edu/teaching/SLT/ (2021).
119. Thanasilp, S., Wang, S., Cerezo, M. & Holmes, Z. Exponential concentration and untrainability in quantum kernel methods. Preprint at https://arxiv.org/abs/2208.11060 (2022).
120. Erdos, P. & Renyi, A. On random graphs i. *Publ. Math. Debrecen* **6**, 18 (1959).
121. Raussendorf, R., Browne, D. E. & Briegel, H. J. Measurement-based quantum computation on cluster states. *Phys. Rev. A* **68**, 022312 (2003).
122. Hein, M., Eisert, J. & Briegel, H. J. Multiparty entanglement in graph states. *Phys. Rev. A* **69**, 062311 (2004).
123. Cheng, R. Quantum geometric tensor (fubini-study metric) in simple quantum system: a pedagogical introduction. Preprint at https://arxiv.org/abs/1012.1337 (2010).
124. Meyer, J. J. Fisher information in noisy intermediate-scale quantum applications. *Quantum* **5**, 539 (2021).
125. Larocca, M., Calzetta, E. & Wisniacki, D. A. Exploiting landscape geometry to enhance quantum optimal control. *Phys. Rev. A* **101**, 023410 (2020).
126. Larocca, M., Calzetta, E. & Wisniacki, D. Fourier compression: a customization method for quantum control protocols. *Phys. Rev. A* **102**, 033108 (2020).
127. Zhu, C., Byrd, R. H., Lu, P. & Nocedal, J. Algorithm 778: L-bfgs-b: Fortran subroutines for large-scale bound-constrained optimization. *ACM Trans. Math. Softw. (TOMS)* **23**, 550–560 (1997).
128. Nakaji, K. & Yamamoto, N. Expressibility of the alternating layered ansatz for quantum computation. *Quantum* **5**, 434 (2021).
129. Anschuetz, E. R., Bauer, A., Kiani, B. T. & Lloyd, S. Efficient classical algorithms for simulating symmetric quantum systems. *Quantum* **7**, 1189 (2023).
130. Kandala, A. et al. Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets. *Nature* **549**, 242–246 (2017).
131. Cincio, L., Subaşı, Y., Sornborger, A. T. & Coles, P. J. Learning the quantum algorithm for state overlap. *New J. Physical.* **20**, 113022 (2018).
132. Pérez-Salinas, A., Cervera-Lierta, A., Gil-Fuster, E. & Latorre, J. I. Data re-uploading for a universal quantum classifier. *Quantum* **4**, 226 (2020).
133. Serre, J.-P. et al. *Linear Representations of Finite Groups*, vol. 42 (Springer, 1977).
134. Fulton, W. & Harris, J. *Representation Theory: A First Course* (Springer, 1991).
135. Sagan, B. *The Symmetric Group: Representations, Combinatorial Algorithms, and Symmetric Functions*, vol. 203 (Springer Science & Business Media, 2001).
136. Knapp, A. W. *Representation Theory of Semisimple Groups: an Overview Based on Examples* (Princeton University Press, Princeton, 2001).
137. Goodman, R. & Wallach, N. R. *Symmetry, Representations, and Invariants*, vol. 255 (Springer, 2009).

138. Zeier, R. & Schulte-Herbrüggen, T. Symmetry principles in quantum systems theory. *J. Math. Phys.* **52**, 113510 (2011).
139. Kazi, S., Larocca, M. & Cerezo, M. On the universality of s_n-equivariant k-body gates. Preprint at https://arxiv.org/abs/2303.00728 (2023).
140. Ragone, M. et al. A unified theory of barren plateaus for deep parametrized quantum circuits. Preprint at https://arxiv.org/abs/2309.09342 (2023).
141. Puchala, Z. & Miszczak, J. A. Symbolic integration with respect to the Haar measure on the unitary groups. *Bull. Pol. Acad. Sci. Tech. Sci.* **65**, 21–27 (2017).
142. Zhang, B., Sone, A. & Zhuang, Q. Quantum computational phase transition in combinatorial problems. *Npj Quantum Inf.* **8**, 1–11 (2022).
143. Fontana, E. et al. The adjoint is all you need: Characterizing barren plateaus in quantum ansätze. Preprint at https://arxiv.org/abs/2309.07902 (2023).
144. Fan, J., Yang, Z. & Yu, M. Understanding implicit regularization in over-parameterized single index model. *J. Am. Stat. Assoc.* **118**, 1–14 (2022).
145. Du, S. S., Zhai, X., Poczos, B. & Singh, A. Gradient descent provably optimizes over-parameterized neural networks. In: *Proc. International Conference on Learning Representations* (2019).
146. Brutzkus, A., Globerson, A., Malach, E. & Shalev-Shwartz, S. SGD learns over-parameterized networks that provably generalize on linearly separable data. In: *Proc. International Conference on Learning Representations* (2018).
147. Bartlett, P. L., Montanari, A. & Rakhlin, A. Deep learning: a statistical viewpoint. *Acta Numer.* **30**, 87–201 (2021).
148. Fukumizu, K. A regularity condition of the information matrix of a multilayer perceptron network. *Neural Netw.* **9**, 871–879 (1996).
149. Liu, M. & Zhang, H. H. Overparameterization in the seminonparametric density estimation. *Econ. Lett.* **60**, 11–18 (1998).
150. RoyChowdhury, A., Sharma, P., Learned-Miller, E. & Roy, A. Reducing duplicate filters in deep neural networks. In: *Proc. Adv. Neural. Inf. Process. Sys. Workshop on Deep Learning* **1** (2017).
151. Vershynin, R. *High-Dimensional Probability: An Introduction with Applications in Data Science* (Cambridge University Press, 2018).
152. Dudley, R. M. *Uniform Central Limit Theorems* (Cambridge University Press, 1999).
153. Harrow, A. W. The church of the symmetric subspace. Preprint at https://arxiv.org/abs/1308.6595 (2013).

ACKNOWLEDGEMENTS

We thank Michael Ragone and Paolo Braccia for insightful discussion on geometric quantum machine learning. We also thank Felix Leditzky for discussion regarding Hermitian Young operators and Dylan Herman for useful questions and comments. L.S. was partially supported by the NSF Quantum Leap Challenge Institute for Hybrid Quantum Architectures and Networks (NSF Award 2016136). L.S. also acknowledges the support of LANL ASC Beyond Moore's Law project. M.L. acknowledges initial support by the Center for Nonlinear Studies at Los Alamos National Laboratory (LANL) and by the U.S. Department of Energy (DOE), Office of Science, Office of Advanced Scientific Computing Research, under the Accelerated Research in Quantum Computing (ARQC) program. F.S. acknowledges support by the Directed Research and Development (LDRD) program of LANL under project number 20220745ER. M.C. and M.L. were partially supported by Directed Research and Development (LDRD) program of LANL under project number 20210116DR and 20230049DR. This work was also supported by NSEC Quantum Sensing at LANL.

AUTHOR CONTRIBUTIONS

The project was conceived by M.L. and M.C. The manuscript was written by L.S., M.L., Q.T.N., F.S. and M.C. Theoretical results were derived by L.S., M.L., Q.T.N. and M.C. Numerical simulations were performed by L.S., M.L. and F.S.

COMPETING INTERESTS

The authors declare no competing interests.

ADDITIONAL INFORMATION

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s41534-024-00804-1.

Correspondence and requests for materials should be addressed to Louis Schatzki or M. Cerezo.

Reprints and permission information is available at http://www.nature.com/reprints

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.