IOP Publishing Quantum Sci. Technol. 10 (2025) 025006 https://doi.org/10.1088/2058-9565/ada79b

Quantum Science and Technology

OPEN ACCESS

RECEIVED
9 October 2024

REVISED
13 December 2024

ACCEPTED FOR PUBLICATION
8 January 2025

PUBLISHED
20 January 2025

PAPER

Learning to classify quantum phases of matter with a few measurements

Mehran Khosrojerdi[1], **Jason L Pereira**[2], **Alessandro Cuccoli**[1,2] **and Leonardo Banchi**[1,2,*]

[1] Department of Physics and Astronomy, University of Florence, via G. Sansone 1, I-50019 Sesto Fiorentino (FI), Italy
[2] INFN Sezione di Firenze, via G. Sansone 1, I-50019 Sesto Fiorentino (FI), Italy
* Author to whom any correspondence should be addressed.

E-mail: leonardo.banchi@unifi.it

Keywords: quantum machine learning, quantum phase transitions, quantum spin models, tensor networks

Abstract

We study the identification of quantum phases of matter, at zero temperature, when only part of the phase diagram is known in advance. Following a supervised learning approach, we show how to use our previous knowledge to construct an observable capable of classifying the phase even in the unknown region. By using a combination of classical and quantum techniques, such as tensor networks, kernel methods, generalization bounds, quantum algorithms, and shadow estimators, we show that, in some cases, the certification of new ground states can be obtained with a polynomial number of measurements. An important application of our findings is the classification of the phases of matter obtained in quantum simulators, e.g. cold atom experiments, capable of efficiently preparing ground states of complex many-particle systems and applying simple measurements, e.g. single qubit measurements, but unable to perform a universal set of gates.

1. Introduction

Identifying phases of quantum matter is a complex task [1, 2] that requires the extraction of possibly non-local features from the ground states of quantum many-particle systems. Given the recent remarkable success of machine learning methods to extract general laws from a few known examples [3], many studies applied machine learning techniques to learn quantum phases of matter [4–11]—see [3] for more references. Among the different learning methods that can be applied to 'quantum data' [12], such as the ground states of quantum many-particle systems, kernel methods stand out for their interpretability, since learning consists in finding a decision hyperplane in a Hilbert space [12–14]—see appendix A for a brief introduction. The problem of classifying quantum phases of matter with kernel methods has been recently considered in [15, 16]—see appendix C for a detailed comparison between the different approaches.

In a quantum phase classification problem, the state we want to classify is the ground state of a parameterized Hamiltonian with a specific form. For some parameter values we may have full (classical) knowledge of the phases. In many physically relevant scenarios, this comes from our ability to classically simulate the true ground state of the Hamiltonian, e.g. using matrix product states (MPS) or other tensor network methods [17, 18]. This suggests a machine learning approach where training is purely classical, performed on states within the known parameter region, with the ultimate aim of learning a general rule (e.g. an observable or a measurement) that can be applied to classify ground states unseen during training.

One physically motivated reason to *test* a machine learning model with different ground states may be that classical approximation methods are not accurate in that parameter regime. A central result of our analysis is that, if such ground state may be prepared in a quantum device such as quantum simulators [19], e.g. by cooling a quantum many-body system, the classification of new ground states, unseen during training, can be obtained with single-qubit measurements. This result is derived by pairing kernel methods, tensor network techniques, generalization bounds [20, 21], and the certification strategy from [22]. Our algorithm is hybrid, since training is performed classically, while testing new states can be done in different ways, e.g. in quantum devices. Since single-qubit measurements are routinely performed in quantum

simulators, an universal quantum computer is never required. Nonetheless, the same approach can also be applied to different situations, e.g. when we have classical knowledge of the test ground states or when they are approximated in a quantum computer via parametric quantum circuits [23].

2. Problem description

We are interested in getting an approximation of the phase diagram for the ground state of a quantum many-particle Hamiltonian $H(\boldsymbol{x})$ that depends on classical external parameters $\boldsymbol{x} = (x_1, x_2, \dots)$. We focus on settings satisfying two main assumptions: (i) the ground state $|\psi(\boldsymbol{x})\rangle$ of $H(\boldsymbol{x})$ can be obtained with sufficiently high precision, for certain values of $\boldsymbol{x}$, using classical methods based on tensor networks [18]; (ii) within the parameter subspace where the above methods are accurate, there is a (usually smaller) subspace $\mathcal{S}$ where the phase of $|\psi(\boldsymbol{x})\rangle$ is known for all $\boldsymbol{x} \in \mathcal{S}$. The task is to use our knowledge from subregion $\mathcal{S}$ to construct an observable Λ that allows us to predict the phase of a ground state. If we are given copies of a state $|\psi(\boldsymbol{x})\rangle$ for unknown $\boldsymbol{x} \in \mathcal{S}$ (e.g. as the outcome of an experiment), we can evaluate the expectation value of Λ to determine its phase (in-distribution generalization). Alternatively, for a test point outside the known subregion ($\boldsymbol{x} \notin \mathcal{S}$), we can use Λ to learn the phase diagram outside $\mathcal{S}$ (out-of-distribution generalization) with the help of a quantum computer or simulator, e.g. a cold-atom experiment, able to efficiently find the ground state when $\boldsymbol{x} \notin \mathcal{S}$.

To construct Λ we use a supervised learning approach, where we first build a training set $\{|\psi(\boldsymbol{x}_n)\rangle, y_n\}$, for different choices of $\boldsymbol{x}_n$ from $\mathcal{S}$, with our best approximation of the true ground state of $H(\boldsymbol{x}_n)$, $|\psi(\boldsymbol{x}_n)\rangle$, and its known phase y_n. The index n is a discrete index labeling the M training pairs. To simplify the presentation, we use the same notation $|\psi(\boldsymbol{x})\rangle$ for the true ground state and our best approximation of it. In choosing the training algorithm we considered two desirable properties: (i) it should work efficiently with both classical (e.g. tensor network) and shallow quantum circuit approximations of many-body quantum states whose Hilbert space is exponentially large, and (ii) it should give, as an output, a quantity that can be efficiently measured on a quantum device. Because of those requirements, we decided to focus on kernel methods [12].

For any two points $\boldsymbol{x}_n$ and $\boldsymbol{x}_m$, the kernel is defined as

$$k(\boldsymbol{x}_n, \boldsymbol{x}_m) = \mathrm{Tr}\left[\rho(\boldsymbol{x}_n)\,\rho(\boldsymbol{x}_m)\right] = |\langle\psi(\boldsymbol{x}_n)\,|\psi(\boldsymbol{x}_m)\rangle|^2, \tag{1}$$

where $\rho(\boldsymbol{x}) = |\psi(\boldsymbol{x})\rangle\langle\psi(\boldsymbol{x})|$. Both training and testing in kernel methods only require the above kernel matrix, and never use the full ground states, which live in a exponentially large Hilbert space. As such, it can be easily extended to settings involving many-particle systems, provided that the inner product (1) can be efficiently computed, as with tensor networks, shallow quantum circuits, or a combination of both [22, 24, 25]. Another, more physical, reason to focus on kernel methods for classifying phases of matter is that, for nearby points $k(\boldsymbol{x}, \boldsymbol{x} + \mathrm{d}\boldsymbol{x})$, the kernel reduces to the fidelity susceptibility, which has been extensively used to predict phase transitions [26–28]. Therefore, it is reasonable to expect that the kernel entries (1) contain all the information about the quantum phases.

In kernel methods, training consists in finding a decision function $y = f(\boldsymbol{x})$ that can infer the label y (namely the quantum phase) given a certain input $\boldsymbol{x}$. The training algorithm involves a convex optimization routine, which always reaches the global optimum, and whose outcome is a weight vector $\boldsymbol{\alpha}$ of real numbers, together with the baseline $b \in \mathbb{R}$. From $\boldsymbol{\alpha}$ and b, the algorithm constructs the *decision function* as a 'kernel expansion of the data' $\sum_n \alpha_n y_n k(\boldsymbol{x}_n, \boldsymbol{x}) + b$, where $\boldsymbol{x}_n$ are the points used during training, and $\boldsymbol{x}$ is a new point, possibly not belonging to $\mathcal{S}$, that we want to test. The crucial observation, already noted in [12], is that the kernel expansion can also be expressed as the expectation value of an observable Λ over the state $\rho(\boldsymbol{x})$. For instance, in the simplest case of binary classification, with two phases labeled as $y = \pm 1$, the model prediction is expressed as

$$y_{\text{predicted}} = \mathrm{sign}\left(\sum_n y_n \alpha_n k(\boldsymbol{x}_n, \boldsymbol{x}) + b\right) \tag{2}$$

$$= \mathrm{sign}\left(\langle\psi(\boldsymbol{x})\,|\Lambda|\psi(\boldsymbol{x})\rangle + b\right), \tag{3}$$

where in the second line we have defined the observable

$$\Lambda = \sum_n y_n \alpha_n \rho(\boldsymbol{x}_n), \tag{4}$$

as a weighted combination of density matrices. In other words, kernel methods allow for the construction of a *decision operator* Λ that, being Hermitian, can be measured in a quantum device capable of preparing the state $|\psi(\boldsymbol{x})\rangle$ even for those values of $\boldsymbol{x}$ for which the true phase is unknown.

Decision functions with multiple labels are mapped into multiple binary decisions, each with a different decision observable. Letting N_C be the number of classes, the one-versus-all strategy builds N_C observables Λ_i (and shifts b_i), each solving the binary decision problem of whether the quantum phase was i or not i; the class of a new state ρ is predicted from the measured output with highest absolute value: $\arg\max_i |\mathrm{Tr}[\rho\Lambda_i] + b_i|$. The one-versus-one strategy builds $N_C(N_C-1)/2$ observables Λ_{ij}, each solving the binary decision problem between two phases i and j. Testing a new state is based on 'voting', i.e. the predicted phase is the one that 'wins' most of the binary decision problems. For numerical results, as a classical training algorithm, we focus on Support Vector Machines (SVMs) [13], which can be computed efficiently using available numerical libraries [29], and use, by default, the one-versus-one strategy. See appendix A for a self-contained introduction.

We now discuss the evaluation of the kernel entries (1). For training and in-distribution generalization, or in any other case when the true states are replaced by their tensor network approximations, inner product (1) can be computed efficiently on classical hardware [17]. When the true states are approximated via a shallow quantum circuit, it can be computed using different methods such as the swap test or inversion test [12]. For out-of-distribution generalization where the overlap is hybrid, between a *classical* and a quantum state, single-qubit measurement suffice [22], provided each amplitude of the classical state can be computed efficiently. When the classical state is represented as an MPS, each amplitude requires $\mathcal{O}(N)$ operations, where N is the number of qubits. Alternatively, each state $\rho(\boldsymbol{x}_n)$ can be mapped into a *shallow* quantum circuit [24, 25], which can be implemented in a quantum device without exponential overhead.

Given the semi-analytic structure of Λ, we can also express the variance as

$$\mathrm{Var}(\Lambda)_{\rho(\boldsymbol{x})} = \sum_{nm} \alpha_n \alpha_m y_n y_m \Big(\mathrm{Tr}\left[\rho(\boldsymbol{x}_n)\rho(\boldsymbol{x}_m)\rho(\boldsymbol{x})\right] - \mathrm{Tr}\left[\rho(\boldsymbol{x})\rho(\boldsymbol{x}_n)\right]\mathrm{Tr}\left[\rho(\boldsymbol{x})\rho(\boldsymbol{x}_m)\right] \Big), \tag{5}$$

which can be classically estimated efficiently for the in-distribution case ($\boldsymbol{x} \in \mathcal{S}$), allowing the learner to infer the number of measurements to be performed in the quantum device to achieve the desired precision. From Hölder's inequality, we get the bound $\mathrm{Var}(\Lambda) \leqslant (\sum_n \alpha_n)^2 \leqslant (CN_S)^2$, where N_S is the number of support vectors and C is the regularization parameter—typically $C = 1$, see appendix A. Clearly $N_S \leqslant M$, where M is the number of training data. Therefore, the number of measurement shots to be performed scales at most quadratically in the number of training data. We will numerically show, using the generalization bounds from [20, 21], that, for the in-distribution case, the number of training data to reach a desired accuracy scales linearly with the number of qubits. When this is the case, the overall complexity of our method scales polynomially with the number of qubits. When paired with [22], the whole measurement will require a polynomial number of single-qubit measurements. More details about the measurements and a brief discussion about the still open related problems are given in appendix B.

2.1. Handling symmetries

Symmetries may introduce degeneracies in the ground state. Consider the simplest case with two degenerate ground states labeled by $|\psi_\pm(\boldsymbol{x})\rangle$. For instance, if $H(\boldsymbol{x})$ commutes with the parity operator, as in one of our numerical results, $\pm$ may label the parity, and $|\psi_\pm(\boldsymbol{x})\rangle$ will be orthogonal, having two different parities. If the symmetry is not enforced by the diagonalization procedure, the ground state appears as a superposition

$$|\psi_c(\boldsymbol{x})\rangle = \alpha_+|\psi_+(\boldsymbol{x})\rangle + \alpha_-|\psi_-(\boldsymbol{x})\rangle, \tag{6}$$

with possibly random coefficients $\alpha_\pm$. Those coefficients represent a problem for kernel methods, as they introduce arbitrariness in the kernel entries (1), which might become random. To avoid this, during training we compute $|\psi_\pm(\boldsymbol{x})\rangle$, e.g. by implementing the symmetry in the tensor network [30] and insert both states into the training set. During test on the other hand, we assume we have less control and we do not implement any of those strategies. Therefore, each test ground state may appear as a possibly random combination of different states. This will be the case in ground states created by quantum devices, unless symmetries are explicitly enforced [31]. Nonetheless, at test the model should have learnt to separately classify the states with different parity, and so it should now be able to classify even states of the form (6).

3. Numerics

3.1. Axial next-nearest-neighbor ising (ANNNI) model

As a first example, we focus on the one-dimensional ANNNI model [32–35], which describes N spin-$\frac{1}{2}$ particles, with the Hamiltonian

$$H = -\sum_{j=1}^{N-1} \sigma_j^{\mathrm{x}}\sigma_{j+1}^{\mathrm{x}} + k\sum_{j=1}^{N-2} \sigma_j^{\mathrm{x}}\sigma_{j+2}^{\mathrm{x}} - h\sum_{j=1}^{N} \sigma_j^{\mathrm{z}}, \tag{7}$$

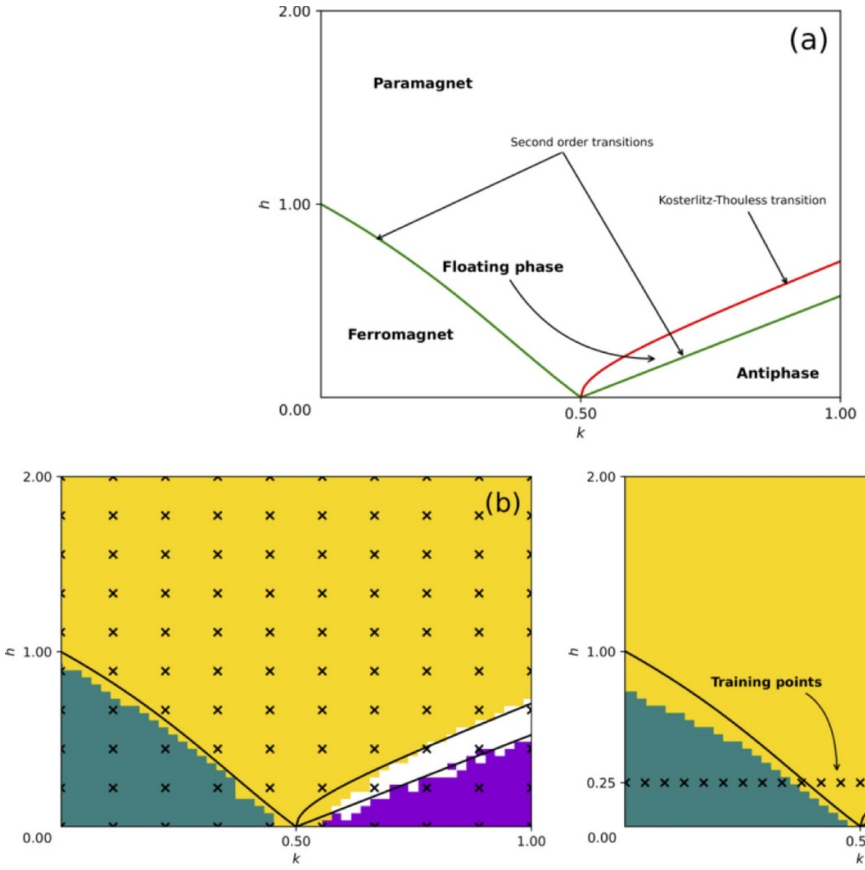

Figure 1. (a) Expected phase diagram of the ANNNI model in the thermodynamic limit. (b), (c) Predicted phase diagram for the ANNNI Hamiltonian (7) with $N = 50$ spins, where the ground state is simulated using tensor network methods with a small bond dimension (20). The training points are marked with a black cross and the phases and phase transition lines are the same of figure 1. In (b) the training point are uniformly scattered, while in (c) they are located on the line $h = 0.25$.

where the two parameters $\boldsymbol{x} = (k, h)$ are both assumed to be positive. The above model displays a global $\mathbb{Z}_2$ symmetry, as $[H, P] = 0$, where $P = \bigotimes_n \sigma_n^{\mathtt{z}}$ is the parity operator. Therefore, energy eigenstates can be degenerate and come as pairs with different parities.

The Hamiltonian (7) models ferromagnetic interactions between the $\mathtt{x}-$components of nearest neighboring spins (NN), and antiferromagnetic interactions (with relative strength k) between the same components of next-nearest neighboring spins (NNN), together with the action of a magnetic field of strength h along the $\mathtt{z}$ direction. For $k = 0$ the model is exactly solvable using fermionization [36], while for $h = 0$ the model is effectively classical, as the Hamiltonian is diagonal in the $\mathtt{x}$ basis. For $k > 0$, the system is frustrated, as the first term favors spin configurations with all spins aligned along the $\mathtt{x}$ axis, while the second term favors configurations where next-nearest spins are anti-aligned[3].

The phase diagram of the above model, shown in figure 1(a), has been characterized with different methods [34, 35, 37–40], and the possibility of using tensor network techniques to compute its ground state was analyzed in [37, 41]. For small h and small k, the nearest neighbor interaction dominates, and the model displays ferromagnetic order; for sufficiently large h the model is in a paramagnetic phase, while for large k and suitably small h the next-nearest neighbor interaction dominates and the model displays the so-called antiphase (AP), where pairs of aligned spins alternate pointing in opposite direction. Between the AP and the paramagnetic phase the model displays a floating phase, where correlation functions $\langle \sigma_j^{\mathtt{x}} \sigma_{j+l}^{\mathtt{x}} \rangle - \langle \sigma_j^{\mathtt{x}} \rangle^2$ decay algebraically rather than exponentially, and the Hamiltonian becomes gapless in the thermodynamic limit $N \to \infty$.

[3] Actually, the sign of the nearest neighbors interaction is not really relevant, as the Hamiltonian can be mapped into itself by changing the NN ferromagnetic interaction into an anti-ferromagnetic one, while rotating by an angle π the spins on the even sites by a $\mathbb{Z}_2$ transformation $\bigotimes_{n \text{ even}} \sigma_n^{\mathtt{z}}$, explicitly proving that the frustrated nature of the model does not depend on the sign of the NN interaction, as two spin configurations cannot display exact anti-ferromagnetic order between next-nearest neighbors either when nearest neighbors are aligned or when they are anti-aligned.

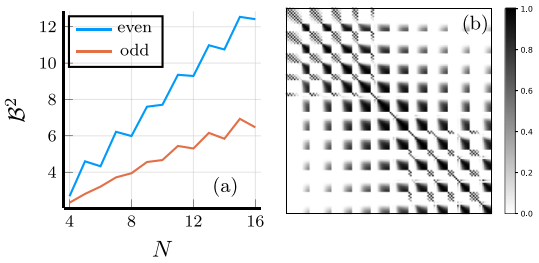

Figure 2. (a) Scaling of the generalization error bound equation (D1), for the ground state of the ANNNI model with $h \in [0, 2]$, $k \in [0, 1]$, as a function of the number of qubits N, and for the two different parity sectors (even and odd). (b) Kernel matrix entries for the training states shown in figure 1.

Aside from these general properties, the phase diagrams obtained with different methods do not always agree [42, 43]: Indeed, while there is a general consensus over the different phases schematically depicted in figure 1(a), some works split the paramagnetic phase into two different phases [35], while others found the floating phase over an extended region. For the phase diagram of figure 1, the transition lines between the different phases have been found using different perturbative and numerical methods to approximately be [5, 37]

$$h_{\mathrm{I}}(k) \simeq \frac{1-k}{k}\left(1-\sqrt{\frac{1-3k+4k^2}{1-k}}\right), \tag{8a}$$

$$h_{\mathrm{KT}}(k) \simeq 1.05\sqrt{(k-0.5)(k-0.1)}, \tag{8b}$$

$$h_{\mathrm{AP}}(k) \simeq 1.05\ (k-0.5). \tag{8c}$$

Our numerical results for the ANNNI model are shown in figure 1 for a chain with $N = 50$ spins. The ground states were obtained using the DMRG algorithm implemented in the `quimb` library [44] . In spite of the small bond dimension (20), the predicted phase diagram is in good agreement with the theoretical prediction, shown in figure 1, with differences only near the phase transition points. These differences might be due to finite size effects, as the phase diagram in figure 1 is expected in the thermodynamic limit $N \to \infty$, and due to the finite bond dimension, as close to the phase transition point the entanglement area law breaks down [45] and a larger bond dimension might be needed. This highlights a use case in which a classically learned observable could be measured in a shallow quantum circuit for ground states near the transition point in order to locate the phase boundary with greater precision than is possible purely classically.

In figure 1 we trained the model in two different ways, and then tested over the same parameter region. In panel (b), the training points are uniformly spaced (in-distribution), while in panel (c) they are chosen along a line that passes through all four different phases (out-of-distribution). Prediction in (b) is simpler, as the model has to interpolate over the training points, while in (c) it is more complex as the model has to extrapolate, resulting in a lower accuracy. Nonetheless, in both cases the model displays remarkable generalization abilities given the limited amount of data.

The ability of the training algorithm to generalize, namely to predict the phase of ground states outside of the training set, can be explained using the language of [20, 21] when the training and test data belong to the same distribution. Indeed, the generalization error, namely the difference between an optimal classifier with full knowledge of the data distribution and an empirical classifier that only has access to M samples from that distribution, is upper bounded by $\mathcal{B}/\sqrt{M}$ where $\mathcal{B}$ depends on the mutual information between the parameter space and the set of possible ground states. The in-distribution generalization error is small as long as $M \gg \mathcal{B}^2$. The values of $\mathcal{B}^2$, estimated by following appendix D, are shown in figure 2(a), where we show that $\mathcal{B}^2$ increases at most linearly with the number of qubits. Therefore, in the worst case, at most a number of training points that scales linearly with the number of qubits is sufficient to ensure a good generalization, in spite of the exponentially growing Hilbert space. In practice though, as discussed in [21], because of the regularization term, the above bound might be loose and the model might be able to generalize even with less data. The good generalization capabilities are due to the 'clustering' of the ground states within the different phases, which is evident from the kernel matrix entries shown in figure 2(b). In that figure, the 'tiled' subregions are due to the degeneracy of the ground states, which can be labeled by the parity and are mutually orthogonal.

When training and test data belong to different distributions, as in figure 1(c), we need to employ out-of-distribution bounds, which have been discussed in the quantum case only for different settings [46].

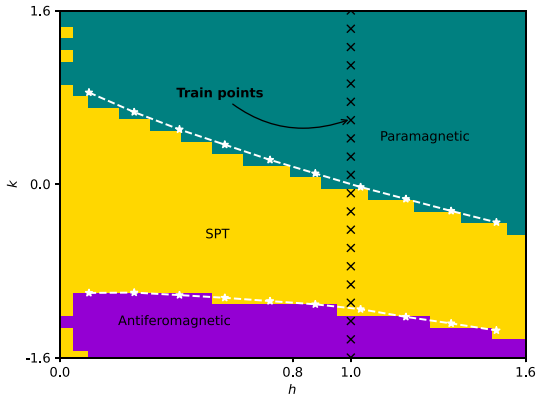

Figure 3. Predicted phase diagram for the Hamiltonian (9) with $N = 41$ spins, where the ground state is simulated using tensor network methods with a bond dimension 150. The training points, marked with black crosses, are located along the line $h = 1$. The phase transition lines (dashed white) are estimated by interpolating the numerically observed points (white asterisks) from [4].

3.2. Haldane model

As a second model, we focus on the one-dimensional symmetry-protected topological spin model described by the Hamiltonian [4]

$$H = -\sum_{j=1}^{N-2} \sigma_j^{\mathrm{x}} \sigma_{j+1}^{\mathrm{z}} \sigma_{j+2}^{\mathrm{x}} - k \sum_{j=1}^{N-1} \sigma_j^{\mathrm{z}} \sigma_{j+1}^{\mathrm{z}} - h \sum_{j=1}^{N} \sigma_j^{\mathrm{z}}, \tag{9}$$

with parameters $\boldsymbol{x} = (h, k)$. When $k = 0$ the model is exactly solvable via the Jordan-Wigner transformation [36]. Such a Hamiltonian has a $\mathbb{Z}_2 \times \mathbb{Z}_2$ symmetry generated by the operators $\hat{X}_{\mathrm{even(odd)}} = \prod_{i \in \mathrm{even(odd)}} \sigma_i^{\mathrm{z}}$. The ground state of H displays, as a function of h and k, a paramagnetic phase, an antiferromagnetic phase and a $\mathbb{Z}_2 \times \mathbb{Z}_2$ symmetry-protected topological (SPT) phase. The latter can be detected by non-zero string order parameters $S_{ab} = \sigma_a^{\mathrm{x}} \prod_{a<i<b} (\sigma_i^{\mathrm{z}}) \sigma_b^{\mathrm{x}}$ or with a quantum convolutional neural network circuit [4].

To train the model, we fix $N = 41$ and provide just the known information from [4, 47] that the model belongs to the SPT phase for $h = 1$ and $-1.15 \leqslant k \leqslant 0$, to the paramagnetic phase for $k > 0$ and to the antiferromagnetic phase for $k < -1.15$. The 50 training points are displayed with black crosses in figure 3. We then test the model over the entire phase space shown in figure 3, discretized as a 30×30 grid. For comparison, we also draw the phase transition lines obtained in [4]. From the results shown in figure 3 we see that the SVM classifier is able to accurately recognise all three phases, generalizing the given information from the states along the $h = 1$ line. Imperfections around the phase transition lines are possibly due to finite size effects or due to the small bond-dimension in the tensor network. Misclassifications around the $h = 0$ line are comparable to those obtained with other methods [4, 47] and may be exacerbated by the fact that the states around $h = 0$ might be qualitatively quite different from the training states along $h = 1$.

4. Conclusions

We studied the classification of quantum phases of matter using a combination of classical and quantum techniques. We studied how to use knowledge from part of a phase diagram to populate other, possibly unknown, parts of the phase diagram. The result of our learning approach is a set of observables that can be measured in quantum devices, with a number of measurements that scale polynomially in the system size. We studied different models with rather rich phase diagrams and showed that our classifier is able to both interpolate and extrapolate the information from the training set to predict the quantum phase for new sets of parameters.

Possible future research directions involve the study of the scaling with the number of qubits, to see whether we can use small N experiments to extrapolate the thermodynamic limit and estimate critical exponents, and the extension to other Hamiltonians and systems. In general, it would also be interesting to study the possible connections between the learnt observables and the order parameters, when those are known.

Data availability statement

No new data were created or analysed in this study.

Acknowledgments

The authors acknowledge financial support from: PNRR Ministero Università e Ricerca Project No. PE0000023-NQSTI funded by European Union-Next-Generation EU (M K, A C and L B:); Prin 2022—DD N. 104 del 2/2/2022, entitled 'understanding the LEarning process of QUantum Neural networks (LeQun)', Proposal Code 2022WHZ5XH, CUP B53D23009530006 (L B); Prin 2022—DD N. 104 del 2/2/2022, Proposal Code 2022SJCKAH, CUP B53D23005250006 (A C); U.S. Department of Energy, Office of Science, National Quantum Information Science Research Centers, Superconducting Quantum Materials and Systems Center (SQMS) under the Contract No. DE-AC02-07CH11359 (J P, A C and L B).

Appendix A. Quantum-enhanced support vector machines

Support Vector Machines [13] are a family of machine learning models based on decision hyperplanes on a Hilbert space, called *feature space*. Training consists in finding such a hyperplane, while classification of a new input is done by checking on which side of the plane the input belongs. Since multiclass classification problems are broken down into a cascade of binary decisions, we will focus on binary classification problems with $Y = \{+1, -1\}$. Given a fixed map $\boldsymbol{x} \mapsto \rho(\boldsymbol{x})$, embedding a classical variable $\boldsymbol{x}$ into the quantum (feature) space, the linear decision hyperplanes in the feature space yields a non-linear hypersurface in the input space, defined as

$$f(\boldsymbol{x}) = \mathrm{Tr}\left[W\rho(\boldsymbol{x})\right] + b = 0. \tag{A1}$$

The above equation describes a hyperplane in the feature space, with parameters encoded into the 'observable' W and the shift b. It can be shown that the optimization of those parameters during training can be cast into a 'dual' formulation that can be solved with open-source libraries [29] with polynomial complexity in the number of training pairs M. The resulting convex optimization problem is described by

$$\underset{\{0\leqslant\alpha_i<C\}}{\arg\max}\left[\sum_{i=1}^{M}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{M} y_i y_j \alpha_i \alpha_j k\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right)\right], \tag{A2}$$

with constraints $\sum_{i=1}^{M}\alpha_i y_i = 0$ and a hyper-parameter C. In the above equation the kernel is defined as in equation (1), and $(\boldsymbol{x}_i, y_i)$ for $i = 1, \ldots, M$ define the M training pairs with inputs $\boldsymbol{x}_i$ and known label y_i. Training consists in finding the coefficients α_i. Once those are obtained, we can get the 'observable' and shift from equation (A1) analytically as

$$W = \sum_{i=1}^{M} y_i \alpha_i \rho\left(\boldsymbol{x}_i\right), \tag{A3}$$

$$b = \sum_{i=1}^{M}\left(\frac{y_i - \sum_{j=1}^{M}\alpha_j y_j k\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right)}{M}\right), \tag{A4}$$

as well, as the predicted class y of a new test $\boldsymbol{x}$, via the decision function (A1)

$$y = \mathrm{sign}\left[f(\boldsymbol{x})\right] = \mathrm{sign}\left[b + \sum_{i=1}^{M}\alpha_i y_i k\left(\boldsymbol{x}_i, \boldsymbol{x}\right)\right]. \tag{A5}$$

We conclude this appendix by clarifying the differences between the classification of quantum and classical data. In many applications of quantum kernel methods [12] we are interested in classifying complex classical data $\boldsymbol{x}$, e.g. images, by first embedding those images into a quantum state $|\psi(\boldsymbol{x})\rangle$. However, there is no general rule to define such an embedding circuit, which can then be trained as well [14, 48]. On the other hand, in our case, the classical parameters $\boldsymbol{x}$ are much less complex (often described by two real numbers rather than by the many pixels of an image), and the encoding map $\boldsymbol{x} \mapsto \rho(\boldsymbol{x})$ is fixed by nature, being the map that associates to any $\boldsymbol{x}$ the ground state of $H(\boldsymbol{x})$. For classical data, the purpose of the kernel is to define a non-linear map into a much larger Hilbert space, called the feature space, where complex classical data are expected to be easier to discriminate. In our case the 'feature' space is the physical Hilbert space of density matrices of quantum many-particle systems.

Appendix B. Measuring kernel entries

In this section we provide further details on the measurement to be performed to test a new state. Using equation (3), we need to estimate $k(\boldsymbol{x},\boldsymbol{x}_n)$, i.e. the kernel entry between a state $\rho(x)$ prepared on a quantum device and a training state $\rho(\boldsymbol{x}_n) = |\psi(\boldsymbol{x}_n)\rangle\langle\psi(\boldsymbol{x}_n)|$, for which we have a classical description as a matrix product state

$$|\psi(x_n)\rangle = \sum_{\{i_n\}} A_1^{i_1} A_2^{i_2} \dots A_N^{i_N} |i_1, \dots, i_N\rangle, \tag{B1}$$

with matrices $(A_n^{i_n})_{jk}$ with maximum bond dimension D, i.e. $j,k \leqslant D$. From the analysis in the main text, we claim that a polynomial number of kernel entries suffice, so if each kernel entry can be measured efficiently, the overall procedure will be efficient. We now discuss two possibilities to efficiently measure $k(\boldsymbol{x},\boldsymbol{x}_n)$, given the above tensor product structure of the training state.

B.1. Mapping the MPS to a shallow circuit

This follows from the techniques developed in [24, 25]. In general, each tensor can be decomposed into a unitary acting on at most $\log_2(D)+1$ qubits. For instance, for the numerics shown in figure 1 we used $D = 20$ which would require unitaries acting on at most 6 neighboring qubits. Even considering the action of the N tensors, the resulting unitary U would be shallow. Therefore, using classical computing only, we can find a shallow quantum circuit U_n such that

$$|\psi(x_n)\rangle = U_n|0\dots0\rangle, \tag{B2}$$

and

$$k(\boldsymbol{x},\boldsymbol{x}_n) = \langle 0\dots0|U_n^\dagger \rho(\boldsymbol{x})\, U_n|0\dots0\rangle. \tag{B3}$$

Empirically we can then measure the kernel by first applying the shallow circuit $U_n^\dagger$ to the quantum state $\rho(x)$, perform measurements in the computational basis, and, from them, estimate the probability that all qubits were in the state $|0\rangle$. The main downside of this approach is that it requires the quantum device to be able to apply the shallow quantum circuit U_n. This might be problematic for quantum devices that are not capable of universal quantum computing. Therefore, in the next section we focus on a different strategy that does not require the application of quantum circuits.

B.2. Shadow based overlap estimator

We apply the method proposed in [22]. In the simplest case, the method works as follows:

1. Select a qubit at random. I.e., randomly select k subject to $1 \leqslant k \leqslant N$.
2. Measure all qubits in ρ, except k, in the computational basis and denote the vector of outcomes $\boldsymbol{z}$.
3. Select a measurement basis X, Y, Z at random and measure the kth qubit of ρ in that basis. Denote by $|s\rangle$ the post-measurement state of the kth qubit.
4. Denote by $\boldsymbol{z}^{(i)}$ the vector $\boldsymbol{z}$ with an added entry in the kth element equal to i.
5. Define the single qubit state

$$|\Psi(\boldsymbol{z})\rangle = \frac{\langle \boldsymbol{z}^{(0)}|\psi\rangle\,|0\rangle + \langle \boldsymbol{z}^{(1)}|\psi\rangle\,|1\rangle}{|\langle \boldsymbol{z}^{(0)}|\psi\rangle|^2 + |\langle \boldsymbol{z}^{(1)}|\psi\rangle|^2}. \tag{B4}$$

6. Compute

$$\omega = \langle\Psi(\boldsymbol{z})|(3|s\rangle\langle s| - \mathbb{1})|\Psi(\boldsymbol{z})\rangle. \tag{B5}$$

7. Repeat the above procedure T times and return the average of ω.

The estimator obtained with the above procedure is related to $\langle\psi|\rho|\psi\rangle$ via bounds. However, numerical experiments from [22] show remarkable agreements between the estimated and true value of the fidelity for a wide range of values. Since support vector machines are generally stable against noise in the data, provided that such noise is not engineered to fool the classifier [14], we believe that small imperfections in the fidelity estimation, and hence in the kernel, will not affect the classification accuracy too much. A full numerical treatment is left to future studies.

The main bottleneck in the use of the shadow overlap in the general case is the calculation of the amplitudes $\langle \boldsymbol{z}^{(i)}|\psi\rangle$ in equation (B4). However, for MPS (B1) and for a given bitstring $\boldsymbol{z}$, the amplitude can be estimated efficiently using a linear (in N) number of matrix products.

Appendix C. Comparison with previous studies

In this section we clarify the main differences between our approach and [15, 16]. First of all, we point out that the main ones are due to the hybrid nature of our method, which is physically motivated (we train with what we know, and test with what we do not), and algorithmically efficient, when paired with [22]. Indeed, the algorithm from [22] is efficient in this hybrid regime, where we have an effective classical description of one of the two states. Clearly though, in order to verify the accuracy of the learnt model, we had to test it for parameter regions where we had full knowledge. In what follows we highlight other major differences.

Compared to [15], in our work we use the fidelity squared rather than the fidelity to define the kernel. This seemingly minor change provides us a physical interpretation of the decision hyperplane as a physical observable quantity, which is lacking for the fidelity kernel [12]. Moreover, Sancho-Lorente *et al* [15] applied their algorithm to the Ising model, where quantum phases are easy to learn [20, 21] and the fidelity can be computed analytically, while in this work we apply our method to more challenging phase diagrams. Compared to [16], we use a different cost function, which allows us to use standard algorithms [29] for processing the kernel entries, and we use classical tensor network methods during training, while [16] uses parametric quantum circuits and a quantum variational eigensolver to approximate the ground state (which requires a quantum device) [23]. It is unknown whether the latter can provide an advantage over classical methods [49], e.g. based on tensor networks, while it certainly adds the extra issue of the measurement cost in estimating expectation values. Wu *et al* [16] also conjectures the *worst case* difficulty of classical training, whilst we focus on specific, physically relevant scenarios in which classical training can be demonstrated to be efficient. Compared to both [15, 16] we also explicitly consider degeneracies that, without a careful treatment, might make kernel entries a random number, thus affecting the ability of the model to learn.

Appendix D. Generalization error bounds

Following [20, 21], since ground states are pure, the bound quantity $\mathcal{B}$ can be estimated using m samples (possibly different from the number of training samples M) as

$$\mathcal{B} \simeq \mathrm{Tr}\sqrt{\mathcal{K}}, \qquad \mathcal{K}_{jk} = \langle \psi(\boldsymbol{x}_j) | \psi(\boldsymbol{x}_k) \rangle / m, \tag{D1}$$

for $i,j = 1,\ldots,m$; the error in approximating $\mathcal{B}$ with m samples goes at most as $\mathcal{O}(m^{-1/2})$ [21]. Note that the above kernel matrix differs from (1), as $K_{jk} = |\mathcal{K}_{jk}|^2/m$. We estimated the values of $\mathcal{B}$ shown in figure 2 by using $m = 1000$ random uniform samples of $h \in [0,2]$ and $k \in [0,1]$, end then checked that $\mathcal{B}$ converged, namely that by slightly increasing the number of samples the outcomes were basically the same. The final value is estimated by averaging over 10 repetitions to average out the (small) fluctuations.

ORCID iDs

Jason L Pereira https://orcid.org/0000-0002-0568-5737
Alessandro Cuccoli https://orcid.org/0000-0002-2556-9944
Leonardo Banchi https://orcid.org/0000-0002-6324-8754

References

[1] Sachdev S 1999 Quantum phase transitions *Phys. World* **12** 33
[2] Zinn-Justin J 2007 *Phase Transitions and Renormalization Group* (Oxford University Press)
[3] Carleo G, Cirac I, Cranmer K, Daudet L, Schuld M, Tishby N, Vogt-Maranto L and Zdeborová L 2019 Machine learning and the physical sciences *Rev. Mod. Phys.* **91** 045002
[4] Cong I, Choi S and Lukin M D 2019 Quantum convolutional neural networks *Nat. Phys.* **15** 1273
[5] Cea M, Grossi M, Monaco S, Rico E, Tagliacozzo L and Vallecorsa S 2024 Exploring the phase diagram of the quantum one-dimensional ANNNI model (arXiv:2402.11022)
[6] Monaco S, Kiss O, Mandarino A, Vallecorsa S and Grossi M 2023 Quantum phase detection generalization from marginal quantum neural network models *Phys. Rev.* B **107** L081105
[7] Huang H-Y, Kueng R, Torlai G, Albert V V and Preskill J 2022 Provably efficient machine learning for quantum many-body problems *Science* **377** eabk3333
[8] Dong X-Y *et al* 2019 Machine learning of quantum phase transitions *Phys. Rev.* B **99** 121104
[9] Uvarov A, Kardashin A and Biamonte J D 2020 Machine learning phase transitions with a quantum processor *Phys. Rev.* A **102** 012415
[10] Carrasquilla J and Melko R G 2017 Machine learning phases of matter *Nat. Phys.* **13** 431
[11] Li Q, Huang Y, Hou X, Li Y, Wang X and Bayat A 2024 Ensemble-learning error mitigation for variational quantum shallow-circuit classifiers *Phys. Rev. Res.* **6** 013027
[12] Schuld M 2021 Supervised quantum machine learning models are kernel methods (arXiv:2101.11020)
[13] Cristianini N and Shawe-Taylor J 2000 *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods* (Cambridge university press)

[14] Montalbano G and Banchi L 2024 Quantum adversarial learning for kernel methods (arXiv:2404.05824)
[15] Sancho-Lorente T, Román-Roche J and Zueco D 2022 Quantum kernels to learn the phases of quantum matter *Phys. Rev.* A **105** 042432
[16] Wu Y, Wu B, Wang J and Yuan X 2023 Quantum phase recognition via quantum kernel methods *Quantum* **7** 981
[17] Schollwöck U 2011 The density-matrix renormalization group in the age of matrix product states *Ann. Phys., NY* **326** 96
[18] Bañuls M C 2023 Tensor network algorithms: a route map *Annu. Rev. Condens. Matter Phys.* **14** 173
[19] Altman E *et al* 2021 Quantum simulators: architectures and opportunities *PRX Quantum* **2** 017003
[20] Banchi L, Pereira J and Pirandola S 2021 Generalization in quantum machine learning: a quantum information standpoint *PRX Quantum* **2** 040321
[21] Banchi L, Pereira J L, Jose S T and Simeone O 2024 Statistical complexity of quantum learning *Adv. Quantum Technol.* 2300311
[22] Huang H-Y, Preskill J and Soleimanifar M 2024 Certifying almost all quantum states with few single-qubit measurements (arXiv:2404.07281)
[23] Bharti K *et al* 2022 Noisy intermediate-scale quantum algorithms *Rev. Mod. Phys.* **94** 015004
[24] Cramer M, Plenio M B, Flammia S T, Somma R, Gross D, Bartlett S D, Landon-Cardinal O, Poulin D and Liu Y-K 2010 Efficient quantum state tomography *Nat. Commun.* **1** 149
[25] Rudolph M S, Miller J, Motlagh D, Chen J, Acharya A and Perdomo-Ortiz A 2023 Synergistic pretraining of parametrized quantum circuits via tensor networks *Nat. Commun.* **14** 8367
[26] Zanardi P and Paunković N 2006 Ground state overlap and quantum phase transitions *Phys. Rev.* E **74** 031123
[27] Campos Venuti L and Zanardi P 2007 Quantum critical scaling of the geometric tensors *Phys. Rev. Lett.* **99** 095701
[28] Banchi L, Giorda P and Zanardi P 2014 Quantum information-geometry of dissipative quantum phase transitions *Phys. Rev.* E **89** 022102
[29] Chang C-C and Lin C-J 2011 LIBSVM: a library for support vector machines *ACM Trans. Intell. Syst. Technol.* **2** 1
[30] Singh S, Pfeifer R N and Vidal G 2010 Tensor network decompositions in the presence of a global symmetry *Phys. Rev.* A **82** 050301
[31] Nguyen Q T, Schatzki L, Braccia P, Ragone M, Coles P J, Sauvage F, Larocca M and Cerezo M 2024 Theory for equivariant quantum neural networks *PRX Quantum* **5** 020328
[32] Elliott R J 1961 Phenomenological discussion of magnetic ordering in the heavy rare-earth metals *Phys. Rev.* **124** 346
[33] Fisher M E and Selke W 1980 Infinitely many commensurate phases in a simple Ising model *Phys. Rev. Lett.* **44** 1502
[34] Arizmendi C M, Rizzo A H, Epele L N and Garcia Canal C A 1991 Phase diagram of the ANNNI model in the Hamiltonian limit *Z. Phys.* B **83** 273
[35] Fumani F K, Nemati S and Mahdavifar S 2021 Quantum critical lines in the ground state phase diagram of spin-1/2 frustrated transverse-field Ising chains *Ann. Phys., Lpz.* **533** 2000384
[36] Mbeng G B, Russomanno A and Santoro G E 2020 The quantum Ising chain for beginners (arXiv:2009.09208)
[37] Beccaria M, Campostrini M and Feo A 2007 Evidence for a floating phase of the transverse ANNNI model at high frustration *Phys. Rev.* B **76** 094410
[38] Chandra A K and Dasgupta S 2007 Floating phase in the one-dimensional transverse axial next-nearest-neighbor Ising model *Phys. Rev.* E **75** 021105
[39] Suzuki S, Inoue J-I and Chakrabarti B K 2013 *Quantum Ising Phases and Transitions in Transverse Ising Models* (*Lecture Notes in Physics* vol 862) (Springer)
[40] Dutta A 2015 *Quantum Phase Transitions in Transverse Field Spin Models: From Statistical Physics to Quantum Information* (Cambridge University Press)
[41] Nagy A 2011 Exploring phase transitions by finite-entanglement scaling of MPS in the 1D ANNNI model *New J. Phys.* **13** 023015
[42] de Alcantara Bonfim O, Boechat B and Florencio J 2017 Quantum fidelity approach to the ground-state properties of the one-dimensional axial next-nearest-neighbor Ising model in a transverse field *Phys. Rev.* E **96** 042140
[43] Nemati S, Fumani F K and Mahdavifar S 2020 Comment on "Quantum fidelity approach to the ground-state properties of the one-dimensional axial next-nearest-neighbor Ising model in a transverse field" *Phys. Rev.* E **102** 016101
[44] Gray J 2018 quimb: a python package for quantum information and many-body calculations *J. Open Source Softw.* **3** 819
[45] Eisert J, Cramer M and Plenio M B 2010 *Colloquium*: Area laws for the entanglement entropy *Rev. Mod. Phys.* **82** 277
[46] Caro M C, Huang H-Y, Ezzell N, Gibbs J, Sornborger A T, Cincio L, Coles P J and Holmes Z 2023 Out-of-distribution generalization for learning quantum dynamics *Nat. Commun.* **14** 3751
[47] Sone A, Tanji A and Yamamoto N 2024 Quantum inception score (arXiv:2311.12163)
[48] Gentinetta G, Sutter D, Zoufal C, Fuller B and Woerner S 2023 Quantum kernel alignment with stochastic gradient descent *2023 IEEE Int. Conf. on Quantum Computing and Engineering (QCE)* vol 1 (IEEE) pp 256–62
[49] Larocca M *et al* 2023 Does provable absence of barren plateaus imply classical simulability? Or, why we need to rethink variational quantum computing (arXiv:2312.09121)