RESEARCH ARTICLE SUMMARY

QUANTUM PHYSICS

Provably efficient machine learning for quantum many-body problems

Hsin-Yuan Huang*, Richard Kueng, Giacomo Torlai, Victor V. Albert, John Preskill

INTRODUCTION: Solving quantum many-body problems, such as finding ground states of quantum systems, has far-reaching consequences for physics, materials science, and chemistry. Classical computers have facilitated many profound advances in science and technology, but they often struggle to solve such problems. Scalable, fault-tolerant quantum computers will be able to solve a broad array of quantum problems but are unlikely to be available for years to come. Meanwhile, how can we best exploit our powerful classical computers to advance our understanding of complex quantum systems? Recently, classical machine learning (ML) techniques have been adapted to investigate problems in quantum many-body physics. So far, these approaches are mostly heuristic, reflecting the general paucity of rigorous theory in ML. Although they have been shown to be effective in some intermediate-size experiments, these methods are generally not backed by convincing theoretical arguments to ensure good performance.

RATIONALE: A central question is whether classical ML algorithms can provably outperform non-ML algorithms in challenging quantum many-body problems. We provide a concrete answer by devising and analyzing classical ML algorithms for predicting the properties of ground states of quantum systems. We prove

that these ML algorithms can efficiently and accurately predict ground-state properties of gapped local Hamiltonians, after learning from data obtained by measuring other ground states in the same quantum phase of matter. Furthermore, under a widely accepted complexity-theoretic conjecture, we prove that no efficient classical algorithm that does not learn from data can achieve the same prediction guarantee. By generalizing from experimental data, ML algorithms can solve quantum many-body problems that could not be solved efficiently without access to experimental data.

RESULTS: We consider a family of gapped local quantum Hamiltonians, where the Hamiltonian $H(x)$ depends smoothly on m parameters (denoted by x). The ML algorithm learns from a set of training data consisting of sampled values of x, each accompanied by a classical representation of the ground state of $H(x)$. These training data could be obtained from either classical simulations or quantum experiments. During the prediction phase, the ML algorithm predicts a classical representation of ground states for Hamiltonians different from those in the training data; ground-state properties can then be estimated using the predicted classical representation. Specifically, our classical ML algorithm predicts expectation values of products of local observables in the ground state, with a

small error when averaged over the value of x. The run time of the algorithm and the amount of training data required both scale polynomially in m and linearly in the size of the quantum system. Our proof of this result builds on recent developments in quantum information theory, computational learning theory, and condensed matter theory. Furthermore, under the widely accepted conjecture that nondeterministic polynomial-time (NP)–complete problems cannot be solved in randomized polynomial time, we prove that no polynomial-time classical algorithm that does not learn from data can match the prediction performance achieved by the ML algorithm.

In a related contribution using similar proof techniques, we show that classical ML algorithms can efficiently learn how to classify quantum phases of matter. In this scenario, the training data consist of classical representations of quantum states, where each state carries a label indicating whether it belongs to phase A or phase B. The ML algorithm then predicts the phase label for quantum states that were not encountered during training. The classical ML algorithm not only classifies phases accurately, but also constructs an explicit classifying function. Numerical experiments verify that our proposed ML algorithms work well in a variety of scenarios, including Rydberg atom systems, two-dimensional random Heisenberg models, symmetry-protected topological phases, and topologically ordered phases.

CONCLUSION: We have rigorously established that classical ML algorithms, informed by data collected in physical experiments, can effectively address some quantum many-body problems. These rigorous results boost our hopes that classical ML trained on experimental data can solve practical problems in chemistry and materials science that would be too hard to solve using classical processing alone. Our arguments build on the concept of a succinct classical representation of quantum states derived from randomized Pauli measurements. Although some quantum devices lack the local control needed to perform such measurements, we expect that other classical representations could be exploited by classical ML with similarly powerful results. How can we make use of accessible measurement data to predict properties reliably? Answering such questions will expand the reach of near-term quantum platforms. ∎
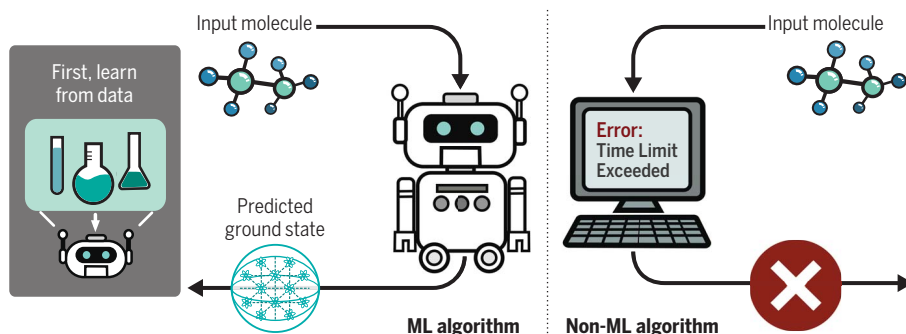
Classical algorithms for quantum many-body problems. Classical ML algorithms learn from training data, obtained from either classical simulations or quantum experiments. Then, the ML algorithm produces a classical representation for the ground state of a physical system that was not encountered during training. Classical algorithms that do not learn from data may require substantially longer computation time to achieve the same task.

S **READ THE FULL ARTICLE AT**
https://doi.org/10.1126/science.abk3333

QUANTUM PHYSICS

Provably efficient machine learning for quantum many-body problems

Hsin-Yuan Huang[1]*, Richard Kueng[2], Giacomo Torlai[3], Victor V. Albert[4], John Preskill[1,3]

Classical machine learning (ML) provides a potentially powerful approach to solving challenging quantum many-body problems in physics and chemistry. However, the advantages of ML over traditional methods have not been firmly established. In this work, we prove that classical ML algorithms can efficiently predict ground-state properties of gapped Hamiltonians after learning from other Hamiltonians in the same quantum phase of matter. By contrast, under a widely accepted conjecture, classical algorithms that do not learn from data cannot achieve the same guarantee. We also prove that classical ML algorithms can efficiently classify a wide range of quantum phases. Extensive numerical experiments corroborate our theoretical results in a variety of scenarios, including Rydberg atom systems, two-dimensional random Heisenberg models, symmetry-protected topological phases, and topologically ordered phases.

Solving quantum many-body problems, such as finding ground states of quantum systems, has far-reaching consequences for physics, materials science, and chemistry. Although classical computers have facilitated many profound advances in science and technology, they often struggle to solve such problems. Powerful methods, such as density functional theory (*1*, *2*), quantum Monte Carlo (*3–5*), and density-matrix renormalization group (*6*, *7*), have enabled solutions to certain restricted instances of many-body problems, but many general classes of problems remain outside the reach of even the most advanced classical algorithms.

Scalable, fault-tolerant quantum computers will be able to solve a broad array of quantum problems but are unlikely to be available for years to come. Meanwhile, how can we best exploit our powerful classical computers to advance our understanding of complex quantum systems? Recently, classical machine learning (ML) techniques have been adapted to investigate problems in quantum many-body physics (*8*, *9*) with promising results (*10–27*). So far, these approaches are mostly heuristic, reflecting the general paucity of rigorous theory in ML. Although they were shown to be effective in some intermediate-size experiments (*28–30*), these methods are generally not backed by convincing theoretical arguments to ensure good performance, particularly for problem instances where traditional classical algorithms falter.

In general, simulating quantum many-body physics is hard for classical computers because accurately describing an n-qubit quantum system may require an amount of classical data that is exponential in n. In prior work, this bottleneck has been addressed using classical shadows—succinct classical descriptions of quantum many-body states that can be used to accurately predict a wide range of properties with rigorous performance guarantees (*31*, *32*). Furthermore, this quantum-to-classical conversion technique can be readily implemented in various existing quantum experiments (*33–35*). Classical shadows create opportunities for addressing quantum problems using classical methods, such as ML. In this paper, we build on the classical shadow formalism and devise polynomial-time classical ML algorithms for quantum many-body problems that are supported by rigorous theory.

We consider two applications of classical ML, indicated in Fig. 1. The first application we examine is learning to predict classical representations of quantum many-body ground states. We consider a family of Hamiltonians, where the Hamiltonian $H(x)$ depends smoothly on m real parameters (denoted by x). The ML algorithm is trained on a set of training data consisting of sampled values of x, each accompanied by the corresponding classical shadow for the ground state $\rho(x)$ of $H(x)$. These training data could be obtained from either classical simulations or quantum experiments. During the prediction phase, the ML algorithm predicts a classical representation of $\rho(x)$ for values of x different from those in the training data. Ground-state properties can then be estimated using the predicted classical representation.

This learning algorithm is efficient, provided that the ground-state properties to be predicted do not vary too rapidly as a function of x. Suf-

ficient upper bounds on the gradient can be derived for any family of gapped, geometrically local Hamiltonians in any finite spatial dimension, as long as the property of interest is the expectation value of a sum of few-body observables. The conclusion is that any such property can be predicted with a small average error, where the amount of training data and the classical computation time are polynomial in m and linear in the system size. Furthermore, we show that classical algorithms that do not learn from data cannot make accurate predictions in polynomial time without violating widely accepted complexity-theoretic conjectures. Together, we rigorously establish the advantage of ML algorithms with data over those without data (*36*) in a physically relevant task.

The classical ML algorithm could generalize from training data that are obtained either through quantum experiments or classical simulations; the same rigorous performance guarantees apply in either case. If the training data are obtained from quantum experiments, the rigorous result shows that classical ML can explore and predict properties of new physical systems that are challenging to prepare and measure in the laboratory. Even if the experimentalists only have limited measurement capability, such as being able to measure a specific property of $\rho(x)$, the theorem established in this work immediately implies that a classical ML model can predict that specific property accurately. If the training data are generated classically, it could be more efficient and more accurate to use the ML model to predict properties for new values of the input x rather than doing new simulations, which could be computationally very demanding. Promising insights into quantum many-body physics are already being obtained using classical ML based on classical simulation data (*10*, *12*, *14*, *17*, *19*, *20*, *23–25*, *37*, *38*). Our rigorous analysis identifies general conditions that guarantee the success of classical ML models and elucidate the advantages of classical ML models over non-ML algorithms, which do not learn from data. These results enhance the prospects for interpretable ML techniques (*38–40*) to further shed light on quantum many-body physics.

In the second application we examine, the goal is to classify quantum states of matter into phases (*41*) in a supervised learning scenario. Suppose that during training we are provided with sample quantum states that carry labels indicating whether each state belongs to phase A or phase B. Our goal is to classify the phase for new quantum states that were not encountered during training. We assume that, during both the training and classification stages, each quantum state is represented by its classical shadow, which could be obtained either from a classical computation or from an experiment on a quantum device. The classical ML model, then, trains on labeled classical shadows

[1]Institute for Quantum Information and Matter and Department of Computing and Mathematical Sciences, Caltech, Pasadena, CA, USA. [2]Institute for Integrated Circuits, Johannes Kepler University, Linz, Austria. [3]AWS Center for Quantum Computing, Pasadena, CA, USA. [4]Joint Center for Quantum Information and Computer Science, National Institute of Standards and Technology and University of Maryland, College Park, MD, USA.
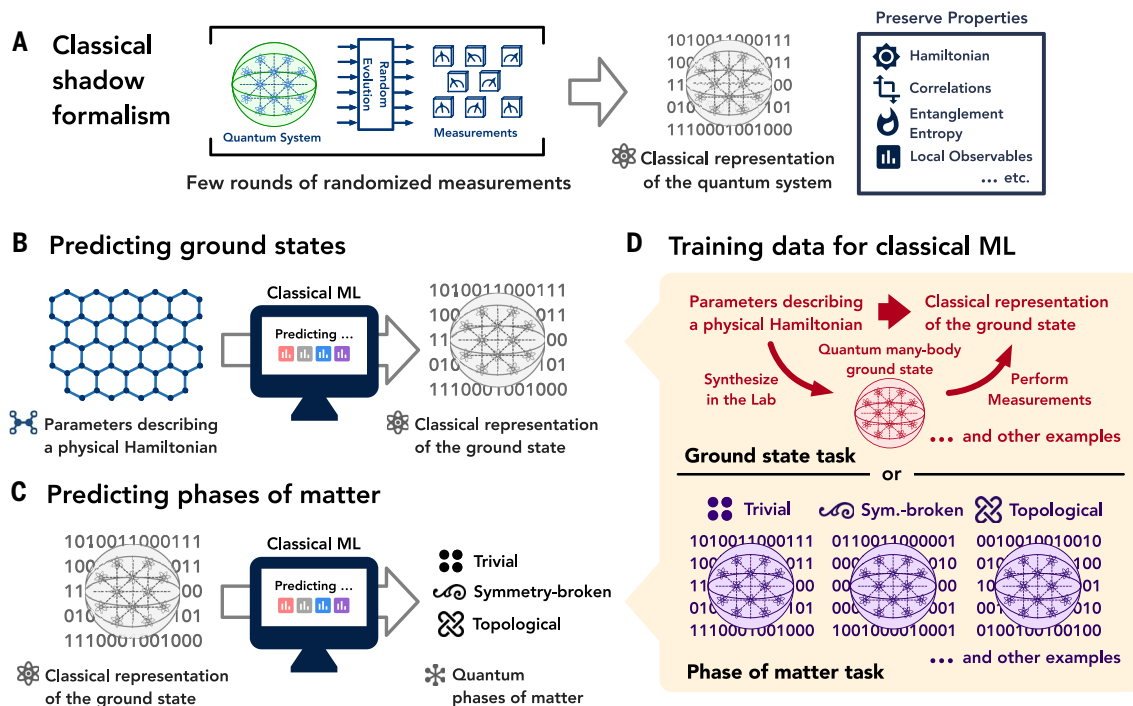*Corresponding author. Email: hsinyuan@caltech.edu

Fig. 1. Central concepts. (**A**) Efficient quantum-to-classical conversion. The classical shadow of a quantum state, constructed by measuring very few copies of the state, can be used to predict many properties of the state with a rigorous performance guarantee. (**B**) Predicting ground-state properties. After training on data obtained in quantum experiments, a classical ML model predicts a classical representation of the ground state ρ(x) of the Hamiltonian H(x) for parameters x spanning the entire phase. This representation yields estimates of the properties of ρ(x), avoiding the need to run exhaustive classical computations or quantum experiments. (**C**) Classifying quantum phases. After training, a classical ML model receives a classical representation of a quantum state and classifies the phase from which the state was drawn. (**D**) Training data. For predicting ground states, the classical ML model receives a classical representation of ρ(x) for each value of x sampled during training. For predicting quantum phases of matter, the training data consist of classical representations of quantum states accompanied by labels identifying the phase to which each state belongs.

and learns to predict labels for new classical shadows.

We assume that the A and B phases can be distinguished by a nonlinear function of marginal density operators of subsystems of constant size. This assumption is reasonable because we expect the phase to be revealed in subsystems that are larger than the correlation length but do not depend on the total system size. We show that if such a function exists, a classical ML model can learn to distinguish the phases using an amount of training data and classical processing that are polynomial in the system size. We do not need to know anything about this nonlinear function in advance, apart from its existence.

Here, we review the classical shadow formalism (31) and use this formalism to derive rigorous guarantees for ML algorithms in predicting ground-state properties and classifying quantum phases of matter. We also describe numerical experiments in a wide range of physical systems to support our theoretical results.

Constructing efficient classical representations of quantum systems

We begin with an overview of the randomized measurement toolbox (31, 32, 42–45), relegating further details to section S1 (46). We approximate an n-qubit quantum state ρ by per-

forming randomized single-qubit Pauli measurements on T copies of ρ. That is, we measure every qubit of the unknown quantum state ρ in a random Pauli basis X, Y, or Z to yield a measurement outcome of ±1. Collapse of the wave function implies that this measurement procedure transforms ρ into a random pure product state $|s^{(t)}\rangle = \otimes_{i=1}^{n}|s_i^{(t)}\rangle$, where $|s_i^{(t)}\rangle \in \{|0\rangle, |1\rangle, |+\rangle, |-\rangle, |i+\rangle, |i-\rangle\}$ are eigenstates of the selected Pauli matrices. Performing one randomized measurement grants us classical access to one such snapshot. Performing a total of T randomized measurements grants us access to an entire collection $S_T(\rho) = \left\{ |s_i^{(t)}\rangle : i \in \{1, ..., n\}, t \in \{1, ..., T\} \right\}$. Each element is a highly structured single-qubit pure state, and there are nT of them in total. So, 3nT bits suffice to store the entire collection in classical memory. The randomized measurements can be performed in actual physical experiments or through classical simulations. Resulting data can then be used to approximate the underlying n-qubit state ρ

$$\rho \approx \sigma_T(\rho) = \frac{1}{T}\sum_{t=1}^{T}\sigma_1^{(t)} \otimes ... \otimes \sigma_n^{(t)}$$

$$\text{where } \sigma_i^{(t)} = 3\left|s_i^{(t)}\right\rangle\left\langle s_i^{(t)}\right| - I \quad (1)$$

and I denotes the 2 × 2 identity matrix. This classical shadow representation (31, 32) exactly reproduces the global density matrix in the limit T→∞, but $T = O[\text{const}^r \log(n)/\varepsilon^2]$ already provides an ε-accurate approximation of all reduced–r-body density matrices (in trace distance). This, in turn, implies that we can use σ_T(ρ) to predict any function that depends on only reduced-density matrices, such as expectation values of (sums of) local observables and (sums of) entanglement entropies of small subsystems. Classical storage and postprocessing costs also remain tractable in this regime. To summarize, the classical shadow formalism equips us with an efficient quantum-to-classical converter that allows classical machines to efficiently and reliably estimate subsystem properties of any quantum state ρ.

Predicting ground states of quantum many-body systems

We consider the task of predicting ground-state representations of quantum many-body Hamiltonians in finite spatial dimensions. Suppose that a family of geometrically local, n-qubit Hamiltonians $\{H(x) : x \in [-1, 1]^m\}$ is parameterized by a classical variable x. That is, H(x) smoothly maps a bounded m-dimensional

vector x (parametrization) to a Hermitian matrix of size $2^n \times 2^n$ (n-qubit Hamiltonian). We do not impose any additional structure on this mapping; in particular, we do not assume knowledge about how the physical Hamiltonian depends on the parameterization. The goal is to learn a model $\hat{\sigma}(x)$ that can predict properties of the ground state $\rho(x)$ associated with the Hamiltonian. This problem arises in many practical scenarios. Suppose diligent experimental effort has produced experimental data for ground-state properties of various physical systems. We would like to use these data to train an ML model that predicts ground-state representations of hitherto unexplored physical systems.

An ML algorithm with rigorous guarantee

We will prove that a classical ML algorithm can predict classical representations of ground states after training on data belonging to the same quantum phase of matter. Formally, we consider a smooth family of Hamiltonians $H(x)$ with a constant spectral gap. During the training phase of the ML algorithm, many values of x are randomly sampled, and for each sampled x, the classical shadow of the corresponding ground state $\rho(x)$ of $H(x)$ is provided, either by classical simulations or quantum experiments. The full training data of size N are given by $\{x_l \rightarrow \sigma_T[\rho(x_l)]\}_{l=1}^N$, where T is the number of randomized measurements in the construction of the classical shadows at each value of x_l.

We train classical ML models using the size-N training data, such that when given the input x_l, the ML model can produce a classical representation $\hat{\sigma}(x)$ that approximates $\sigma_T[\rho(x_l)]$. During prediction, the classical ML model produces $\hat{\sigma}(x)$ for values of x different from those in the training data. Although $\hat{\sigma}(x)$ and $\sigma_T[\rho(x_l)]$ classically represent exponentially large density matrices, the training and prediction can be done efficiently on a classical computer using various existing classical ML models, such as neural networks with large hidden layers (47–50) and kernel methods (51, 52). In particular, the predicted output of the trained classical ML models can be written as the extrapolation of the training data using a learned metric $\kappa(x, x_l) \in R$

$$\hat{\sigma}(x) = \frac{1}{N} \sum_{l=1}^N \kappa(x, x_l) \sigma_T[\rho(x_l)] \qquad (2)$$

For example, prediction using a trained neural network with large hidden layers (46) is equivalent to using the metric $\kappa(x, x_l) = \sum_{l'=1}^N f^{(NTK)}(x, x_{l'})(F^{-1})_{l'l}$, where $f^{(NTK)}(x, x')$ is the neural tangent kernel (47) corresponding to the neural network and $F_{l'l} = f^{(NTK)}(x_l, x_l)$ [see section S3 (46) for more discussion]. The

ground-state properties are then estimated using these predicted classical representations $\hat{\sigma}(x)$. Specifically, $f_O(x) = tr[O\rho(x)]$ can be predicted efficiently whenever O is a sum of few-body operators.

To derive a provable guarantee, we consider the simple metric $\kappa(x, x_l) = \sum_{k \in Z^m, \|k\|_2 \leq \Lambda} \cos[\pi k \cdot (x - x_l)]$ with cutoff Λ, which we refer to as the l_2-Dirichlet kernel. We prove that the prediction will be accurate and efficient if the function $f_O(x)$ does not vary too rapidly when x changes in any direction. Sufficient upper bounds on the gradient magnitude of $f_O(x)$ can be derived using quasi-adiabatic continuation (53, 54).

Under the l_2-Dirichlet kernel, the classical ML model is equivalent to learning a truncated Fourier series to approximate the function $f_O(x)$. The parameter Λ is a cutoff for the wave number k that depends on (upper bounds on) the gradient of $f_O(x)$. Using statistical analysis, one can guarantee that $E_x |tr[O\hat{\sigma}(x)] - f_O(x)|^2 \leq \varepsilon$ as long as the amount of training data obeys $N = m^{O(1/\varepsilon)}$ in the $m \rightarrow \infty$ limit. The conclusion is that any such $f_O(x)$ can be predicted with a small constant average error, where the amount of training data and the classical computation time are polynomial in m and at most linear in the system size n. Moreover, the training data need only contain a single classical shadow snapshot at each point x_l in the parameter space (i.e., $T = 1$). An informal statement of the theorem is given below; we explain the proof strategy in section S5 and provide more details in section S6 (46). We also discuss how one could generalize the proof to long-range interacting systems, electronic Hamiltonians, and other settings, including when one cannot perform classical shadow tomography (31), in section S6.2 (46).

Theorem 1 (learning to predict ground-state representations; informal)

For any smooth family of Hamiltonians $\{H(x) : x \in [-1, 1]^m\}$ in a finite spatial dimension with a constant spectral gap, the classical ML algorithm can learn to predict a classical representation of the ground state $\rho(x)$ of $H(x)$ that approximates few-body reduced-density matrices up to a constant error ε when averaged over x. The required training data size N and computation time are polynomial in m and linear in the system size n.

Though formally efficient in the sense that N scales polynomially with m for any fixed approximation error ε, the required amount of training data scales badly with ε. This unfortunate scaling is not a shortcoming of the considered ML algorithm, but a necessary feature. In section S7 (46), we show that the data size and time complexity cannot be improved further without making stronger assumptions about the class of gapped local Hamiltonians.

However, in cases of practical interest, the Hamiltonian may obey restrictions such as translational invariance or graph structure that can be exploited to obtain better results. Incorporating these restrictions can be achieved by using a suitable $\kappa(x, x_l)$, such as one that corresponds to a large-width convolutional neural network (CNN) (48) or a graph neural network (49). Rigorously establishing that neural network–based ML algorithms can achieve improved prediction performance and efficiency for particular classes of Hamiltonians requires further investigation.

Computational hardness for non-ML algorithms

In the following proposition, we show that a classical polynomial-time algorithm that does not learn from data cannot achieve the same guarantee in estimating ground-state properties without violating the widely believed conjecture that nondeterministic polynomial-time (NP)–complete problems cannot be solved in randomized polynomial time. This proposition is a corollary of standard complexity-theoretic results (55, 56). See section S8 (46) for the detailed statement and proof.

Proposition 1 (informal)

Consider a randomized polynomial-time classical algorithm A that does not learn from data. Suppose for any smooth family of two-dimensional (2D) Hamiltonians $\{H(x) : x \in [-1, 1]^m\}$ with a constant spectral gap, A can efficiently compute expectation values of one-body observables in the ground state $\rho(x)$ of $H(x)$ up to a constant error when averaged over x. Then, there is a randomized classical algorithm that can solve NP-complete problems in polynomial time.

It is instructive to observe that a classical ML algorithm with access to data can perform tasks that cannot be achieved by classical algorithms that do not have access to data. This phenomenon is studied in (36), where it is shown that the complexity class defined by classical algorithms that can learn from data is strictly larger than the class of classical algorithms that do not learn from data. (The data can be regarded as a restricted form of randomized advice string.) We caution that obtaining the data to train the classical ML model could be challenging. However, if we focus only on data that could be efficiently generated by quantum-mechanical processes, it is still possible that a classical ML algorithm that learns from data could be more powerful than classical computers. In section S8 (46), we present a contrived family of Hamiltonians that establishes this claim based on the (classical) computational hardness of factoring.

Classifying quantum phases of matter

Classifying quantum phases of matter is another important application of ML to physics.

We will consider this classification problem in the case where quantum states are succinctly represented by their classical shadows. For simplicity, we consider the classification of two phases (denoted A and B), but the analysis naturally generalizes to classifying any number of phases.

ML algorithms

We envision training a classical ML algorithm with classical shadows, where each classical shadow carries a label y indicating whether it represents a quantum state ρ from phase A [$y(\rho) = 1$] or phase B [$y(\rho) = -1$]. We want to show that a suitably chosen classical ML algorithm can learn to efficiently classify the phase for new classical shadows beyond those encountered during training. Following a strategy standard in learning theory, we consider a classical ML algorithm that maps each classical shadow to a corresponding feature vector in a high-dimensional feature space and then attempts to find a hyperplane that separates feature vectors in the A phase from feature vectors in the B phase. The learning is efficient if the geometry of the feature space is efficiently computable and if the feature map is sufficiently expressive. Thus, our task is to construct a feature map with the desired properties.

In the simpler task of classifying symmetry-breaking phases, there is typically a local order parameter $O = \sum_i O_i$ given as a sum of r-body observables for some $r > 0$ that satisfies

$$tr(O\rho) \geq 1, \forall \rho \in \text{phase } A, tr(O\rho) \leq -1, \forall \rho \in \text{phase } B \qquad (3)$$

Under this criterion, the classification function may be chosen to be $y(\rho) = sign[tr(O\rho)]$. Hence, classifying symmetry-breaking phases can be achieved by finding a hyperplane that separates the two phases in the high-dimensional feature space that subsumes all r-body reduced-density matrices of the quantum state ρ. The feature vector consisting of all r-body reduced-density matrices of the quantum state ρ can be accurately reconstructed from the classical shadow representation $S_T(\rho)$ when T is sufficiently large.

Finding a suitable choice of hyperplane in the feature space can be cast as a convex optimization problem known as the soft-margin support vector machine (SVM), discussed in more detail in section S10.1 (46). With a sufficient amount of training data, the hyperplane found by the classical ML model will generalize so that the phase $y(\rho)$ can be predicted accurately for a previously unseen quantum state ρ. The classical ML model is not merely a black box; it also discovers the order parameter (encoded by the hyperplane), guiding physicists toward a deeper understanding of the phase structure.

For more exotic quantum phases of matter, such as topologically ordered phases, the above classical ML model no longer suffices. The topological phase of a state is invariant under a constant-depth quantum circuit, and a phase containing the product state $|0\rangle^{\otimes n}$ is called the trivial phase. Using these notions, we can prove that no observable—not even one that acts on the entire system—can be used to distinguish between two topological phases. The proof, given in section S9 (46), uses the observation that random single-qubit unitaries can confuse any global or local order parameter.

Proposition 2

Consider two distinct topological phases A and B (one of the phases could be the trivial phase). No observable O exists such that

$$tr(O\rho) > 0, \forall \rho \in \text{phase } A, tr(O\rho) \leq 0, \\ \forall \rho \in \text{phase } B \qquad (4)$$

Although this proposition implies that no linear function $tr(O\rho)$ can be used to classify topologically ordered phases, it does not exclude nonlinear functions, such as quadratic functions $tr(O\rho \otimes \rho)$, degree-d polynomials $tr(O\rho^{\otimes d})$, and more general analytic functions. For example, it is known that the topological entanglement entropy (57, 58), a nonlinear function of ρ, can be used to classify a wide variety of topologically ordered phases. For this purpose, it suffices to consider a subsystem whose size is large compared with the correlation length of the state but is independent of the total size of the system. The correlation length in the ground state of a local Hamiltonian increases when the spectral gap between the ground state and the first excited state becomes smaller (59). On the other hand, a linear function on the full system will fail even with constant correlation length.

To learn nonlinear functions, we need a more expressive ML model. For this purpose, we devise a powerful feature map that takes the classical shadow $S_T(\rho)$ of the quantum state ρ to a feature vector that includes arbitrarily-large r-body reduced-density matrices, as well as an arbitrarily-high–degree polynomial expansion

$$\varphi^{(\text{shadow})}[S_T(\rho)]$$
$$= \lim_{D,R\to\infty} \oplus_{d=0}^{D} \sqrt[D]{\frac{\tau^d}{d!}} \left(\oplus_{r=0}^{R} \sqrt[R]{\frac{1}{r!}\left(\frac{\gamma}{n}\right)^r} \oplus_{i_1=1}^{n} \cdots \right.$$
$$\left. \oplus_{i_r=1}^{n} vec\left[\frac{1}{T}\sum_{t=1}^{T} \otimes_{l=1}^{r} \sigma i_l^{(t)}\right] \right)^{\otimes d} \qquad (5)$$

where $\tau, \gamma > 0$ are hyperparameters. The direct sum $\oplus_{r=0}^{R}$ is a concatenation of all r-body reduced-density matrices, and the other direct sum $\oplus_{d=0}^{D}$ subsumes all degree-d polynomial expansions. The computational cost of finding a hyperplane in feature space that separates the training data into two classes is dominated by the cost of computing inner products between feature vectors. The inner product $\langle \varphi^{(\text{shadow})}[S_T(\rho)], \varphi^{(\text{shadow})}[S_T(\tilde{\rho})]\rangle$ can be analytically computed by reorganizing the direct sums, writing it as a double series, and wrapping both series into an exponential, which gives

$$k^{(\text{shadow})}[S_T(\rho), S_T(\tilde{\rho})]$$
$$= \exp\left\{ \frac{\tau}{T^2} \sum_{t,t'=1}^{T} \exp\left[\frac{\gamma}{n}\sum_{i=1}^{n} tr\left(\sigma_i^{(t)} \tilde{\sigma}_i^{(t)}\right)\right] \right\} \qquad (6)$$

where $S_T(\rho)$ and $S_T(\tilde{\rho})$ are classical shadow representations of ρ and $\tilde{\rho}$, respectively. The computation time for the inner product is $O(nT^2)$, linear in the system size n and quadratic in T, the number of copies of each quantum state that are measured to construct the classical shadow.

Rigorous guarantee

By statistical analysis, we can establish a rigorous guarantee for the classical ML model $\langle \alpha, \varphi^{(\text{shadow})}[S_T(\rho)]\rangle$, where α is the trainable vector defining the classifying hyperplane. The result is the following theorem, proven in section S10 (46).

Theorem 2 (classifying quantum phases of matter; informal)

If there is a nonlinear function of few-body reduced-density matrices that classifies phases, then the classical algorithm can learn to classify these phases accurately. The required amount of training data and computation time scale polynomially in system size.

If there is an efficient procedure based on few-body reduced-density matrices for classifying phases, the proposed ML algorithm is guaranteed to find the procedure efficiently. This includes local order parameters for classifying symmetry-breaking phases and topological entanglement entropy in a sufficiently large local region for partially classifying topological phases (57, 58). We expect that, to classify topological phases accurately, the classical ML model will need access to local regions that are sufficiently large compared with the correlation length, and as we approach the phase boundary, the correlation length increases. As a result, the classifying function for topological phases may depend on r-body subsystems with a larger r, and the amount of training data and computation time required would increase accordingly. The classical ML model not only classifies phases accurately but also constructs a classifying function explicitly.

Our classical ML model may also be useful for classifying and understanding symmetry-protected topological (SPT) phases. SPT phases are characterized much like topological phases but with the additional constraint that all structures involved (states, Hamiltonians, and quantum circuits) respect a particular symmetry. It

is reasonable to expect that an SPT phase can be identified by examining reduced-density matrices on constant-size regions (*60–65*), where the size of the region is large compared with the correlation length. The existence of classifying functions based on reduced matrices has been rigorously established in some cases (*66–73*). In section S12 (*46*), we prove that the ML algorithm is guaranteed to efficiently classify a class of gapped spin-1 chains in one dimension. For more general SPT phases, the ML algorithm should be able to corroborate known classification schemes, determine new and potentially more-compact classifiers, and shed light on interacting SPT phases in two or more dimensions for which complete classification schemes have not yet been firmly established.

The hypothesis of theorem 2, stating that phases can be recognized by inspecting regions of constant size independent of the total system size, is particularly plausible for gapped phases, but it might apply to some gapless phases as well. Our classical ML model would be able to efficiently classify such gapless phases. On the other hand, the contrapositive of theorem 2 asserts that if the classical ML model is not able to distinguish between two distinct gapless phases, then nonlocal data are required to characterize at least one of those phases.

Numerical experiments

We have conducted numerical experiments assessing the performance of classical ML algorithms in some practical settings. The results demonstrate that our theoretical claims carry over to practice, with the results sometimes turning out even better than our guarantees suggest.

Predicting ground-state properties

For predicting ground states, we consider classical ML models encompassed by Eq. 2. We examine various metrics $\kappa(x,x_l)$ equivalent to training neural networks with large hidden layers (*47*, *50*) or training kernel methods (*51*, *74*). We find the best ML model and the hyperparameters using a validation set to minimize root mean square error (RMSE) and report the predictions on a test set. The full details of the models and hyperparameters, as well as their comparisons, are given in sections S4.2 and S4.3 (*46*).

Rydberg atom chain

Our first example is a system of trapped Rydberg atoms (*75*, *76*), a programmable and highly controlled platform for Ising-type quantum simulations (*77–82*). Following (*77*), we consider a 1D array of $n = 51$ atoms, with each atom effectively described as a two-level system composed of a ground state $|g\rangle$ and a highly excited Rydberg state $|r\rangle$. The atomic chain is characterized by a Hamiltonian $H(x)$ (given in Fig. 2A) whose parameters are the laser detuning $x_1 = \Delta/\Omega$ and the interaction range $x_2 = R_b/a$. The phase diagram (Fig. 2B) features a disordered phase and several broken-symmetry phases, stemming from the competition between the detuning and the Rydberg blockade (arising from the repulsive van der Waals interactions).

We trained a classical ML model using 20 randomly chosen values of the parameter $x =$ (x_1,x_2); these values are indicated by gray circles in Fig. 2B. For each such x, an approximation to the exact ground state was found using density matrix renormalization group (DMRG) (*6*) based on the formalism of matrix product states (MPSs) (*83*). For each MPS, we performed $T = 500$ randomized Pauli measurements to construct a classical shadow. The classical ML model then predicted classical representations at the testing points in the parameter space, and these predicted classical representations were used to estimate expectation values of local observables at the testing points.

Predictions for expectation values of Pauli operators Z_i and X_i at the testing points are shown in Fig. 2C and were found to agree well with exact values obtained from the DMRG computation of the ground state at the testing points. Additional predictions can be found in section S4.1 (*46*). Also shown are results from a more-naïve procedure, in which properties are predicted using only the data at the point in the training set that is closest to the testing point. The naïve procedure predicts poorly, illustrating that the considered classical ML model effectively leverages the data from multiple points in the training set.

This example corroborates our expectation that classical machines can learn to efficiently predict ground-state representations. An important caveat is that the rigorous guarantee in theorem 1 applies only when the training points and the testing points are sampled from the same phase, whereas in this example, the training data include values of x from
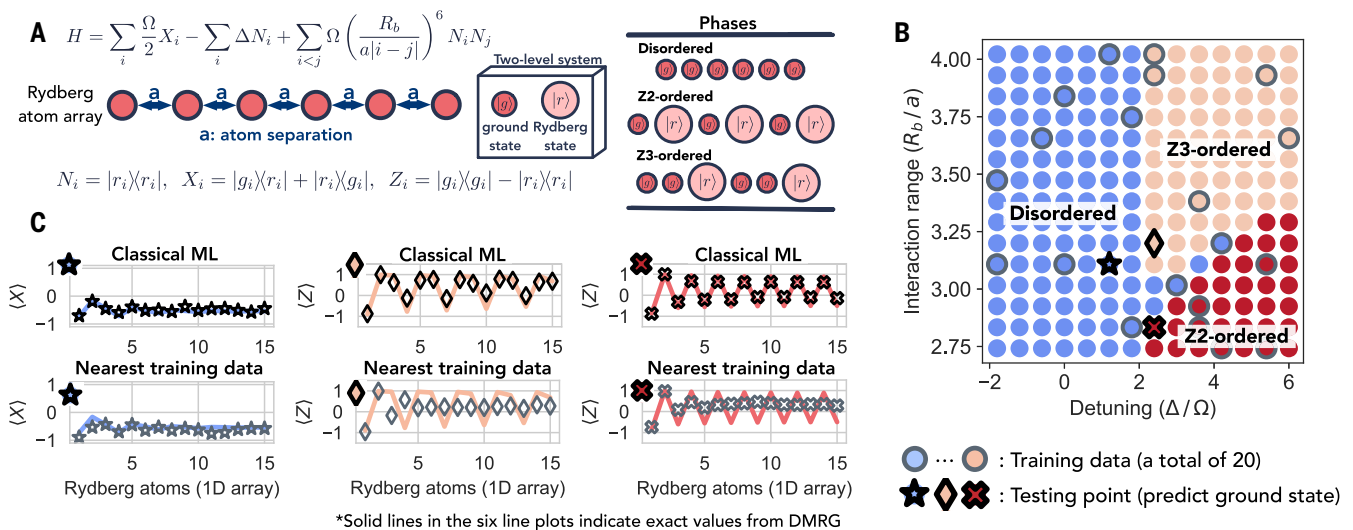
Fig. 2. Numerical experiment for predicting ground-state properties in a 1D Rydberg atom system with 51 atoms. (**A**) Hamiltonian and the illustrations of the Rydberg array geometry and the three resulting phases. (**B**) Phase diagram. The system's three distinct phases (*77*) are characterized by two order parameters (for Z_2 and Z_3 orders). Training data are enclosed by gray circles, and three specific testing points are indicated by the star, diamond, and cross symbols, respectively. (**C**) Local expectation values. We use classical ML (the best model is selected from a set of ML models) to predict the expectation values of Pauli operators X_i and Z_i for each atom at the three testing points. (Top) Results for the first 15 atoms. (Bottom) Predictions obtained from the training data nearest to the testing points. The markers denote predicted values, whereas the solid lines denote exact values obtained from DMRG. Additional predictions are shown in section S4.1 (*46*).

three different phases. Nevertheless, our numerics show that classical machines can still learn to predict well.

2D antiferromagnetic Heisenberg model

Our next example is the 2D antiferromagnetic Heisenberg model. Spin-½ particles (i.e., qubits) occupy sites on a square lattice, and for each pair (ij) of neighboring sites, the Hamiltonian contains a term $J_{ij}(X_iX_j + Y_iY_j + Z_iZ_j)$, where the couplings $\{J_{ij}\}$ are uniformly sampled from the interval $[0, 2]$. The parameter x is a list of all J_{ij} couplings; hence, in this case, the dimension of the parameter space is $m = O(n)$, where n is the number of qubits. The Hamiltonian $H(x)$ on a 5×5 lattice is shown in Fig. 3A.

We trained a classical ML model using 90 randomly chosen values of the parameter $x = \{J_{ij}\}$. For each such x, the exact ground state was found using DMRG, and we simulated $T = 500$ randomized Pauli measurements to construct a classical shadow. The classical ML model predicted the classical representation at new values of x, and we used the predicted classical representation to estimate a two-body correlation function, the expectation value of $C_{ij} = \frac{1}{3}(X_iX_j + Y_iY_j + Z_iZ_j)$, for each pair of qubits (ij). In Fig. 3B, the predicted and actual values of the correlation function are displayed for a particular value of x, showing reasonable agreement.

Figure 3C shows the prediction performance for all pairs of spins and for variable system sizes. Each red point in the plot represents the RMSE in the correlation function estimated using our predicted classical representation for a particular pair of spins and averaged over sampled values of x. For comparison, each blue point is the RMSE when the correlation function is predicted using the classical shadow obtained by measuring the actual ground state $T = 500$ times. For most correlation functions, the prediction error achieved by the best classical ML model is comparable to the error achieved by measuring the actual ground state.

Classifying quantum phases of matter

For classifying quantum phases of matter, we consider an unsupervised classical ML model that constructs an infinite-dimensional nonlinear feature vector for each quantum state ρ by applying the map $\varphi^{(\text{shadow})}$ in Eq. 5 with $\tau, \gamma = 1$ to the classical shadow $S_T(\rho)$ of the quantum state ρ. We then perform a principal components analysis (PCA) (84) in the infinite-dimensional nonlinear feature space. The low-dimensional subspace found by PCA in the nonlinear feature space corresponds to a nonlinear low-dimensional manifold in the original quantum state space. This method is efficient using the shadow kernel $k^{(\text{shadow})}$ given in Eq. 6 and the kernel PCA procedure (85). Details are given in sections S4.4. and S4.5 (46).

Bond-alternating XXZ model

We begin by considering the bond-alternating XXZ model with $n = 300$ spins. The Hamiltonian is given in Fig. 4A; it encompasses the bond-alternating Heisenberg model (δ = 1) and the bosonic version of the Su-Schrieffer-Heeger model (δ = 0) (86). The phase diagram in Fig. 4B is obtained by evaluating the partial reflection many-body topological invariant (62, 87). There are three distinct phases: trivial, SPT, and symmetry broken.

For each value of J and δ considered, we construct the exact ground state using DMRG and find its classical shadow by performing randomized Pauli measurement $T = 500$ times.

We then consider a 2D principal subspace of the infinite-dimensional nonlinear feature space found by the unsupervised ML based on the shadow kernel, which is visualized in Fig. 4, C and D. We can clearly see that the different phases are well separated in the principal subspace. This shows that even without any phase labels on the training data, the ML model can classify the phases accurately. Hence, when trained with only a small amount of labeled data, the ML model will be able to correctly classify the phases as guaranteed by theorem 2.

Distinguishing a topological phase from a trivial phase

We consider the task of distinguishing the toric code topological phase from the trivial phase in a system of $n = 200$ qubits. Figure 5A illustrates the sampled topological and trivial states. We generate representatives of the nontrivial topological phase by applying low-depth geometrically local random quantum circuits to Kitaev's toric code state (88) with code distance 10, and we generate representatives of the trivial phase by applying random circuits to a product state.

Randomized Pauli measurements are performed $T = 500$ times to convert the states to their classical shadows, and these classical shadows are mapped to feature vectors in the high-dimensional feature space using the feature map $\varphi^{(\text{shadow})}$. Figure 5B displays a 1D projection of the feature space using the unsupervised classical ML model for various values of the circuit depth, indicating that the phases become harder to distinguish as the circuit depth increases. In Fig. 5C, we show the classification accuracy of the unsupervised classical ML model. We also compare with training CNNs that use measurement outcomes
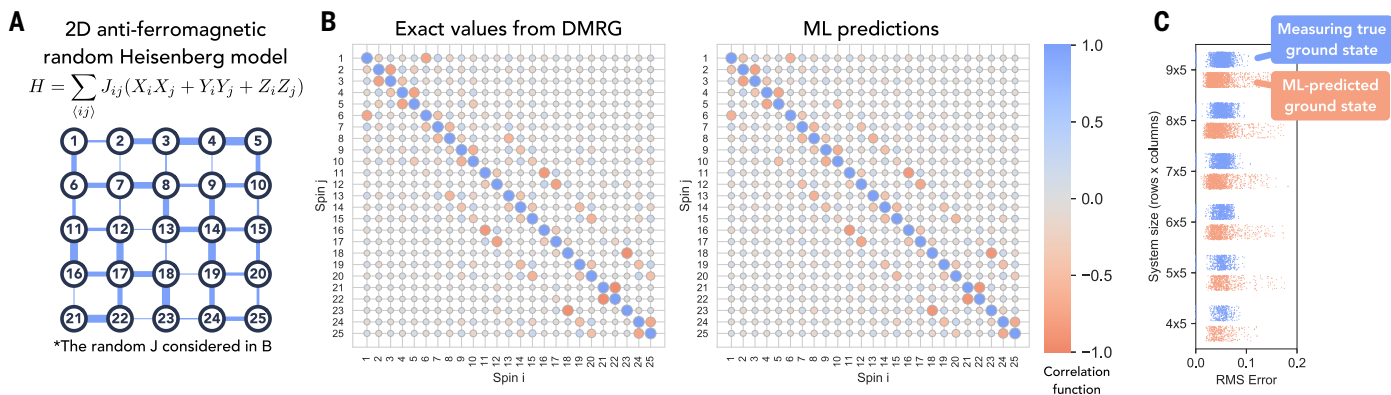
Fig. 3. Numerical experiment for predicting ground-state properties in the 2D antiferromagnetic Heisenberg model with 25 atoms. (**A**) Hamiltonian and the illustration of the Heisenberg model geometry. We consider random couplings J_{ij}, sampled uniformly from $[0, 2]$. A particular instance is shown with coupling strength indicated by the thickness of the edges connecting lattice points. (**B**) Two-point correlator. Exact values and ML predictions of the expectation value of the correlation function $C_{ij} = \frac{1}{3}(X_iX_j + Y_iY_j + Z_iZ_j)$ for all spin pairs (ij) in the lattice, for the Hamiltonian instance shown in (A). The absolute value of C_{ij} is represented by the size of each circle, and the circle's color indicates the actual value. (**C**) Prediction error. Each blue point indicates the RMSE (averaged over Heisenberg model instances) of the correlation function for a particular pair (ij), where the estimate of C_{ij} is obtained using a classical shadow with $T = 500$ randomized Pauli measurements of the true ground state. Red points indicate errors in ML predictions for C_{ij}.
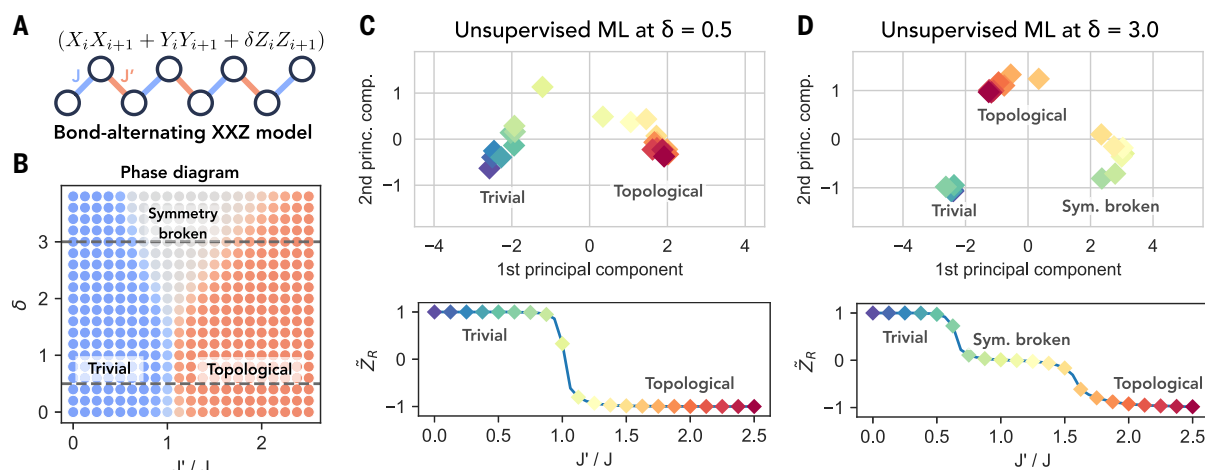
Fig. 4. Numerical experiments for classifying quantum phases in the bond-alternating XXZ model. (**A**) Illustration of the model—a 1D qubit chain, where the coefficient of $(X_iX_{i+1} + Y_iY_{i+1} + \delta Z_iZ_{i+1})$ alternates between J and J'. (**B**) Phase diagram. The system's three distinct phases are characterized by the many-body topological invariant $\tilde{Z}_R$, discussed in (62, 87). Blue denotes $\tilde{Z}_R = 1$, red denotes $\tilde{Z}_R = -1$, and gray denotes $\tilde{Z}_R \approx 0$. (**C** and **D**) Unsupervised phase classification. (Bottom) $\tilde{Z}_R$ versus J/J' at cross sections $\delta = 0.5$ (C) and $\delta = 3.0$ (D) of the phase diagram. (Top) Visualization of the quantum states projected to two dimensions using the unsupervised ML (nonlinear PCA with shadow kernel). In all panels, the colors of the points indicate the value of J/J'; the upper panels suggest that the phases naturally cluster in the expressive feature space.
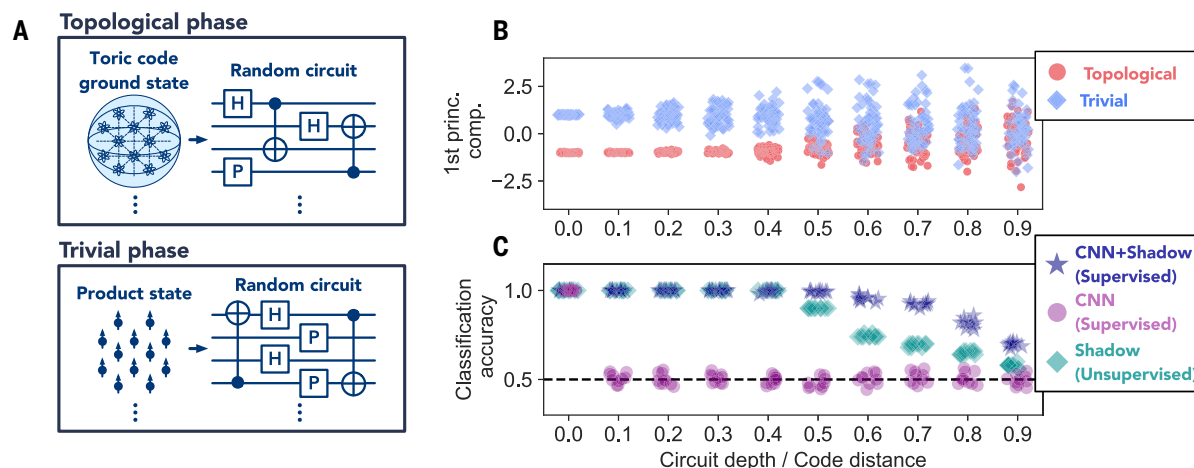
Fig. 5. Numerical experiments for distinguishing between trivial and topological phases. (**A**) State generation. Trivial or topological states are generated by applying local random quantum circuits of some circuit depth to a product state or exactly solved topological state, respectively. (**B**) Unsupervised phase classification. Visualization of the quantum states projected to one dimension using the unsupervised ML (nonlinear PCA with shadow kernel), shown for varying circuit depth (divided by the code distance 10, which quantifies the depth at which the topological properties are washed out). The feature space is sufficiently expressive to resolve the phases for a small enough depth without training, with classification becoming more difficult as the depth increases. (**C**) Classification accuracy for three ML algorithms described in the text.

from the Pauli-6 positive operator-valued measure (POVM) (89) as input to learn an observable for classifying the phases. Because proposition 2 establishes that no observable (even a global one) can classify topological phases, this CNN approach is doomed to fail. On the other hand, if the CNN takes classical shadow representations as input, then it can learn nonlinear functions and successfully classify the phases.

Outlook

We have rigorously established that classical ML algorithms, informed by data collected in physical experiments or using classical calculations, can effectively address some quantum many-body problems. These results boost our hopes that classical ML trained on experimental data can solve practical problems in chemistry and materials science that would be too hard to solve using classical processing alone.

Our arguments build on the concept of a classical shadow derived from randomized Pauli measurements. We expect, though, that other succinct classical representations of quantum states could be exploited by classical ML with similarly powerful results. For example, some currently available quantum simulators are highly programmable but lack the local control needed to perform arbitrary single-qubit Pauli measurements. Instead, after preparing a many-body quantum state of interest, one might switch rapidly to a different Hamiltonian and then allow the state to evolve for a short time before performing a computational basis measurement. How can we make use of that measurement data to predict properties reliably (90, 91)? For that matter, might we be able to generalize from experimental

data that are already routinely available to predict properties of chemical compounds and materials that have not yet been synthesized? Answering such questions will be important goals for future research.

Materials and methods summary

Here, we provide the key ideas for designing ML algorithms to predict ground states and to classify quantum phases of matter. We refer the readers to the supplementary materials (46) for algorithmic details and the proofs of the main theorems.

Predicting ground states

To understand why the ML algorithm works, we begin by considering a simpler task: training an ML model to predict a single ground-state property $tr(O \rho)$, where O is an observable and ρ is the ground state. In this simpler task, the training data are $\{x_l \rightarrow tr[O\rho(x_l)]\}_{l=1}^N$, where $x_l \in [-1, 1]^m$ is a classical description of the Hamiltonian $H(x_l)$ and $\rho(x_l)$ is the ground state of $H(x_l)$. Intuitively, in a quantum phase of matter, the ground-state property $tr[O \rho(x)]$ changes smoothly as a function of the input parameter x. The smoothness condition can be rigorously established as an upper bound on the average magnitude of the gradient of $tr[O \rho(x)]$ using quasi-adiabatic evolution (53, 54), assuming that the spectral gap of $H(x)$ is bounded below by a nonzero constant throughout the parameter space. The upper bound on the average gradient magnitude enables us to design a simple classical ML model based on an l_2-Dirichlet kernel for generalizing from the training set to a new input $x \in [-1, 1]^m$

$$\hat{O}(x) = \frac{1}{N} \sum_{l=1}^N \kappa(x, x_l) tr[O\rho(x_l)] \qquad (7)$$

where $\kappa(x, x_l) = \sum_{k \in Z^m, \|k\|_2 \leq \Lambda} \cos[\pi k \cdot (x - x_l)]$ is the l_2-Dirichlet kernel with cutoff Λ. Using statistical analysis, we can guarantee that the prediction error is small given a number of training data N polynomial in the number of parameters m.

The main idea of the statistical analysis is to bound the model complexity. In particular, the model complexity depends on the number of wave vectors k in the l_2-Dirichlet kernel. The more wave vectors k that we include, the higher the model complexity and the more data needed in the ML model to achieve good prediction performance. We show that the number of m-dimensional wave vectors with a Euclidean norm bounded by Λ is $m^{O(\Lambda)}$, and we only need to consider Λ to be of order $\sqrt{1/\varepsilon}$ to achieve prediction error at most ε.

We then generalize this idea to the task of predicting the ground-state representation. We consider a training data $\{x_l \rightarrow \sigma_T[\rho(x_l)]\}_{l=1}^N$,

where $\sigma_T[\rho(x_l)]$ is the classical shadow representation of the quantum state $\rho(x_l)$ obtained from performing randomized Pauli measurement on the state $\rho(x_l)$. Following the expression for predicting a fixed property, the predicted ground-state representation is given by

$$\hat{\sigma}(x) = \frac{1}{N} \sum_{l=1}^N \kappa(x, x_l) \sigma_T[\rho(x_l)] \qquad (8)$$

Using the property of classical shadows, we have $tr\{O\sigma_T[\rho(x_l)]\} \approx tr[O\rho(x_l)]$ for a wide range of observables O. By moving the sum outside of the trace, we can reduce the problem to predicting a fixed ground-state property. Hence, if the classical ML model based on an l_2-Dirichlet kernel can predict ground-state properties accurately, then it can predict the ground-state representation accurately.

Classifying quantum phases of matter

The ML algorithm is based on the SVM model. The underlying idea of SVM is simple and intuitive. Suppose that we have N data points that form two well-separated clusters. We may try to separate these training clusters with a linear hyperplane. When we get a new data point, we simply check which half space it belongs to and assign the label accordingly. However, there could be many hyperplanes that separate these two training clusters. SVM considers the hyperplane that yields the largest margin, which is equivalent to maximizing the distance from each cluster to the hyperplane. Intuitively, maximizing the margin allows the hyperplane to be most robust to the sampling errors of the training data. Using statistical analysis, one can rigorously show that the bigger the margin, the better the generalization performance would be.

SVM can be enhanced using the kernel trick. When the N data points cannot be separated using a linear hyperplane, we need to separate them using a more complex surface. This is achieved by mapping each data point to a high-dimensional vector space through a nonlinear mapping and looking for a linear hyperplane in the high-dimensional space. One can perform the training and prediction in the high-dimensional space by only computing inner products between two points in the high-dimensional space. The inner product is often referred to as the kernel function, and this technique of mapping to a much larger space is known as the kernel trick. In many situations, one considers the high-dimensional space to be infinite dimensional. The shadow kernel that we defined in Eq. 6 also corresponds to an infinite-dimensional vector space.

For the task of classifying quantum phases of matter, we assume that there exists a classifying function $f(\rho)$ based on a nonlinear function of the reduced-density matrices of the quantum state. More precisely, we assume that states ρ_A

in phase A satisfy $f(\rho_A) > 1$ and states in phase B satisfy $f(\rho_A) < -1$. This assumption is often satisfied when we focus on states not too close to the phase boundary. We show in the supplementary materials (46) that various SPT phases and topologically ordered phases do satisfy this assumption. Because the shadow kernel corresponds to an inner product in an infinite-dimensional space containing all possible non-linear combinations of the reduced-density matrices, SVM based on the shadow kernel is able to learn the classifying function. The amount of data required to learn this classifying function depends on the margin of the hyperplane in the infinite-dimensional space, which can be shown to scale polynomially in system size.

Numerical experiments

For experiments on predicting ground-state properties, we consider the supervised ML algorithm described in Eq. 2. We examine metrics $\kappa(x, x_l) \in R$ based on Gaussian kernel, Dirichlet kernel, and neural tangent kernel (50). Depending on different training data sizes and the number of measurements per quantum state, we found that different kernels perform better than others. For classifying quantum phases of matter, we consider an unsupervised ML algorithm, where no labeled training data are provided. The kernel trick described above can also be applied to unsupervised ML algorithms. A standard example is kernel PCA. PCA tries to find a direction, known as the principal component, such that the data points along this direction are most separated. If the points are not well separated in any direction, then we can consider mapping all points to an infinite-dimensional space. Similar to the supervised setting, we only need to consider inner products between pairs of points in the infinite-dimensional space (kernel function) to find the principal component. Hence, we can also apply the shadow kernel to classify quantum phases of matter in an unsupervised fashion. This is what we considered in the numerical experiments shown in Fig. 4 and Fig. 5.

REFERENCES AND NOTES

1. P. Hohenberg, W. Kohn, Inhomogeneous electron gas. *Phys. Rev.* **136**, B864–B871 (1964). doi: 10.1103/PhysRev.136.B864
2. W. Kohn, Nobel lecture: Electronic structure of matter—wave functions and density functionals. *Rev. Mod. Phys.* **71**, 1253–1266 (1999). doi: 10.1103/RevModPhys.71.1253
3. D. Ceperley, B. Alder, Quantum monte carlo. *Science* **231**, 555–560 (1986). doi: 10.1126/science.231.4738.555; pmid: 17750966
4. A. W. Sandvik, Stochastic series expansion method with operator-loop update. *Phys. Rev. B* **59**, R14157–R14160 (1999). doi: 10.1103/PhysRevB.59.R14157
5. F. Becca, S. Sorella, *Quantum Monte Carlo Approaches for Correlated Systems* (Cambridge Univ. Press, 2017).
6. S. R. White, Density matrix formulation for quantum renormalization groups. *Phys. Rev. Lett.* **69**, 2863–2866 (1992). doi: 10.1103/PhysRevLett.69.2863; pmid: 10046608
7. S. R. White, Density-matrix algorithms for quantum renormalization groups. *Phys. Rev. B* **48**, 10345–10356 (1993). doi: 10.1103/PhysRevB.48.10345; pmid: 10007313

8. G. Carleo et al., Machine learning and the physical sciences. *Rev. Mod. Phys.* **91**, 045002 (2019). doi: 10.1103/RevModPhys.91.045002

9. J. Carrasquilla, Machine learning for quantum matter. *Adv. Phys. X* **5**, 1797528 (2020). doi: 10.1080/23746149.2020.1797528

10. D.-L. Deng, X. Li, S. Das Sarma, Machine learning topological states. *Phys. Rev. B* **96**, 195145 (2017). doi: 10.1103/PhysRevB.96.195145

11. J. Carrasquilla, R. G. Melko, Machine learning phases of matter. *Nat. Phys.* **13**, 431–434 (2017). doi: 10.1038/nphys4035

12. G. Carleo, M. Troyer, Solving the quantum many-body problem with artificial neural networks. *Science* **355**, 602–606 (2017). doi: 10.1126/science.aag2302; pmid: 28183973

13. G. Torlai, R. G. Melko, Learning thermodynamics with Boltzmann machines. *Phys. Rev. B* **94**, 165134 (2016). doi: 10.1103/PhysRevB.94.165134

14. Y. Nomura, A. S. Darmawan, Y. Yamaji, M. Imada, Restricted boltzmann machine learning for solving strongly correlated quantum systems. *Phys. Rev. B* **96**, 205152 (2017). doi: 10.1103/PhysRevB.96.205152

15. E. P. L. van Nieuwenburg, Y.-H. Liu, S. D. Huber, Learning phase transitions by confusion. *Nat. Phys.* **13**, 435–439 (2017). doi: 10.1038/nphys4037

16. L. Wang, Discovering phase transitions with unsupervised learning. *Phys. Rev. B* **94**, 195105 (2016). doi: 10.1103/PhysRevB.94.195105

17. J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, G. E. Dahl, in *Proceedings of the 34th International Conference on Machine Learning*, D. Precup, Y. W. Teh, Eds. (PMLR, 2017), pp. 1263–1272.

18. G. Torlai et al., Neural-network quantum state tomography. *Nat. Phys.* **14**, 447–450 (2018). doi: 10.1038/s41567-018-0048-5

19. R. A. Vargas-Hernández, J. Sous, M. Berciu, R. V. Krems, Extrapolating quantum observables with machine learning: Inferring multiple phase transitions from properties of a single phase. *Phys. Rev. Lett.* **121**, 255702 (2018). doi: 10.1103/PhysRevLett.121.255702; pmid: 30608785

20. K. T. Schütt, M. Gastegger, A. Tkatchenko, K.-R. Müller, R. J. Maurer, Unifying machine learning and quantum chemistry with a deep neural network for molecular wavefunctions. *Nat. Commun.* **10**, 5024 (2019). doi: 10.1038/s41467-019-12875-2; pmid: 31729373

21. I. Glasser, N. Pancotti, M. August, I. D. Rodriguez, J. I. Cirac, Neural-network quantum states, string-bond states, and chiral topological states. *Phys. Rev. X* **8**, 011006 (2018). doi: 10.1103/PhysRevX.8.011006

22. J. F. Rodriguez-Nieva, M. S. Scheurer, Identifying topological order through unsupervised machine learning. *Nat. Phys.* **15**, 790–795 (2019). doi: 10.1038/s41567-019-0512-x

23. Z. Qiao, M. Welborn, A. Anandkumar, F. R. Manby, T. F. Miller 3rd, OrbNet: Deep learning for quantum chemistry using symmetry-adapted atomic-orbital features. *J. Chem. Phys.* **153**, 124111 (2020). doi: 10.1063/5.0021955; pmid: 33003742

24. K. Choo, A. Mezzacapo, G. Carleo, Fermionic neural-network states for ab-initio electronic structure. *Nat. Commun.* **11**, 2368 (2020). doi: 10.1038/s41467-020-15724-9; pmid: 32398658

25. H. Kawai, Y. O. Nakagawa, Predicting excited states from ground state wavefunction by supervised quantum machine learning. *Mach. Learn. Sci. Technol.* **1**, 045027 (2020). doi: 10.1088/2632-2153/aba183

26. J. R. Moreno, G. Carleo, A. Georges, Deep learning the Hohenberg-Kohn maps of density functional theory. *Phys. Rev. Lett.* **125**, 076402 (2020). doi: 10.1103/PhysRevLett.125.076402; pmid: 32857556

27. K. Kottmann, P. Corboz, M. Lewenstein, A. Acín, Unsupervised mapping of phase diagrams of 2D systems from infinite projected entangled-pair states via deep anomaly detection. *SciPost Phys.* **11**, 025 (2021). doi: 10.21468/SciPostPhys.11.2.025

28. A. Bohrdt et al., Classifying snapshots of the doped Hubbard model with machine learning. *Nat. Phys.* **15**, 921–924 (2019). doi: 10.1038/s41567-019-0565-x

29. B. S. Rem et al., Identifying quantum phase transitions using artificial neural networks on experimental data. *Nat. Phys.* **15**, 917–920 (2019). doi: 10.1038/s41567-019-0554-0

30. G. Torlai et al., Integrating neural networks with a quantum simulator for state reconstruction. *Phys. Rev. Lett.* **123**, 230504 (2019). doi: 10.1103/PhysRevLett.123.230504; pmid: 31868463

31. H.-Y. Huang, R. Kueng, J. Preskill, Predicting many properties of a quantum system from very few measurements. *Nat. Phys.* **16**, 1050–1057 (2020). doi: 10.1038/s41567-020-0932-7

32. M. Paini, A. Kalev, D. Padilha, B. Ruck, Estimating expectation values using approximate quantum states. *Quantum* **5**, 413 (2021). doi: 10.22331/q-2021-03-16-413

33. G. Struchalin, Y. A. Zagorovskii, E. Kovlakov, S. Straupe, S. Kulik, Experimental estimation of quantum state properties from classical shadows. *PRX Quantum* **2**, 010307 (2021). doi: 10.1103/PRXQuantum.2.010307

34. A. Elben et al., Mixed-state entanglement from local randomized measurements. *Phys. Rev. Lett.* **125**, 200501 (2020). doi: 10.1103/PhysRevLett.125.200501; pmid: 33258654

35. J. Choi et al., Emergent quantum randomness and benchmarking from Hamiltonian many-body dynamics. arXiv:2103.03535 [quant-ph] (2021).

36. H.-Y. Huang et al., Power of data in quantum machine learning. *Nat. Commun.* **12**, 2631 (2021). doi: 10.1038/s41467-021-22539-9; pmid: 33976136

37. Y. Zhang, R. G. Melko, E.-A. Kim, Machine learning $\mathbb{Z}_2$ quantum spin liquids with quasiparticle statistics. *Phys. Rev. B* **96**, 245119 (2017). doi: 10.1103/PhysRevB.96.245119

38. Y. Zhang, P. Ginsparg, E.-A. Kim, Interpreting machine learning of topological quantum phase transitions. *Phys. Rev. Res.* **2**, 023283 (2020). doi: 10.1103/PhysRevResearch.2.023283

39. M. T. Ribeiro, S. Singh, C. Guestrin, in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Association for Computing Machinery, 2016), pp. 1135–1144.

40. A. Barredo Arrieta et al., Explainable artificial intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Inf. Fusion* **58**, 82–115 (2020). doi: 10.1016/j.inffus.2019.12.012

41. N. Read, Topological phases and quasiparticle braiding. *Phys. Today* **65**, 38–43 (2012). doi: 10.1063/PT.3.1641

42. M. Ohliger, V. Nesme, J. Eisert, Efficient and feasible state tomography of quantum many-body systems. *New J. Phys.* **15**, 015024 (2013). doi: 10.1088/1367-2630/15/1/015024

43. S. J. van Enk, C. W. J. Beenakker, Measuring Trρ^n on single copies of ρ using random measurements. *Phys. Rev. Lett.* **108**, 110503 (2012). doi: 10.1103/PhysRevLett.108.110503; pmid: 22540446

44. A. Elben, B. Vermersch, C. F. Roos, P. Zoller, Statistical correlations between locally randomized measurements: A toolbox for probing entanglement in many-body quantum states. *Phys. Rev. A* **99**, 052323 (2019). doi: 10.1103/PhysRevA.99.052323

45. T. J. Evans, R. Harper, S. T. Flammia, Scalable Bayesian Hamiltonian learning. arXiv:1912.07636 [quant-ph] (2019).

46. See the supplementary materials.

47. A. Jacot, F. Gabriel, C. Hongler, in *Advances in Neural Information Processing Systems*, S. Bengio et al, Eds. (Curran Associates, Inc., 2018), pp. 8571–8580.

48. Z. Li et al., Enhanced convolutional neural tangent kernels. arXiv:1911.00809 [cs.LG] (2019).

49. S. S. Du, K. Hou, B. Póczos, R. Salakhutdinov, R. Wang, K. Xu, in *Proceedings of the 33rd International Conference on Neural Information Processing Systems* (Curran Associates, Inc., 2019), pp. 5723–5733.

50. R. Novak, L. Xiao, J. Hron, J. Lee, A. A. Alemi, J. Sohl-Dickstein, S. S. Schoenholz, Neural Tangents: Fast and Easy Infinite Neural Networks in Python. arXiv:1912.02803 [stat.ML] (2020).

51. C. Cortes, V. Vapnik, Support-vector networks. *Mach. Learn.* **20**, 273–297 (1995). doi: 10.1007/BF00994018

52. C.-C. Chang, C.-J. Lin, LIBSVM: A library for support vector machines. *ACM Trans. Intell. Syst. Technol.* **2**, 1–27 (2011). doi: 10.1145/1961189.1961199

53. M. B. Hastings, X.-G. Wen, Quasiadiabatic continuation of quantum states: The stability of topological ground-state degeneracy and emergent gauge invariance. *Phys. Rev. B* **72**, 045141 (2005). doi: 10.1103/PhysRevB.72.045141

54. S. Bachmann, S. Michalakis, B. Nachtergaele, R. Sims, Automorphic equivalence within gapped phases of quantum lattice systems. *Commun. Math. Phys.* **309**, 835–871 (2012). doi: 10.1007/s00220-011-1380-0

55. D. Lichtenstein, Planar formulae and their uses. *SIAM J. Comput.* **11**, 329–343 (1982). doi: 10.1137/0211025

56. L. Valiant, V. Vazirani, NP is as easy as detecting unique solutions. *Theor. Comput. Sci.* **47**, 85–93 (1986). doi: 10.1016/0304-3975(86)90135-0

57. A. Kitaev, J. Preskill, Topological entanglement entropy. *Phys. Rev. Lett.* **96**, 110404 (2006). doi: 10.1103/PhysRevLett.96.110404; pmid: 16605802

58. M. Levin, X.-G. Wen, Detecting topological order in a ground state wave function. *Phys. Rev. Lett.* **96**, 110405 (2006). doi: 10.1103/PhysRevLett.96.110405; pmid: 16605803

59. M. B. Hastings, T. Koma, Spectral gap and exponential decay of correlations. *Commun. Math. Phys.* **265**, 781–804 (2006). doi: 10.1007/s00220-006-0030-4

60. H. Li, F. D. M. Haldane, Entanglement spectrum as a generalization of entanglement entropy: Identification of topological order in non-Abelian fractional quantum Hall effect states. *Phys. Rev. Lett.* **101**, 010504 (2008). doi: 10.1103/PhysRevLett.101.010504; pmid: 18764098

61. F. Pollmann, A. M. Turner, E. Berg, M. Oshikawa, Entanglement spectrum of a topological phase in one dimension. *Phys. Rev. B* **81**, 064439 (2010). doi: 10.1103/PhysRevB.81.064439

62. F. Pollmann, A. M. Turner, Detection of symmetry-protected topological phases in one dimension. *Phys. Rev. B* **86**, 125441 (2012). doi: 10.1103/PhysRevB.86.125441

63. J. Haegeman, D. Pérez-García, I. Cirac, N. Schuch, Order parameter for symmetry-protected phases in one dimension. *Phys. Rev. Lett.* **109**, 050402 (2012). doi: 10.1103/PhysRevLett.109.050402; pmid: 23006149

64. H. Shapourian, K. Shiozaki, S. Ryu, Many-Body Topological Invariants for Fermionic Symmetry-Protected Topological Phases. *Phys. Rev. Lett.* **118**, 216402 (2017). doi: 10.1103/PhysRevLett.118.216402; pmid: 28598638

65. H. Dehghani, Z.-P. Cian, M. Hafezi, M. Barkeshli, Extraction of the many-body Chern number from a single wave function. *Phys. Rev. B* **103**, 075102 (2021). doi: 10.1103/PhysRevB.103.075102

66. A. Y. Kitaev, Anyons in an exactly solved model and beyond. *Ann. Phys.* **321**, 2–111 (2006). doi: 10.1016/j.aop.2005.10.005

67. Y. Zhang, E.-A. Kim, Quantum loop topography for machine learning. *Phys. Rev. Lett.* **118**, 216401 (2017). doi: 10.1103/PhysRevLett.118.216401; pmid: 28598670

68. M. B. Hastings, S. Michalakis, Quantization of Hall Conductance for Interacting Electrons on a Torus. *Commun. Math. Phys.* **334**, 433–471 (2015). doi: 10.1007/s00220-014-2167-x

69. A. Kapustin, N. Sopenko, Hall conductance and the statistics of flux insertions in gapped interacting lattice systems. *J. Math. Phys.* **61**, 101901 (2020). doi: 10.1063/5.0022944

70. S. Bachmann, A. Bols, W. De Roeck, M. Fraas, A Many-Body Index for Quantum Charge Transport. *Commun. Math. Phys.* **375**, 1249–1272 (2020). doi: 10.1007/s00220-019-03537-x

71. S. Bachmann, B. Nachtergaele, On Gapped Phases with a Continuous Symmetry and Boundary Operators. *J. Stat. Phys.* **154**, 91–112 (2014). doi: 10.1007/s10955-013-0850-5

72. H. Tasaki, Topological phase transition and $\mathbb{Z}_2$ index for $S = 1$ quantum spin chains. *Phys. Rev. Lett.* **121**, 140604 (2018). doi: 10.1103/PhysRevLett.121.140604; pmid: 30339444

73. H. Tasaki, *Physics and Mathematics of Quantum Many-Body Systems*, (Springer, 2020).

74. K. P. Murphy, *Machine Learning: A Probabilistic Perspective* (MIT Press, 2012).

75. P. Fendley, K. Sengupta, S. Sachdev, Competing density-wave orders in a one-dimensional hard-boson model. *Phys. Rev. B* **69**, 075106 (2004). doi: 10.1103/PhysRevB.69.075106

76. A. Browaeys, T. Lahaye, Many-body physics with individually controlled Rydberg atoms. *Nat. Phys.* **16**, 132–142 (2020). doi: 10.1038/s41567-019-0733-z

77. H. Bernien et al., Probing many-body dynamics on a 51-atom quantum simulator. *Nature* **551**, 579–584 (2017). doi: 10.1038/nature24622; pmid: 29189778

78. P. Schauß et al., Crystallization in Ising quantum magnets. *Science* **347**, 1455–1458 (2015). doi: 10.1126/science.1258351; pmid: 25814579

79. M. Endres et al., Atom-by-atom assembly of defect-free one-dimensional cold atom arrays. *Science* **354**, 1024–1027 (2016). doi: 10.1126/science.aah3752; pmid: 27811284

80. H. Labuhn et al., Tunable two-dimensional arrays of single Rydberg atoms for realizing quantum Ising models. *Nature* **534**, 667–670 (2016). doi: 10.1038/nature18274

81. S. Ebadi et al., Quantum phases of matter on a 256-atom programmable quantum simulator. *Nature* **595**, 227–232 (2021). doi: 10.1038/s41586-021-03582-4; pmid: 34234334

82. P. Scholl et al., Quantum simulation of 2D antiferromagnets with hundreds of Rydberg atoms. *Nature* **595**, 233–238 (2021). doi: 10.1038/s41586-021-03585-1; pmid: 34234335

83. U. Schollwöck, The density-matrix renormalization group in the age of matrix product states. *Ann. Phys.* **326**, 96–192 (2011). doi: 10.1016/j.aop.2010.09.012

84. K. Pearson, LIII. On lines and planes of closest fit to systems of points in space. *Lond. Edinb. Dublin Philos. Mag. J. Sci.* **2**, 559–572 (1901). doi: 10.1080/14786440109462720

85. B. Schölkopf, A. Smola, K.-R. Müller, Nonlinear component analysis as a kernel eigenvalue problem. *Neural Comput.* **10**, 1299–1319 (1998). doi: 10.1162/089976698300017467

86. W. Su, J. Schrieffer, A. J. Heeger, Solitons in polyacetylene. *Phys. Rev. Lett.* **42**, 1698–1701 (1979). doi: 10.1103/PhysRevLett.42.1698

87. A. Elben *et al.*, Many-body topological invariants from randomized measurements in synthetic quantum matter. *Sci. Adv.* **6**, eaaz3666 (2020). doi: 10.1126/sciadv.aaz3666; pmid: 32300654

88. A. Y. Kitaev, Fault-tolerant quantum computation by anyons. *Ann. Phys.* **303**, 2–30 (2003). doi: 10.1016/S0003-4916(02)00018-0

89. J. Carrasquilla, G. Torlai, R. G. Melko, L. Aolita, Reconstructing quantum states with generative models. *Nat. Mach. Intell.* **1**, 155–161 (2019). doi: 10.1038/s42256-019-0028-1

90. J. S. Cotler *et al.*, Emergent quantum state designs from individual many-body wavefunctions. arXiv:2103.03536 [quant-ph] (2021).

91. H.-Y. Hu, Y.-Z. You, Hamiltonian-driven shadow tomography of quantum states. *Phys. Rev. Res.* **4**, 013054 (2022). doi: 10.1103/PhysRevResearch.4.013054

92. M. Fishman, S. R. White, E. M. Stoudenmire, The Itensor Software Library for Tensor Network Calculations. arXiv:2007.14822 [cs.MS] (2020).

93. G. Torlai, M. Fishman, PastaQ.jl: A Package for Simulation, Tomography and Analysis of Quantum Computers, Github (2020); https://github.com/GTorlai/PastaQ.jl.

94. H.-Y. Huang, hsinyuan-huang/provable-ml-quantum: (Science), *Zenodo* (2022); .doi: 10.5281/zenodo.6942183

95. H.-Y. Huang, Provably efficient machine learning for quantum many-body problems, data set, *Zenodo* (2022). doi: 10.5281/zenodo.6525856

SUPPLEMENTARY MATERIALS

science.org/doi/10.1126/science.abk3333
Materials and Methods
Supplementary Text
Figs. S1 to S8
References (*96–209*)

Submitted 7 July 2021; accepted 16 August 2022
10.1126/science.abk3333