Learning ground states of gapped quantum Hamiltonians with Kernel Methods

Clemens Giuliani[1,2], Filippo Vicentini[1,2], Riccardo Rossi[1,2,3], and Giuseppe Carleo[1,2]

[1]Institute of Physics, École Polytechnique Fédérale de Lausanne (EPFL), CH-1015 Lausanne, Switzerland
[2]Center for Quantum Science and Engineering, École Polytechnique Fédérale de Lausanne (EPFL), CH-1015 Lausanne, Switzerland
[3]Sorbonne Université, CNRS, Laboratoire de Physique Théorique de la Matière Condensée, LPTMC, F-75005 Paris, France

Neural network approaches to approximate the ground state of quantum hamiltonians require the numerical solution of a highly nonlinear optimization problem. We introduce a statistical learning approach that makes the optimization trivial by using kernel methods. Our scheme is an approximate realization of the power method, where supervised learning is used to learn the next step of the power iteration. We show that the ground state properties of arbitrary gapped quantum hamiltonians can be reached with polynomial resources under the assumption that the supervised learning is efficient. Using kernel ridge regression, we provide numerical evidence that the learning assumption is verified by applying our scheme to find the ground states of several prototypical interacting many-body quantum systems, both in one and two dimensions, showing the flexibility of our approach.

1 Introduction

The exact simulation of quantum many-body systems on a classical computer requires computational resources that grow exponentially with the number of degrees of freedom. However, to address scientifically relevant problems such as strongly-correlated materials [1, 2] or quantum chemistry [3, 4], it is necessary to study large systems. Over the years, a variety of numerical methods have been proposed that exploit the specific structure of the system at hand and resort to approximation schemes to lower the computational cost. For example, tensor networks [5] can efficiently encode one-dimensional systems, but they face challenges in higher dimensions [6]. Quantum Monte-Carlo methods [7, 8] can give accurate results for stoquastic Hamiltonians [9], but is in general plagued by the so-called *sign-problem* [10, 11]. Finally, traditional variational methods [12, 13, 14, 8] require that the ground- or time-evolving state be well approximated by a physically-inspired parameterized function [15, 16, 17].

Recently, *data-driven* approaches for compressing the wave function based on neural networks [18, 19] and kernel methods [20, 21, 22] have been proposed. However, data-driven approaches require knowledge of the exact wave function, either from experiments or from numerically-exact calculations. In contrast, the variational principle for ground-state calculations provides a *principle-driven* approach to wave function optimization problem, and it has lead to the proposal of Neural(-network) Quantum States (NQS) [23] and Gaussian process states [24, 25]. The NQS approach produced state-of-the-art ground-state results on a variety of systems such as the J_1-J_2 spin model [26, 27, 28], atomic nuclei [29], and molecules [30]. While neural-networks are universal function approximators and can, in principle, represent any wave-function, in practice the variational energy optimization of NQS is a non-trivial task [31, 32]. Ref. [33, 34, 35] proposed schemes which solve a series of simpler supervised-learning tasks instead. When used to solve the ground-state problem with a first-order approximation of imaginary time evolution with a large time step, equivalent to the power method, we refer to this approach as the Self-Learning Power Method (SLPM). This supervised approach does not immediately solve the optimization hardness of NQS, as there is still a non-trivial optimization problem to solve at every step of the procedure.

Kernel methods are a popular class of machine learning methods for supervised learning tasks [36, 37]. They map the input data to a high-dimensional space by a non-linear transformation, with the goal of making the input data correlations approximately linear. The similarity between the input data is encoded by the kernel, whose choice is problem-dependent. When compared to neural-network approaches, kernel methods have the crucial advantage that the solution of certain optimization problems can be obtained by solving a linear system of equations.

In this article we combine the SLPM with kernel methods, rendering the optimization problem at each step of the power method straightforward. We prove the convergence of SLPM methods to the ground state of gapped quantum Hamiltonians under a learning-efficiency assumption by generalizing previous results of Ref. [38]. Considering the SLPM with kernel ridge regression, we numerically verify the learning-efficiency assumption for small quantum systems. For

larger systems, we estimate the ground-state energy directly and find a favorable system-size scaling.

The article is organized as follows: in Section 2, we recall the power method and the basics of supervised learning, setting the notation we use throughout the text. In Section 3, we introduce the SLPM, with Section 3.2 containing an in-depth theoretical analysis of its convergence properties, which is the first major result of our work. Then, after briefly recalling Kernel Ridge Regression in Section 4.1, we discuss our particular choice of kernel and numerical implementation of the SLPM in Sections 4.2 and 4.3. Finally, in Section 5, we provide comprehensive numerical results obtained on the transverse-field Ising (TFI) and antiferromagnetic Heisenberg (AFH) models in one and two dimensions, concluding with a discussion in Section 6.

2 Preliminaries

First, we briefly introduce our notation and recap some well-known concepts regarding the Power Method (PM), in Section 2.1. We then briefly overview supervised learning in Section 2.2.

Let $\hat{H}$ be a hamiltonian of a quantum system. We denote its normalized eigenstates by $|\Upsilon_k\rangle$, and we order them with respect to their corresponding eigenvalues E_k, such that $E_0 \leq E_1 \leq \cdots \leq E_{\max}$. We wish to determine the ground state $|\Upsilon_0\rangle$ and its energy E_0. The gap of $\hat{H}$ is defined as $\delta = E_1 - E_0$, and we say that the Hamiltonian is gapped if $\delta > 0$. We remark that the method remains efficient for Hamiltonians that are gapless in the thermodynamic limit, as long as the gap closes polynomially with the inverse system size.

2.1 Power Method

The PM is a procedure to find the dominant eigenvector[1] of a matrix. Following the notation of Ref. [8], we consider a gapped hamiltonian $\hat{H}$ and a constant $\Lambda \in \mathbb{R}$. The PM relies on the repeated application of the shifted Hamiltonian $\Lambda - \hat{H}$ to a trial state $|\Phi^{(0)}\rangle$. The state obtained at the $(n+1)-$th step is, therefore,

$$|\Phi^{(n+1)}\rangle = (\Lambda - \hat{H})\,|\Phi^{(n)}\rangle\,. \tag{1}$$

To make the ground state the dominant eigenvector, one must take $\Lambda > \frac{E_0+E_{\max}}{2}$. Starting from a $|\Phi^{(0)}\rangle$ with non-zero overlap with the true ground state $|\Upsilon_0\rangle$, we have that $\lim_{n\to\infty} |\Phi^{(n)}\rangle \propto |\Upsilon_0\rangle$, as the infidelity with the ground state decreases exponentially with $(\frac{\Lambda-E_1}{\Lambda-E_0})^n$ as long as $\Lambda \geq \frac{E_1+E_{\max}}{2}$ and with $(\frac{E_{\max}-\Lambda}{\Lambda-E_0})^n$ otherwise. We note that, in the case of a degenerate ground state, the PM converges to a linear combination of the degenerate eigenstates, dependent on their overlap with the initial state $|\Upsilon_0\rangle$.

The PM is widely adopted in exact diagonalization studies, where one works with vectors storing the wave-function amplitude $\langle x|\Phi^{(n)}\rangle$ for all the basis states x. This approach requires exponential resources to store the vector encoding the wave function amplitudes in a chosen basis.

[1] The dominant eigenvector is the eigenvector with the largest eigenvalue by magnitude.

2.2 Supervised Learning

Suppose we are given a set of observations $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N_s}$ of an unknown function $f^0 : \mathcal{X} \to \mathbb{R}$ where $\mathcal{X} \subseteq \mathbb{R}^n$ is the input space, $x_i \in \mathcal{X}$ are samples and $y_i = f^0(x_i) \in \mathbb{R}$ are the corresponding function values, also known as *labels*. The task of supervised learning is to find the optimal function $f^\star$ in some suitable space of Ansatz functions $\mathcal{H}$ which best describes the observations. This is done by minimizing a so-called *loss function* $\mathcal{L}$, which quantifies the distance between the predictions of each Ansatz function and the observations. The corresponding optimization problem is given by

$$f^\star \in \operatorname*{argmin}_{f\in\mathcal{H}} \mathcal{L}(f, \mathcal{D}). \tag{2}$$

Notably, supervised learning can be done with artificial neural networks, a specific instance of highly expressive parameterized maps that can typically approximate complex unknown functions with high accuracy. After fixing the architecture, the optimization problem Eq. (2) is solved by finding the optimal parameters, for example, by using gradient-based optimization methods. For a more complete overview of supervised learning, we refer the reader to one of the standard textbooks in the literature, such as Ref. [39].

3 Self-Learning Power Method

In this section, we introduce an approximate version of the PM that has polynomial complexity (Section 3.1) and provide a quantitative theoretical discussion of its convergence properties (Section 3.2). We call this approach the Self-Learning Power Method (SLPM), which is sketched in Fig. 1. The SLPM encodes the wave function with an approximate representation $\left|\Psi^{(n)}\right\rangle \approx \left|\Phi^{(n)}\right\rangle$, taken from a space $\mathcal{H}$ of functions with a polynomial memory and query complexity in the computational basis[2] to bypass the exponential computational cost of the exact PM, as discussed in the previous section. In the following, we show that the state at step $n+1$ can be computed by solving an optimization problem given the state at step n.

[2] By query complexity in the computational basis we mean that computing $\left\langle x\middle|\Psi^{(n)}\right\rangle = \Psi^{(n)}(x)$ requires polynomial resources in the system size.

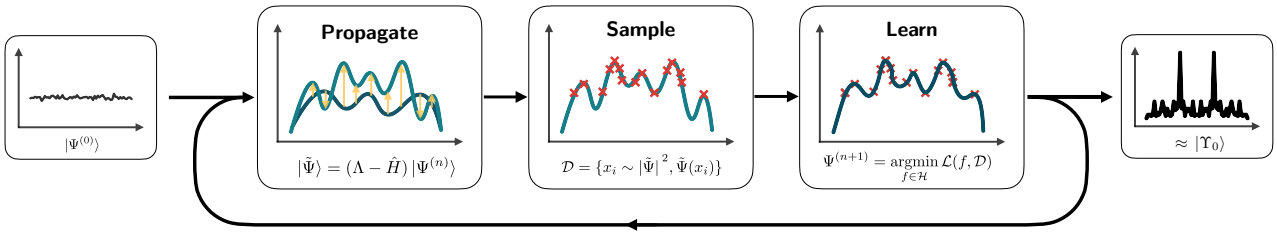

Figure 1: *Sketch of the Self-Learning Power Method.* Starting from an initial state $\Psi^{(0)}$, we propagate the state $\Psi^{(n)}$ at step $(n+1)$ to $\tilde{\Psi}$ by applying $\Lambda - \hat{H}$. Configurations x_i are sampled from $|\tilde{\Psi}(x)|^2$, paired with wave-function amplitudes $\tilde{\Psi}(x_i)$ to form a data set, which is then learned with supervised learning, obtaining a new state $\Psi^{(n+1)}$, approximating $\tilde{\Psi}$, which can again be propagated and sampled from. The procedure is repeated until convergence to a state close to the true ground state. In this paper the learning is done with kernel ridge regression (see Section 4).

3.1 Algorithm

Given $\Psi^{(n)}$ the state $\Psi^{(n+1)} \in \mathcal{H}$ is the solution of the optimization problem

$$\Psi^{(n+1)} \in \underset{f\in\mathcal{H}}{\operatorname{argmin}}\,\mathcal{L}(f, (\Lambda - \hat{H})\Psi^{(n)}), \tag{3}$$

for any similarity metric $\mathcal{L}$. In this article, we treat this optimization problem in the framework of supervised learning, which we have introduced in Section 2.2, replacing the "target" state $(\Lambda - \hat{H})\Psi^{(n)}$ with a data-set $\mathcal{D}^{(n+1)} = \{(x_i, y_i)\}$ where

$$\begin{aligned} x_i &\sim \Pi(x) = \left|\,\langle x|\Lambda - \hat{H}|\Psi^{(n)}\rangle\right|^2 \\ y_i &= \langle x_i|\Lambda - \hat{H}|\Psi^{(n)}\rangle \end{aligned} \tag{4}$$

Here $\sim$ indicates that x_i are sampled from the target distribution Π, which we do with Markov-chain Monte Carlo methods (see Appendix B for a discussion). We remark that most physical Hamiltonians are sparse and therefore the elements $\langle x|\Lambda - \hat{H}|\Psi^{(n)}\rangle$ can be queried efficiently if $\Psi^{(n)}$ can be queried efficiently in the computational basis[3].

In practice, when considering spin systems on a lattice with N sites, the wave-function $\psi(x)$ takes as inputs the bit-strings $x \in \{-1, 1\}^N$ encoding basis states $|x\rangle$. The data set contains a polynomially-large set of bit-strings x, sampled from the Born-probability distribution $|\psi(x)|^2$, and associated with their corresponding amplitude $\psi(x)$.

We remark that if $\mathcal{H}$ spans the whole Hilbert space and if the data set contains all (exponentially-many) bit-strings, the solution to the optimization problem given by Eq. (3) would match the PM exactly. By truncating the data-set size and considering only a subset of all possible wave functions, the solution is only approximate and therefore there is a finite difference between an exact step of the PM and the approximate procedure. We quantify this difference with the *step infidelity*, which we define as

[3]More in detail, one has to calculate $\sum_{x'} \langle x|\Lambda - \hat{H}|x'\rangle\,\langle x'|\Psi^{(n)}\rangle$. Physical Hamiltonians, in general, have a polynomial number of nonzero terms $\langle x|\hat{H}|x'\rangle \neq 0$, therefore the sum over x' runs over a polynomial number of elements and this query is efficient.

Definition 1 (Step Infidelity). *Let $\left|\Psi^{(n+1)}\right\rangle$ be the state after step n of the noisy power method. We define the step infidelity as*

$$I^{(n)} := 1 - \mathcal{F}\left(\Psi^{(n+1)}, (\Lambda - \hat{H})\Psi^{(n)}\right) \tag{5}$$

where $\mathcal{F}(\psi, \phi) = \frac{|\langle\psi|\phi\rangle|^2}{\|\phi\|^2\|\psi\|^2}$ is the fidelity between two states.

3.2 Discussion of convergence properties

The SLPM is approximating the propagation of the state

$$|\Psi^{(n+1)}\rangle \approx (\Lambda - \hat{H})\,|\Psi^{(n)}\rangle\,, \tag{6}$$

which is instead exact when using the standard PM. It is well known (see Section 2.1) that the PM converges exponentially fast to the dominant eigenstate. In this subsection, we discuss how the noise introduced by a non-zero step-infidelity affects the convergence. To derive quantitative bounds, we prove that if this noise is small enough, it does not significantly hinder the convergence to the ground state as the relative error of the energy is bounded (as we will see in Eq. (13)). The discussion is based on Ref. [38], but has been adapted to the language of computational physics.

Power method with noise

We consider the general case where, at every step of the PM, a small noise term $|\Delta\rangle$ is added to the state. In the setting we are interested in, this noise arises from the self-learning procedure, but we wish to keep the theoretical treatment general and accordingly we make no assumptions on the origin of the noise. Formally, we define the power method with noise as:

Definition 2 (Noisy Power Method). *Take an initial state $|\Psi^{(0)}\rangle$. Step $(n+1)$ of the noisy power method is defined recursively as*

$$|\Psi^{(n+1)}\rangle = \gamma^{(n)}\left[(\Lambda - \hat{H})\,|\Psi^{(n)}\rangle + |\Delta^{(n)}\rangle\right] \tag{7}$$

where $|\Delta^{(n)}\rangle$ *is a additive noise term, which, without loss of generality*[4]*, is taken such that* $\langle\Delta^{(n)}|\Lambda - \hat{H}|\Psi^{(n)}\rangle = 0$*. The factor* $\gamma^{(n)} \in \mathbb{C}$ *captures both a potential drift in the global phase as well as the normalization.*

If $I^{(n)} = 0$, the noise must be zero as well, while in the general case the step-infidelity bounds the amplitude of the noise term according to

$$\frac{\left\|\Delta^{(n)}\right\|}{\left\|\Psi^{(n)}\right\|} \leq (\Lambda - E_0)\sqrt{\frac{I^{(n)}}{1 - I^{(n)}}}. \tag{8}$$

Theorem 1 (Convergence of the noisy power method)**.** *Let* $|\Upsilon_0\rangle$ *represent the ground state of the Hamiltonian* $\hat{H}$*. Take* $\Lambda \geq \frac{E_1+E_{\max}}{2}$ *and assume that the initial state* $|\Psi^{(0)}\rangle$ *and noise* $|\Delta^{(n)}\rangle$ *respect the conditions*

$$\frac{\left|\langle\Upsilon_0|\Delta^{(n)}\rangle\right|}{\left\|\Psi^{(n)}\right\|} \leq \frac{\delta}{5}\frac{\left|\langle\Upsilon_0|\Psi^{(0)}\rangle\right|}{\left\|\Psi^{(0)}\right\|} \tag{9}$$

$$\frac{\left\|\Delta^{(n)}\right\|}{\left\|\Psi^{(n)}\right\|} \leq \frac{\delta}{5}\varepsilon, \tag{10}$$

at every step n *of the noisy power method for some* $\varepsilon < \frac{1}{2}$*. Then there exists a minimum number of steps* $M \leq \frac{4}{1-\frac{\Lambda-E_1}{\Lambda-E_0}} \log\left(\varepsilon^{-1}\sqrt{\frac{1-\mathcal{F}(\Upsilon_0,\Psi^{(0)})}{\mathcal{F}(\Upsilon_0,\Psi^{(0)})}}\right)$ *such that for all steps* $n \geq M$ *we have* $\sqrt{\frac{1-\mathcal{F}(\Upsilon_0,\Psi^{(n)})}{\mathcal{F}(\Upsilon_0,\Psi^{(n)})}} \leq \varepsilon$*.*

A proof adapted from Ref. [38] is given in Appendix A. Eq. (9) requires that, if the initial state $\Psi^{(0)}$ has an exponentially small overlap with the ground state (as in random initialization [38]), the noise parallel to ground state wave-function must also be exponentially small. Instead, the assumption of Eq. (10) requires that the noise amplitude be smaller than ε. As the final infidelity is bounded by ε^2 we want to choose the smallest ε possible. For a given step infidelity $I^{(n)}$ the smallest ε we can guarantee using Eq. (8) is given by

$$\varepsilon^\star = \frac{5}{1-\frac{\Lambda-E_1}{\Lambda-E_0}} \max_n \sqrt{\frac{I^{(n)}}{1-I^{(n)}}}. \tag{11}$$

Requiring $\varepsilon^\star < \frac{1}{2}$, it is possible to show that the step-infidelity must be sufficiently small and satisfy $I^{(n)} < \frac{1}{100}(1-\frac{\Lambda-E_1}{\Lambda-E_0})^2$.

When those requirements are satisfied, Theorem 1 states that in a number of steps M, logarithmic in both the initial overlap $\left|\langle\Upsilon_0|\Psi^{(0)}\rangle\right|$ and in the final infidelity ε^2, we reach a state with at most

$$\mathcal{I} = 1 - \mathcal{F}(\Upsilon_0, \Psi^{(M)}) \leq \frac{1-\mathcal{F}(\Upsilon_0,\Psi^{(M)})}{\mathcal{F}(\Upsilon_0,\Psi^{(M)})} \leq \varepsilon^2. \tag{12}$$

[4]In the case that $|\Delta\rangle$ is not orthogonal to $|\tilde{\Psi}\rangle \coloneqq (\Lambda - \hat{H})|\Psi\rangle$ we can replace it with $|\Delta_\perp\rangle \coloneqq \frac{\langle\tilde{\Psi}|\tilde{\Psi}\rangle|\Delta\rangle - \langle\tilde{\Psi}|\Delta\rangle|\tilde{\Psi}\rangle}{\langle\tilde{\Psi}|\tilde{\Psi}\rangle + \langle\tilde{\Psi}|\Delta\rangle}$, then we have that $\langle\Delta_\perp|\tilde{\Psi}\rangle = 0$ and $|\tilde{\Psi}\rangle + |\Delta_\perp\rangle \propto \left|\tilde{\Psi}\right\rangle + |\Delta\rangle$, and the resulting proportionality constant is absorbed into γ.

This state has an accuracy on the ground-state energy given by the relative error,

$$\epsilon_{\text{rel}} \coloneqq \frac{\langle\hat{H}\rangle - E_0}{|E_0|} \leq \frac{E_{max} - E_0}{|E_0|}\mathcal{I} \tag{13}$$

For simplicity, the theorem assumes that the noise bounds are constant throughout the run-time of the noisy power method. While this might not be the case in practice, it is easy to generalize the result to a varying ε. Doing so, one finds that the first assumption (Eq. (9)) is necessary to start the method while asymptotically, the bound is given only by the latter assumption (this is discussed more in detail in Appendix C).

Self-Learning Power Method

While the discussion of the noisy power method convergence so far is general, we now contextualize it to the case of the SLPM. To do so we assume that the *learning is efficient*, precisely defined as follows.

Definition 3 (Efficient supervised learning)**.** *We say that the supervised learning is efficient if its step-infidelity is of the order of* $1/{N_S}^\alpha$ *for some* $\alpha > 0$*, where* N_S *is the size of the data-set.*

As a consequence of Theorem 1, summing up the discussion of the convergence properties, we present the following corollary for the convergence of the SLPM:

Corollary 1 (Convergence of the self-learning power method)**.** *Let* $\hat{H}$ *be a gapped Hamiltonian, take* $\Lambda \geq \frac{E_1+E_{\max}}{2}$*, and assume that*

- *The supervised learning is efficient, meaning that* $I^{(n)} \leq \frac{A}{{N_S}^\alpha} \leq \frac{1}{100}(1-\frac{\Lambda-E_1}{\Lambda-E_0})^2$ *for* $A, \alpha > 0$*.*
- *The error parallel to the ground state is bounded by* $\frac{\left|\langle\Upsilon_0|\Delta^{(n)}\rangle\right|}{\left\|\Psi^{(n)}\right\|} \leq \frac{\delta}{5}\frac{\left|\langle\Upsilon_0|\Psi^{(0)}\rangle\right|}{\left\|\Psi^{(0)}\right\|}$*.*

Then the final infidelity $\mathcal{I}$ *is bounded by*

$$\mathcal{I} \leq \frac{25}{\left(1-\frac{\Lambda-E_1}{\Lambda-E_0}\right)^2}\frac{A}{{N_S}^\alpha}. \tag{14}$$

and the error on the ground-state energy of $\hat{H}$ *is of the order of*

$$\epsilon_{\text{rel}} \lesssim \frac{1}{\delta^2 {N_S}^\alpha}. \tag{15}$$

Therefore, assuming the supervised learning is efficient, it is possible to consider a polynomially large data set to compute the ground-state energy of a gapped Hamiltonian with a polynomial cost. The same holds for gapless Hamiltonians, as long as the gap closes polynomially with the inverse system size, as we show in Appendix A.

4 The Self-learning Power Method with Kernel Ridge Regression

There are several practical ways to implement the SLPM defined in Section 3, by solving the supervised learning problem of learning the next state (Eq. (3)) with a suitable approach. In this section, we specialize our discussion on realizing the SLPM with a kernel method called Kernel Ridge Regression.

4.1 Kernel Ridge Regression

Given that we are discussing a kernel method we start with a brief, formal definition of the *kernel*. A positive definite kernel k (named *Mercer Kernel* after the author of Ref. [40]), is a function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, where $\mathcal{X} \subseteq \mathbb{R}^n$, with the following properties:

- It is symmetric: $k(x, y) = k(y, x)$
- For any set $\{x_1, \ldots, x_n\} \subseteq \mathcal{X}$ the kernel matrix K with entries $K_{ij} = k(x_i, x_j)$ is positive semi-definite.

It can be shown that every kernel uniquely defines a function space [41], the so-called *Reproducing Kernel Hilbert Space* (RKHS)

$$\mathcal{H}_k = \{f(\cdot) = \sum_{i=1}^{\ell} w_i k(\cdot, x_i) \,|\, \ell \in \mathbb{N}, w_i \in \mathbb{R}, x_i \in \mathcal{X}\} \tag{16}$$

where, for two functions $f, g \in \mathcal{H}_k$, $f(x) = \sum_{i=1}^{\ell} \alpha_i k(x, x_i)$, $g(x) = \sum_{j=1}^{m} \beta_j k(x, y_j)$ the inner product is given by $\langle f, g\rangle_{\mathcal{H}_k} = \sum_{i=1}^{\ell}\sum_{j=1}^{m} \alpha_i \beta_j k(x_i, y_j)$.

The RKHS can be used as space of Ansatz functions for the supervised learning problem Eq. (2). When using the regularized least squares loss (ridge loss)

$$\mathcal{L}(f, \mathcal{D}) = \sum_i |f(x_i) - y_i|^2 + \lambda \|f\|^2_{\mathcal{H}_k}, \tag{17}$$

this approach is called Kernel Ridge Regression (see e.g. Ref. [36] for more details). Here $\lambda \|f\|^2_{\mathcal{H}_k}$ is the regularization term with $\|f\|^2_{\mathcal{H}_k} = \langle f, f\rangle_{\mathcal{H}_k}$ and $\lambda \geq 0$. It can be shown that, in this setting, the supervised learning problem has an analytical solution of the form [42, 43]

$$f^\star(x) = \sum_{i=1}^{N_S} w_i k(x, x_i), \tag{18}$$

where the sum only goes over the finitely many training samples and the weights w_i are uniquely determined by solving the linear system of equations

$$\sum_{j=1}^{N_S} (k(x_i, x_j) + \lambda \delta_{i,j})\, w_j = y_i. \tag{19}$$

In the case where $k(x_i, x_j)$ is singular, there is an infinite number of w that satisfy Eq. (19); The presence of the infinitesimal regularization term λ in this equation ensures that we choose the solution with the minimal norm $\|f\|_{\mathcal{H}_k}$.

4.2 Implementation

The most straightforward approach would be to learn the amplitudes $\Psi^{(n+1)}$ with functions from the reproducing kernel Hilbert space $\mathcal{H}_k$ (Eq. (16)) using kernel ridge regression. However, as is commonly done with NQS, and in a previous work using a different kernel method (Ref. [20]), we use the method to learn the log-amplitudes $\log \Psi^{n+1}(x)$ instead. The loss function remains the same, but the data-set $D^{(n+1)} = \{(x_i, y_i)\}$ is changed to include log-amplitudes as labels,

$$\begin{aligned} x_i &\sim \Pi(x) \\ y_i &= \log \langle x_i | \Lambda - \hat{H} | \Psi^{(n)} \rangle \end{aligned} \tag{20}$$

and we have to take the exponential to make predictions

$$\Psi^{n+1}(x) = \exp\left\{\sum_{i=1}^{N_S} w_i k(x, x_i)\right\}, \tag{21}$$

where the weights w_i are found through Eq. (19) for the modified data-set of Eq. (20).

We remark that this prediction is different from what one would obtain with relevance-vector regression [44, 45] used in Refs. [20, 21].

While both methods aim to minimize the mean-squared error on the training dataset (first term in Eq. (17)), they favour different solutions. The KRR finds predictions with small RKHS norm $\|f\|_{\mathcal{H}_k}$ due to the regularization term (favouring smooth $\log \Psi$). The relevance-vector regression instead favours solutions which have small weights w, due to a zero-mean gaussian prior on them [5].

We make one final approximation for the simulations in this article to reduce the computational cost. We assume that the distribution of the previous state $\Psi^{(n)}$ is sufficiently close to the distribution of the propagated state $(\Lambda - \hat{H})\Psi^{(n)}$, and sample from

$$\Pi(x) = \left|\Psi^{(n)}(x)\right|^2, \tag{22}$$

instead of $|(\Lambda - \hat{H})\Psi^{(n)}(x)|^2$, reducing the number of evaluations of $\Psi^{(n)}$ required.

4.3 Kernel choice and symmetries

The properties of the kernel $k(\cdot, \cdot)$ are fundamental, as they are reflected on the encoded wave-function. For example, discrete symmetries can be explicitly

[5] The variance of the gaussian prior is given by hyperparmeters found by a type II maximum likelihood optimization[46], which, in particular, can force some weights to be zero.

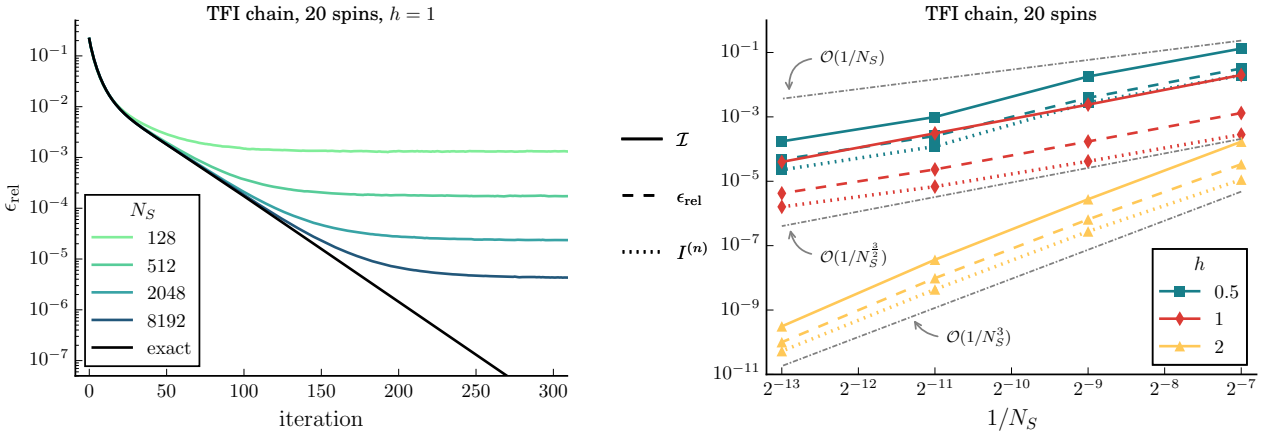

Figure 2: Convergence of the Self-learning power method for the TFI model on a one-dimensional chain of $N = 20$ spins. **(left panel)**: Relative error of the predicted energy with the true ground state energy as a function of the number of iterations n, compared to the power method for $h = 1$. Starting from an initial uniform superposition state, after a certain number of iterations, a steady state is reached, with an energy that becomes more accurate with increasing data-set size N_S, taking the average over 100 runs. **(right panel)**: Final state convergence. Plotted are $I^{(n)}$: step infidelity of learning the final state (see Definition 1), $\mathcal{I}$: infidelity of the final state with the true ground state (defined in Eq. (12)), and ϵ_{rel}: relative error of the predicted energy of the final state (defined in Eq. (13)) after convergence of the self-learning power method, as a function of the number of samples in the data-set N_S. Statistical error bars are smaller than the markers and have been omitted from the plot.

enforced by constructing a kernel that averages the output over all possible input permutations.

In this article, we consider a symmetric kernel of the form

$$k(x, y) = \frac{1}{|G|} \sum_{g \in G} \sigma\Big(\frac{1}{L} \sum_{i=1}^{L} (g\, x)_i\, y_i\Big), \qquad (23)$$

where x, y are vectors which encode the basis states, $\sigma(x) = x \arcsin(\gamma x)$ is our choice of non-linear function. We remark that by taking $\gamma \approx 0.5808$, this kernel corresponds to a symmetrized Restricted Boltzmann Machine (RBM) in the infinite hidden-neuron density limit through the neural tangent kernel theory [47]. Details of this connection are explained in Appendix D but are not needed for the discussion. To contain the computational cost, when simulating lattice systems, we consider the group of all possible translations rather than taking the full space-group of the lattice. Additionally, spin-inversion ($\mathbb{Z}_2$) symmetry can be enforced by choosing an even non-linear function σ.

5 Numerical experiments

To numerically investigate the viability of the SLPM we benchmark it on the transverse-field Ising model (TFI) with periodic boundary conditions in one and two dimensions, and on the antiferromagnetic Heisenberg model (AFH) on the square lattice.

5.1 TFI model in one dimension at fixed system size

The Hamiltonian of the TFI model is

$$\hat{H}_{TFI} = \sum_{\langle i,j \rangle} \hat{\sigma}_i^z \hat{\sigma}_j^z - h \sum_i \hat{\sigma}_i^x, \qquad (24)$$

where $\hat{\sigma}_i^{x,y,z}$ are Pauli matrices on site i, $\langle i, j \rangle$ iterates over all nearest-neighbor pairs, and h is the strength of the external field in the transverse direction.

We start by considering a 1-dimensional chain of 20 spins with transverse field $h \in \{0.5, 1, 2\}$. The initial state is always taken to be $\left|\Psi^{(0)}\right\rangle \propto \sum_x |x\rangle$, the uniform superposition of all computational basis states, and we fix $\Lambda = 1$.

In the left panel of Fig. 2, we plot the relative error of the energy with respect to the ground-state value as a function of the number of iterations for $h = 1$. We compare the SLPM for several data set sizes N_s against the exact version, observing a crossover from an initial regime where the effect of the noise is negligible. The SLPM closely matches the exact one to a regime where the noise dominates and the bound given by Theorem 1 prevents further improvements and a steady-state is reached. As expected, the number of steps at which we observe the crossover depends on the number of samples.

5.2 Numerical verification of the efficient-learning assumption

In the right panel of Fig. 2, we numerically prove the assumption of efficient learning by showing that the

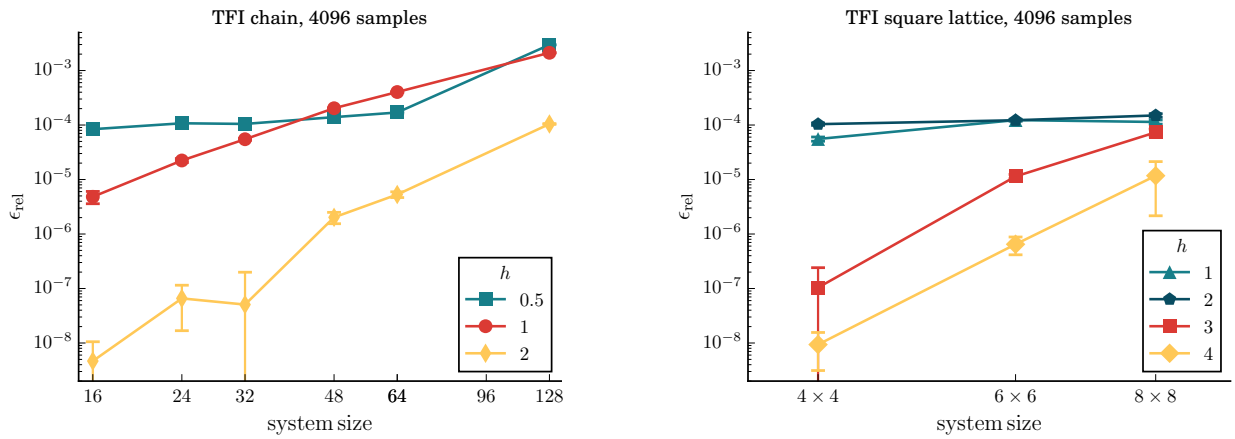

Figure 3: Scaling of the SLPM ground-state energy relative error as a function of the system size for 1D **(left panel)** and 2D **(right panel)** periodic lattices of the TFI Hamiltonian with varying values of the transverse field h. The number of samples is fixed in all simulations at $N_S = 4096$ and the energies are estimated by taking 2^{20} samples from the final state. The horizontal axes use a logarithmic scale of the total number of spins. The reference energies for the relative error are computed analytically for 1-D systems, with exact diagonalization for 2-D up to 40 sites (using the code from Ref. [48, 49]) and with Quantum Monte Carlo for larger systems (loop algorithm from the ALPS library [50, 51]). They are provided in Appendix C.

step-infidelity at $n = 300$ is compatible with a power law $I^n \propto N_S^{-\alpha}$, where the exponent α depends on the parameters of the Hamiltonian. In the same figure, we also report that the best relative error follows a similar power law with the same exponent. Interestingly, we see that the scaling exponent α of the step-infidelity and relative error are degraded for values of h below the critical point ($h = 1$ in this case). This shows that Corollary 1 is valid and therefore gives further grounding to the theoretical analysis we carried out in Section 3.2. In Fig. 7 of the appendix we show the same for the TFI in two dimensions and for the AFH in one and two dimensions.

5.3 TFI model in one and two dimensions and scaling as function of system size

Continuing, we investigate the scaling of the accuracy of the SLPM at increasing system sizes. In Fig. 3 we plot the relative error of the ground-state energy for 1D (left panel) and 2D (right panel) periodic lattices. The data-set size is kept fixed at $N_S = 4096$ for all system sizes. In both cases, for values of the transverse field h above the critical point[6] we observe a behavior consistent with a power-law dependency of the relative error with the system size. As the Hilbert-space size is increasing exponentially, this means that a tiny fraction of the Hilbert space is sufficient to compute the ground-state energy accurately in a few hundred steps. Evidently, at the critical point, the gap of the Hamiltonian becomes smaller and therefore we need to perform more iterations ($M \approx 2000$) to converge.

As in the previous simulations, the scaling with the system size degrades for values of the transverse field below the critical point. This is linked to a *less efficient* supervised learning of the state (in terms of the number of samples) at every step of the SLPM, and is probably related to poor generalization properties of the kernel in this regime. In principle, we expect that by choosing a different kernel function, it should be possible to improve the learning efficiency and, therefore, the algorithm's overall performance.

5.4 AFH model in one and two dimensions

In addition to the TFI, we also benchmark the SLPM against the antiferromagnetic Heisenberg model, whose Hamiltonian is given by

$$\hat{H}_{AFH} = \sum_{\langle i,j \rangle} \hat{\sigma}_i^x \hat{\sigma}_j^x + \hat{\sigma}_i^y \hat{\sigma}_j^y + \hat{\sigma}_i^z \hat{\sigma}_j^z, \qquad (25)$$

where we assume periodic boundary conditions. The AFH hamiltonian is gapless in the thermodynamic limit. However, the gap is nonvanishing on finite lattices, and the SLPM can be applied. The ground state has a well-known sign structure, which can be accounted for by rotating the Hamiltonian according to the Marshall sign rule[57]. The SLPM is then used to learn the amplitudes. To simplify the problem, the ground-state search is constrained to the symmetry sector with zero magnetization by introducing a proper constraint in the sampling step used to generate the data set. For the simulations of the AFH we fix $\Lambda = 0$. In Fig. 4, we show the dependence of the

[6] the critical point of the TFI Hamiltonian in the thermodynamic limit is $h = 1$ in 1-D chains and $h \approx 3.044$ for 2-D square lattices [52, 53, 54, 55, 56].

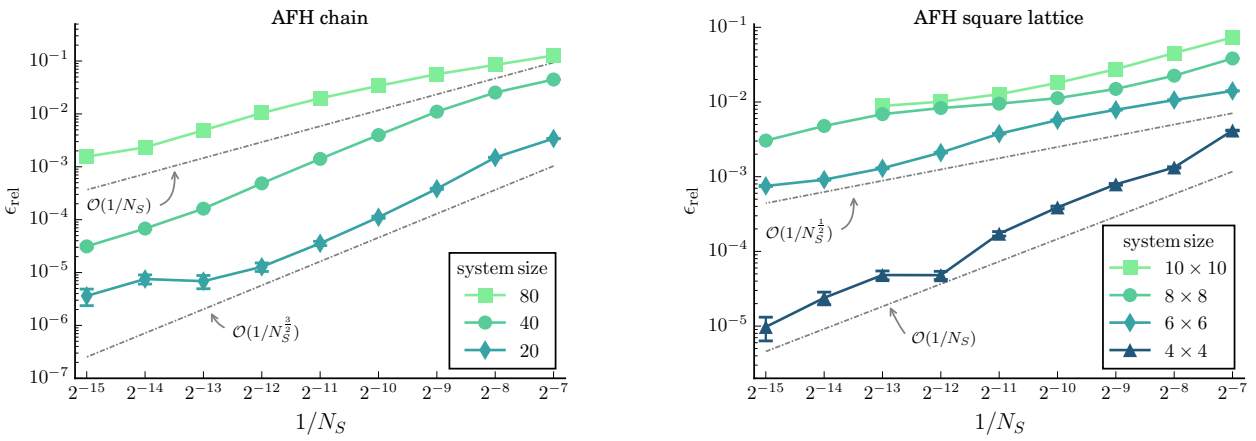

Figure 4: Scaling of the SLPM ground-state energy relative error as a function of the number of samples in the data-set size for 1D **(left panel)** and 2D **(right panel)** periodic lattices of the AFH Hamiltonian for different systems sizes. Estimates and reference values computed as in Fig. 3.

final relative error of the energy as a function of the number of samples in the data set, in the left panel for a one-dimensional chain and in the right panel for two-dimensional square lattices. Both result in a power-law-like scaling, with an exponent lower than that of the TFI, meaning that supervised learning is less efficient and requires us to use more samples to get a comparable accuracy.

6 Discussion

In this article, we have presented a kernel-method realization of the SLPM that can be used to find the ground state of gapped quantum hamiltonians by solving a series of quadratic optimization problems. We have shown that if the supervised learning requires a polynomial number of samples at each step of the power method, a logarithmic number of steps in both the initial overlap and final infidelity of the SLPM is sufficient to reach a ground state infidelity that scales polynomially with the number of samples. In our numerical experiments, we have considered a relatively simple kernel that is reasonably cheap to evaluate and enforces physical symmetries on the ground state wave-function. For the TFI and AFH models in one and two dimensions we have numerically verified that the efficient-learning assumption is valid for kernel ridge regression using this kernel, at least for the small system sizes for which the exact infidelity computation is tractable. For larger system sizes, a direct computation shows a favorable scaling of the energy relative error in terms of number of samples and as a function of the system size.

Our kernel ridge regression approach is ultimately limited by the number of samples as the computational resources needed to compute and store the kernel matrix scale quadratically with the data-set size, while the solution of the linear system of equations scales cubically. Therefore, in practice the number of samples is at most of the order of 10^5 due to hardware limitations. Algorithmic improvements such as re-using information from the matrix decomposition of the previous step when most of the samples remain the same or using iterative solvers enabling parallelization could ease some of these limitations. Nevertheless, we believe that the primary focus for improvements has to be laid on increasing the efficiency of the supervised learning. This can be done either by developing kernels with superior generalization properties for the problems at hand, with kernel methods which do not require to compute the full kernel matrix, like those used in Refs. [20, 21], or by using methods based on neural networks.

Possible extensions of this work include the application of the SLPM to non-stoquastic Hamiltonians. This can be achieved by learning the sign structure or the phase of the wave-function in addition to the absolute value of the amplitude. It might be worth exploring a more general Ansatz using pseudo-kernels [58], where the absolute value and phase of the wave-function amplitude are learned simultaneously.

The code used to run the simulations in this article can be found in Ref. [59].

Acknowledgements

The authors would like to thank George Booth for insightful discussions. This work was supported by the Swiss National Science Foundation under Grant No. 200021_200336.

References

[1] P. A. Lee, N. Nagaosa, and X.-G. Wen. "Doping a mott insulator: Physics of high-temperature superconductivity". Rev. Mod. Phys. **78**, 17–85 (2006). doi: 10.1103/RevModPhys.78.17.

[2] A. Kopp and S. Chakravarty. "Criticality in correlated quantum matter". Nature Physics **1**, 53–56 (2005). doi: 10.1038/nphys105.

[3] P. O. Dral. "Quantum chemistry in the age of machine learning". The Journal of Physical Chemistry Letters **11**, 2336–2347 (2020). doi: 10.1021/acs.jpclett.9b03664.

[4] S. McArdle, S. Endo, A. Aspuru-Guzik, S. C. Benjamin, and X. Yuan. "Quantum computational chemistry". Reviews of Modern Physics**92** (2020). doi: 10.1103/revmodphys.92.015003.

[5] S. R. White. "Density matrix formulation for quantum renormalization groups". Phys. Rev. Lett. **69**, 2863–2866 (1992). doi: 10.1103/PhysRevLett.69.2863.

[6] F. Verstraete and J. I. Cirac. "Renormalization algorithms for quantum-many body systems in two and higher dimensions" (2004) arXiv:cond-mat/0407066.

[7] D. Ceperley and B. Alder. "Quantum monte carlo". Science **231**, 555–560 (1986). doi: 10.1126/science.231.4738.555.

[8] F. Becca and S. Sorella. "Quantum monte carlo approaches for correlated systems". Cambridge University Press. (2017). doi: 10.1017/9781316417041.

[9] S. Bravyi, D. DiVincenzo, R. Oliveira, and B. Terhal. "The complexity of stoquastic local hamiltonian problems". Quantum Information and Computation **8**, 361–385 (2008). doi: 10.26421/qic8.5-1.

[10] E. Loh Jr, J. Gubernatis, R. Scalettar, S. White, D. Scalapino, and R. Sugar. "Sign problem in the numerical simulation of many-electron systems". Phys. Rev. B **41**, 9301–9307 (1990). doi: 10.1103/PhysRevB.41.9301.

[11] M. Troyer and U.-J. Wiese. "Computational complexity and fundamental limitations to fermionic quantum monte carlo simulations". Phys. Rev. Lett. **94**, 170201 (2005). doi: 10.1103/PhysRevLett.94.170201.

[12] R. Jastrow. "Many-body problem with strong forces". Phys. Rev. **98**, 1479–1484 (1955). doi: 10.1103/PhysRev.98.1479.

[13] J. Bardeen, L. N. Cooper, and J. R. Schrieffer. "Theory of superconductivity". Phys. Rev. **108**, 1175–1204 (1957). doi: 10.1103/PhysRev.108.1175.

[14] S. Sorella. "Green function monte carlo with stochastic reconfiguration". Phys. Rev. Lett. **80**, 4558–4561 (1998). doi: 10.1103/PhysRevLett.80.4558.

[15] H. Yokoyama and H. Shiba. "Variational Monte-Carlo studies of Hubbard model. I". Journal of the Physical Society of Japan **56**, 1490–1506 (1987). doi: 10.1143/JPSJ.56.1490.

[16] C. Gros, R. Joynt, and T. M. Rice. "Antiferromagnetic correlations in almost-localized fermi liquids". Phys. Rev. B **36**, 381–393 (1987). doi: 10.1103/PhysRevB.36.381.

[17] C. Gros. "Superconductivity in correlated wave functions". Phys. Rev. B **38**, 931–934 (1988). doi: 10.1103/PhysRevB.38.931.

[18] J. Carrasquilla and R. G. Melko. "Machine learning phases of matter". Nature Physics **13**, 431–434 (2017). doi: 10.1038/nphys4035.

[19] G. Torlai, G. Mazzola, J. Carrasquilla, M. Troyer, R. Melko, and G. Carleo. "Neural-network quantum state tomography". Nature Physics **14**, 447–450 (2018). doi: 10.1038/s41567-018-0048-5.

[20] A. Glielmo, Y. Rath, G. Csányi, A. De Vita, and G. H. Booth. "Gaussian process states: A data-driven representation of quantum many-body physics". Phys. Rev. X **10**, 041026 (2020). doi: 10.1103/PhysRevX.10.041026.

[21] Y. Rath, A. Glielmo, and G. H. Booth. "A bayesian inference framework for compression and prediction of quantum states". The Journal of Chemical Physics **153**, 124108 (2020). doi: 10.1063/5.0024570.

[22] D. Luo and J. Halverson. "Infinite neural network quantum states: entanglement and training dynamics". Machine Learning: Science and Technology **4**, 025038 (2023). doi: 10.1088/2632-2153/ace02f.

[23] G. Carleo and M. Troyer. "Solving the quantum many-body problem with artificial neural networks". Science **355**, 602–606 (2017). doi: 10.1126/science.aag2302.

[24] Y. Rath and G. H. Booth. "Quantum gaussian process state: A kernel-inspired state with quantum support data". Phys. Rev. Research **4**, 023126 (2022). doi: 10.1103/PhysRevResearch.4.023126.

[25] Y. Rath and G. H. Booth. "Framework for efficient ab initio electronic structure with gaussian process states". Phys. Rev. B **107**, 205119 (2023). doi: 10.1103/PhysRevB.107.205119.

[26] Y. Nomura and M. Imada. "Dirac-type nodal spin liquid revealed by refined quantum many-body solver using neural-network wave function,

correlation ratio, and level spectroscopy". Phys. Rev. X **11**, 031034 (2021). doi: 10.1103/PhysRevX.11.031034.

[27] C. Roth and A. H. MacDonald. "Group convolutional neural networks improve quantum state accuracy" (2021) arXiv:2104.05085.

[28] N. Astrakhantsev, T. Westerhout, A. Tiwari, K. Choo, A. Chen, M. H. Fischer, G. Carleo, and T. Neupert. "Broken-symmetry ground states of the heisenberg model on the pyrochlore lattice". Phys. Rev. X **11**, 041021 (2021). doi: 10.1103/PhysRevX.11.041021.

[29] A. Lovato, C. Adams, G. Carleo, and N. Rocco. "Hidden-nucleons neural-network quantum states for the nuclear many-body problem". Phys. Rev. Res. **4**, 043178 (2022). doi: 10.1103/PhysRevResearch.4.043178.

[30] T. Zhao, J. Stokes, and S. Veerapaneni. "Scalable neural quantum states architecture for quantum chemistry". Machine Learning: Science and Technology (2023). doi: 10.1088/2632-2153/acdb2f.

[31] T. Westerhout, N. Astrakhantsev, K. S. Tikhonov, M. I. Katsnelson, and A. A. Bagrov. "Generalization properties of neural network approximations to frustrated magnet ground states". Nature Communications **11**, 1593 (2020). doi: 10.1038/s41467-020-15402-w.

[32] A. Szabó and C. Castelnovo. "Neural network wave functions and the sign problem". Phys. Rev. Research **2**, 033075 (2020). doi: 10.1103/PhysRevResearch.2.033075.

[33] D. Kochkov and B. K. Clark. "Variational optimization in the ai era: Computational graph states and supervised wave-function optimization" (2018) arXiv:1811.12423.

[34] B. Jónsson, B. Bauer, and G. Carleo. "Neural-network states for the classical simulation of quantum computing" (2018) arXiv:1808.05232.

[35] H. Atanasova, L. Bernheimer, and G. Cohen. "Stochastic representation of many-body quantum states". Nature Communications**14** (2023). doi: 10.1038/s41467-023-39244-4.

[36] J. Shawe-Taylor and N. Cristianini. "Kernel methods for pattern analysis". Cambridge university press. (2004). doi: 10.1017/CBO9780511809682.

[37] T. Hofmann, B. Schölkopf, and A. J. Smola. "Kernel methods in machine learning". The Annals of Statistics **36**, 1171 – 1220 (2008). doi: 10.1214/009053607000000677.

[38] M. Hardt and E. Price. "The noisy power method: A meta algorithm with applications". In Advances in Neural Information Processing Systems. Volume 27. (2014). url: https://proceedings.neurips.cc/paper/2014/file/729c68884bd359ade15d5f163166738a-Paper.pdf.

[39] S. Russell and P. Norvig. "Artificial intelligence: A modern approach". Prentice Hall Press. (2020). 4th edition.

[40] J. Mercer. "Functions ofpositive and negativetypeand theircommection with the theory ofintegral equations". Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character **209**, 415–446 (1909). doi: 10.1098/rsta.1909.0016.

[41] N. Aronszajn. "Theory of reproducing kernels". Transactions of the American Mathematical Society **68**, 337–404 (1950). doi: 10.1090/s0002-9947-1950-0051437-7.

[42] G. S. Kimeldorf and G. Wahba. "A Correspondence Between Bayesian Estimation on Stochastic Processes and Smoothing by Splines". The Annals of Mathematical Statistics **41**, 495 – 502 (1970). doi: 10.1214/aoms/1177697089.

[43] B. Schölkopf, R. Herbrich, and A. J. Smola. "A generalized representer theorem". In Computational Learning Theory. Pages 416–426. Springer Berlin Heidelberg (2001). doi: 10.1007/3-540-44581-1_27.

[44] M. Tipping. "The relevance vector machine". In S. Solla, T. Leen, and K. Müller, editors, Advances in Neural Information Processing Systems. Volume 12. MIT Press (1999). url: https://proceedings.neurips.cc/paper_files/paper/1999/file/f3144cefe89a60d6a1afaf7859c5076b-Paper.pdf.

[45] M. E. Tipping. "Sparse bayesian learning and the relevance vector machine". J. Mach. Learn. Res. **1**, 211–244 (2001). url: https://www.jmlr.org/papers/volume1/tipping01a/tipping01a.pdf.

[46] M. E. Tipping and A. C. Faul. "Fast marginal likelihood maximisation for sparse bayesian models". In Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics. Pages 276–283. PMLR (2003). url: http://proceedings.mlr.press/r4/tipping03a/tipping03a.pdf.

[47] A. Jacot, F. Gabriel, and C. Hongler. "Neural tangent kernel: Convergence and generalization in neural networks". In Advances in Neural Information Processing Systems. Volume 31. (2018). url: https://proceedings.neurips.cc/paper/2018/file/5a4be1fa34e62bb8a6ec6b91d2462f5a-Paper.pdf.

[48] T. Westerhout. "lattice-symmetries: A package for working with quantum many-body bases". Journal of Open Source Software **6**, 3537 (2021). doi: 10.21105/joss.03537.

[49] T. Westerhout. "SpinED: User-friendly exact diagonalization package for quantum many-body systems". url: https://github.com/twesterhout/spin-ed.

[50] A. Albuquerque et al. "The alps project release 1.3: Open-source software for strongly correlated systems". Journal of Magnetism and Magnetic Materials **310**, 1187–1193 (2007). doi: https://doi.org/10.1016/j.jmmm.2006.10.304.

[51] B. Bauer et al. "The ALPS project release 2.0: open source software for strongly correlated systems". Journal of Statistical Mechanics: Theory and Experiment **2011**, P05001 (2011). doi: 10.1088/1742-5468/2011/05/p05001.

[52] R. J. Elliott, P. Pfeuty, and C. Wood. "Ising model with a transverse field". Phys. Rev. Lett. **25**, 443–446 (1970). doi: 10.1103/PhysRevLett.25.443.

[53] M. S. L. du Croo de Jongh and J. M. J. van Leeuwen. "Critical behavior of the two-dimensional ising model in a transverse field: A density-matrix renormalization calculation". Phys. Rev. B **57**, 8494–8500 (1998). doi: 10.1103/PhysRevB.57.8494.

[54] H. Rieger and N. Kawashima. "Application of a continuous time cluster algorithm to the two-dimensional random quantum ising ferromagnet". The European Physical Journal B - Condensed Matter and Complex Systems **9**, 233–236 (1999). doi: 10.1007/s100510050761.

[55] H. W. J. Blöte and Y. Deng. "Cluster monte carlo simulation of the transverse ising model". Phys. Rev. E **66**, 066110 (2002). doi: 10.1103/PhysRevE.66.066110.

[56] A. F. Albuquerque, F. Alet, C. Sire, and S. Capponi. "Quantum critical scaling of fidelity susceptibility". Phys. Rev. B **81**, 064418 (2010). doi: 10.1103/PhysRevB.81.064418.

[57] W. Marshall. "Antiferromagnetism". Proceedings of the Royal Society of London **232**, 48–68 (1955). doi: 10.1098/rspa.1955.0200.

[58] R. Boloix-Tortosa, J. J. Murillo-Fuentes, I. Santos, and F. Pérez-Cruz. "Widely linear complex-valued kernel methods for regression". IEEE Transactions on Signal Processing **65**, 5240–5248 (2017). doi: 10.1109/TSP.2017.2726991.

[59] "cqsl/learning-ground-states-with-kernel-methods". doi: 10.5281/zenodo.7738168.

[60] J. Bradbury et al. "JAX: composable transformations of Python+NumPy programs". url: https://github.com/google/jax.

[61] F. Vicentini et al. "NetKet 3: Machine Learning Toolbox for Many-Body Quantum Systems". SciPost Phys. CodebasesPage 7 (2022). doi: 10.21468/SciPostPhysCodeb.7.

[62] G. Carleo et al. "Netket: A machine learning toolkit for many-body quantum systems". SoftwareX **10**, 100311 (2019). doi: https://doi.org/10.1016/j.softx.2019.100311.

[63] D. Häfner and F. Vicentini. "mpi4jax: Zero-copy mpi communication of jax arrays". Journal of Open Source Software **6**, 3419 (2021). doi: 10.21105/joss.03419.

[64] A. W. Sandvik. "Finite-size scaling of the ground-state parameters of the two-dimensional heisenberg model". Phys. Rev. B **56**, 11678–11690 (1997). doi: 10.1103/PhysRevB.56.11678.

[65] R. Novak, L. Xiao, J. Hron, J. Lee, A. A. Alemi, J. Sohl-Dickstein, and S. S. Schoenholz. "Neural tangents: Fast and easy infinite neural networks in python" (2020) arXiv:1912.02803.

[66] C. Williams. "Computing with infinite networks". In Advances in Neural Information Processing Systems. Volume 9. MIT Press (1996). url: https://proceedings.neurips.cc/paper/1996/file/ae5e3ce40e0404a45ecacaaf05e5f735-Paper.pdf.

Appendix

We provide a proof of Theorem 1 and touch on the scaling in the case of gapless Hamiltonians in Appendix A, discuss the details of the numerical implementation of the sampling and kernel ridge regression in Appendix B and give additional numerical results in Appendix C. Finally in Appendix D we highlight the connection between the kernel used in this article and symmetrized Restricted Boltzmann Machine (RBM) in the infinite hidden neuron density limit.

A Proof of Theorem 1 (Convergence of the noisy power method)

We provide a proof of Theorem 1 presented in the main text, conceptually following the steps of the proof in the supplementary material of Ref. [38].

The main idea of the proof is to show that, if the noise is small enough, at every step the distance between the current state and the ground state decreases with respect to the previous step, where the Fubini-Study metric, given by

$$\theta(\psi,\phi) \coloneqq \arccos\sqrt{\mathcal{F}(\phi,\psi)} = \arccos\left(\frac{|\langle\phi|\psi\rangle|}{\|\psi\|\|\phi\|}\right) \tag{26}$$

is used as as the measure of distance. We remark that the the Fubini-Study metric is a monotonic function of the infidelity. To simplify notation we define the following short-hand notation for the overlap of the state and noise at step n with the eigenstates of the Hamiltoinian

$$\begin{aligned}\Psi_k^{(n)} &= \langle\Upsilon_k|\Psi^{(n)}\rangle\\ \Delta_k^{(n)} &= \langle\Upsilon_k|\Delta^{(n)}\rangle\,.\end{aligned} \tag{27}$$

We assume that $\Lambda > \frac{E_1+E_{\max}}{2}$, and therefore the two largest (also in absolute terms) eigenvalues of $\Lambda-\hat{H}$ are given by $\sigma_0 = \Lambda - E_0$ and $\sigma_1 = \Lambda - E_1$. Their difference is is equal to the gap of $\hat{H}$ since $\sigma_0 - \sigma_1 = E_1 - E_0 = \delta$.

Assuming normalized eigenstates $\|\Upsilon_k\| = 1$ we have that the cosine of the Fubini-Study distance is given by

$$\cos\theta(\Psi^{(n)},\Upsilon_0) = \sqrt{\mathcal{F}(\Psi^{(n)},\Upsilon_0)} = \frac{\left|\Psi_0^{(n)}\right|}{\left\|\Psi^{(n)}\right\|}. \tag{28}$$

Then, using basic trigonometric identities we find

$$\tan\theta(\Psi^{(n)},\Upsilon_0) = \sqrt{\frac{1-\mathcal{F}(\Psi^{(n)},\Upsilon_0)}{\mathcal{F}(\Psi^{(n)},\Upsilon_0)}} = \frac{\sqrt{\sum_{k\geq 1}\left|\Psi_k^{(n)}\right|^2}}{\left|\Psi_0^{(n)}\right|} \tag{29}$$

where we used $\sum_k \left|\Psi_k^{(n)}\right|^2 = \left\|\Psi^{(n)}\right\|^2$.

We can show that the distance decreases at every step of the noisy power method under the following assumptions on the noise at each step, in terms of the current state $\Psi^{(n)}$:

$$\begin{aligned}\left|\Delta_0^{(n)}\right| &\leq \frac{\delta}{4}\left\|\Psi^{(n)}\right\|\cos\theta(\Psi^{(n)},\Upsilon_0)\\ \left\|\Delta^{(n)}\right\| &\leq \frac{\delta}{4}\left\|\Psi^{(n)}\right\|\varepsilon\end{aligned} \tag{30}$$

where $\varepsilon < \frac{1}{2}$.

For $|\Psi^{(n+1)}\rangle = \gamma^{(n)}\left((\Lambda-\hat{H})|\Psi^{(n)}\rangle + |\Delta^{(n)}\rangle\right)$ the tangent of the distance of the propagated state with the ground state, which is a monotonic function of it, can be bounded as

$$\begin{aligned}&\tan\theta(\Psi^{(n+1)},\Upsilon_0)\\ &= \frac{\sqrt{\sum_{k\geq 1}\left|\langle\Upsilon_k|\Lambda-\hat{H}|\Psi^{(n)}\rangle + \Delta_k^{(n)}\right|^2}}{\left|\langle\Upsilon_0|\Lambda-\hat{H}|\Psi^{(n)}\rangle + \Delta_0^{(n)}\right|}\\ &\leq \frac{\sigma_1\sqrt{\sum_{k\geq 1}\left|\,|\Upsilon_k\rangle\,\Psi^{(n)}\right|} + \sqrt{\sum_{k\geq 1}\left|\Delta_k^{(n)}\right|^2}}{\sigma_0\left|\Psi_0^{(n)}\right| - \left|\Delta_0^{(n)}\right|}\\ &\leq \frac{\sigma_1\sqrt{\sum_{k\geq 1}\left|\,|\Upsilon_k\rangle\,\Psi^{(n)}\right|} + \sqrt{\sum_{k}\left|\Delta_k^{(n)}\right|^2}}{\sigma_0\left|\Psi_0^{(n)}\right| - \left|\Delta_0^{(n)}\right|}\\ &= \frac{\sigma_1\tan\theta(\Psi^{(n)},\Upsilon_0) + \frac{\left\|\Delta^{(n)}\right\|}{\left|\Psi_0^{(n)}\right|}}{\sigma_0 - \frac{\left|\Delta_0^{(n)}\right|}{\left|\Psi_0^{(n)}\right|}}\\ &= \frac{\sigma_1\tan\theta(\Psi^{(n)},\Upsilon_0) + \frac{\left\|\Delta^{(n)}\right\|}{\left\|\Psi^{(n)}\right\|\cos\theta(\Psi^{(n)},\Upsilon_0)}}{\sigma_0 - \frac{\left|\Delta_0^{(n)}\right|}{\left\|\Psi^{(n)}\right\|\cos\theta(\Psi^{(n)},\Upsilon_0)}}\end{aligned} \tag{31}$$

where in the first step we applied the triangle inequality and bounded with the largest eigenvalue in the nominator and applied the reverse triangle inequality in the denominator, assuming that $\sigma_0\left|\Psi_0^{(n)}\right| \geq \left|\Delta_0^{(n)}\right|$. Under the assumptions of Eq. (30) and bounding $\frac{1}{\cos\theta} \leq 1+\tan\theta$ [7] we have

$$\begin{aligned}&\tan\theta(\Psi^{(n+1)},\Upsilon_0)\\ &\leq \frac{\sigma_1\tan\theta(\Psi^{(n)},\Upsilon_0) + \frac{\delta}{4}\varepsilon(1+\tan\theta(\Psi^{(n)},\Upsilon_0))}{\sigma_0 - \frac{\delta}{4}}\\ &= (1-\frac{\xi}{\sigma_1+3\xi})\frac{\sigma_1+\varepsilon\xi}{\sigma_1+2\xi}\tan\theta(\Psi^{(n)},\Upsilon_0) + \frac{\xi}{\sigma_1+3\xi}\varepsilon\\ &\leq \max\left(\frac{\sigma_1+\varepsilon\xi}{\sigma_1+2\xi}\tan\theta(\Psi^{(n)},\Upsilon_0),\varepsilon\right)\end{aligned} \tag{32}$$

where we defined $\xi \coloneqq \frac{\delta}{4}$ and split $\frac{1}{\sigma_1+3\xi} = (1-\frac{\xi}{\sigma_1+3\xi})\frac{1}{\sigma_1+2\xi}$. Then, realizing that $\frac{\xi}{\sigma_1+3\xi} \in [0,1]$,

[7] This is equivalent to bounding $\frac{1}{\sqrt{\mathcal{F}}} \leq 1+\sqrt{\frac{1-\mathcal{F}}{\mathcal{F}}}$.

we used that the weighed mean of two terms is less than the maximum [8]. Splitting again $\frac{1}{\sigma_1+2\xi} = (1 - \frac{\xi}{\sigma_1+2\xi})\frac{1}{\sigma_1+\xi}$ we have

$$\begin{aligned}\frac{\sigma_1+\varepsilon\xi}{\sigma_1+2\xi} &= (1-\frac{\xi}{\sigma_1+2\xi})\frac{\sigma_1}{\sigma_1+\xi} + \frac{\xi}{\sigma_1+2\xi}\varepsilon \\ &\leq \max\left(\frac{\sigma_1}{\sigma_1+\xi}, \varepsilon\right).\end{aligned} \tag{33}$$

Since $(1+\frac{\xi}{\sigma_1})^4 \geq 1 + 4\frac{\xi}{\sigma_1}$ we can bound $\frac{\sigma_1}{\sigma_1+\xi} \leq (\frac{\sigma_1}{\sigma_1+4\xi})^{\frac{1}{4}} = (\frac{\sigma_1}{\sigma_0})^{\frac{1}{4}} = (\frac{\Lambda-E_1}{\Lambda-E_0})^{\frac{1}{4}}$, and find

$$\tan\theta(\Psi^{(n+1)}, \Upsilon_0) \leq \max\left(\varepsilon, \omega\tan\theta(\Psi^{(n)}, \Upsilon_0)\right). \tag{34}$$

where $\omega \coloneqq \max\left((\frac{\Lambda-E_1}{\Lambda-E_0})^{\frac{1}{4}}, \varepsilon\right)$. Therefore under the assumptions of Eq. (30), the noisy power method decreases $\tan\theta(\Psi^{(n)}, \Upsilon_0)$ at every step until at some step M it reaches $\tan\theta(\Psi^{(M)}, \Upsilon_0) = \varepsilon$ and then it stays at that distance.

Finally we show that the following strengthened assumptions, given in terms of the initial distance, imply the assumptions in Eq. (30), and thus the convergence of the method.

$$\begin{aligned}\left|\Delta_0^{(n)}\right| &\leq \frac{\delta}{5}\left\|\Psi^{(n)}\right\|\cos\theta(\Psi^{(0)}, \Upsilon_0) \\ \left\|\Delta^{(n)}\right\| &\leq \frac{\delta}{5}\left\|\Psi^{(n)}\right\|\varepsilon\end{aligned} \tag{35}$$

It is clear that $\|\Delta^{(n)}\| \leq \frac{\delta}{5}\|\Psi^{(n)}\|$ implies $\|\Delta^{(n)}\| \leq \frac{\delta}{4}\|\Psi^{(n)}\|$. From Eq. (34), since $\omega \leq 1$, it follows that at every step $\tan\theta(\Psi^{(n+1)}, \Upsilon_0) \leq \max\left(\varepsilon, \tan\theta(\Psi^{(n)}, \Upsilon_0)\right)$ which implies $\tan\theta(\Psi^{(n)}, \Upsilon_0) \leq \max\left(\varepsilon, \tan\theta(\Psi^{(0)}, \Upsilon_0)\right)$. Then, since $\cos\theta = \sqrt{\frac{1}{1+\tan^2\theta}} \geq 1 - \frac{\tan^2\theta}{2}$ for $\varepsilon \leq \frac{1}{2}$ this implies $\cos\theta(\Psi^{(n)}, \Upsilon_0) \geq \min\left(1 - \frac{\varepsilon^2}{2}, \cos\theta(\Psi^{(0)}, \Upsilon_0)\right) \geq \frac{7}{8}\cos\theta(\Psi^{(0)}, \Upsilon_0)$ and thus $\frac{1}{5}\cos\theta(\Psi^{(0)}, \Upsilon_0) \leq \frac{1}{4}\cos\theta(\Psi^{(n)}, \Upsilon_0)$.

Number of steps

We derive the bound on the number of steps M needed to reach a state with $\tan\theta \leq \varepsilon$. As ω in Eq. (34) is a multiplicative factor it is clear that a logarithmic number of steps M suffices to reach ε. In order to bound the number of steps it is convenient to first bound $\ln(\frac{1}{\omega})$. Using the definition of ω we have that

$$\ln\left(\frac{1}{\omega}\right) = \min\left(\ln\left(\frac{1}{\varepsilon}\right), \frac{1}{4}\ln\left(\frac{1}{\frac{\Lambda-E_1}{\Lambda-E_0}}\right)\right) \tag{36}$$

[8] Let $\alpha \in [0,1]$, $x, y \in \mathbb{R}$, then $\alpha x + (1-\alpha)y \leq \max(x, y)$

Assuming $\varepsilon < \frac{1}{2}$, we bound $\ln(\frac{1}{\varepsilon}) \geq \ln(2)$. For the other term we use $\ln\left(\frac{1}{\frac{\Lambda-E_1}{\Lambda-E_0}}\right) \geq 1 - \frac{\Lambda-E_1}{\Lambda-E_0}$ and thus

$$\ln\left(\frac{1}{\omega}\right) \geq \min\left(\ln(2), \frac{1-\frac{\Lambda-E_1}{\Lambda-E_0}}{4}\right) = \frac{1-\frac{\Lambda-E_1}{\Lambda-E_0}}{4}. \tag{37}$$

Then, recursively applying Eq. (34) until step M where we assume to reach ε we set

$$\tan\theta(\Psi^{(M)}, \Upsilon_0) = \omega^M \tan\theta(\Psi^{(0)}, \Upsilon_0) \stackrel{!}{=} \varepsilon. \tag{38}$$

Taking the log on both sides, solving for M and using Eq. (37) we find

$$M \leq \frac{4}{1-\frac{\Lambda-E_1}{\Lambda-E_0}}\ln\left(\frac{\tan\theta(\Psi^{(0)}, \Upsilon_0)}{\varepsilon}\right). \tag{39}$$

Scaling for gapless hamiltonians

We briefly analyze the scaling of the SLPM in the case of a gapless Hamiltonian with a gap which closes polynomially with the inverse system size.

In Theorem 1 it is easy to see that the number of steps needed is inversely proportional to the gap $\delta = E_1 - E_0$, since $(1 - \frac{\Lambda-E_1}{\Lambda-E_0})^{-1} = \frac{\Lambda-E_0}{\delta}$. If we assume for a system of size L that, (I) the gap closes as $\delta \propto L^{-\beta}$, (II) we can choose a constant $|\Lambda| \ll E_0$ and (III) the ground state energy scales as $|E_0| \propto L$, we have that $\frac{\Lambda-E_1}{\Lambda-E_0} = \frac{\delta}{\Lambda-E_0} \propto L^{-(\beta+1)}$ and thus a number of steps proportional to $L^{\beta+1}$ are needed. We note that the same applies to the power method in absence of noise.

In the case of noise however, we also need a step infidelity which is small enough so that our method can resolve the gap (see Corollary 1). Under the assumptions above this would result in a required step infidelity $\propto L^{-2(\beta+1)}$, meaning that we would need in the order of $L^{\frac{2(\beta+1)}{\alpha}}$ samples. Therefore, as long as the gap closes polynomially with the inverse system size, the method is efficient.

B Implementation Details

Details of the Monte-Carlo sampling procedure

We use Markov-Chain Monte Carlo sampling with the Metropolis-Hastings algorithm (see e.g. Ref. [8] Sec. 3.9 and references therein) to generate samples $x \sim |\phi|^2$, where ϕ is given in terms of log-amplitudes by the kernel ridge regression predictor

$$\log\phi(x) = \sum_i w_i\, k(x, x_i). \tag{40}$$

For the TFI model we perform single-spin flip updates, and for the AFH model we propose to exchange two spins at each step, initializing the markov chains with states which have total spin 0.

system size	method	$h = 1.0$	$h = 2.0$	$h = 3.0$	$h = 4.0$
4×4	ED	-34.010598	-40.190194	-51.448129	-66.223620
6×6	ED	-76.523833	-90.407093	-115.23271	-148.83265
8×8	QMC	$-136.043(2)$	$-160.722(2)$	$-204.632(3)$	$-264.569(3)$
10×10	QMC	$-212.567(2)$	$-251.132(2)$	$-319.615(3)$	$-413.379(4)$

Table 1: Reference ground state energies of the the transverse-field Ising model in two dimensions with periodic boundary conditions for several values of h. ED computed with SpinED [48, 49], and QMC with Alps [50, 51].

Multiple occurences of the same sample

Monte-Carlo sampling can produce the same sample multiple times. Formally, including multiple occurrences of the same samples in the data-set reduces the rank of the kernel matrix, which can lead to numerical instabilities when decomposing the matrix, and needs to be accounted for with the regularization. We note that when using a kernel which is symmetric with respect to a symmetry group G, then two samples x, x' belonging to the same orbit, meaning that $gx = x'$ for some $g \in G$, have the same effect, and we consider them to be the "same" as well.

By introducing a weight factor c_i into the loss, we can account for the number of occurences in the loss, and keep only a unique set of samples. This reduces the cost, as the kernel matrix is smaller since there are fewer samples, while formally keeping the loss invariant, and results in a de-facto sample-dependent regularization as we see in the following. The modified loss is given by

$$\mathcal{L} = \underset{w}{\operatorname{argmin}} \sum_i c_i |f(w; x_i) - y_i|^2 + \lambda \|f\|^2, \qquad (41)$$

and the optimal weights are given by the solution of

$$(k(x_i, x_j) + \lambda \frac{\delta_{i,j}}{c_i}) w_j = y_i \qquad (42)$$

This is very similar to the original system of equations (Eq. (19) in the main text), except that the regularization is now sample-dependent, except in the limit $\lambda \to 0$, where removing repeated samples from the data-set has no effect. We found that in practice, when the regularization λ is already very small, it is sufficient to remove repetitions, while not using the sample-dependent regularization, and therefore adopt this procedure for the simulations in this paper.

Numerical Details of the supervised learning procedure

The self-learning power method is initialized with a data-set for the uniform superposition state, taking uniform samples from the whole Hilbert space (taking only states with magnetization 0 for the AFH) and setting all log-amplitudes for the labels to $y = 0$, resulting in a state $\Psi^0(x) = 1$. For the kernel Eq. (23) we use the non-linearity $\sigma(x) = x \arcsin(\gamma x)$ fixing $\gamma = 0.5808$. Throughout our experiments we keep the regularization fixed at $\lambda = 10^{-8}$, except in rare cases where we get nan's and have to increase it to 10^{-7}.

We solve the linear system of equations for the weights (Eq. (19)) using the Cholesky decomposition of the regularized kernel matrix. In general we work with un-normalized states. As the operator $\lambda - \hat{H}$ is not unitary, to avoid underflow/overflow and ensure numerical stability, at every step we subtract the largest log-amplitude present in the data-set from all the labels, effusively normalizing the state so that $\max_{x_i} \left|\Psi^{(n)}(x_i)\right| = 1$ for all samples x_i in the data-set. We wrote the code for the numerical simulations with the jax library [60], using the sampler from netket [61, 62] and we optionally parallelized it using mpi4jax [63]. All of the simulations in this article were run serially on a NVIDIA V100 gpu.

C Additional Numerical Experiments

system size	ED	QMC	Ref. [64]
20	-35.617546	n/a	n/a
40	-70.986091	n/a	n/a
80	n/a	$-141.848(2)$	n/a
4×4	-44.913933	n/a	n/a
6×6	-97.757590	n/a	n/a
8×8	n/a	$-172.414(2)$	$-172.413(2)$
10×10	n/a	$-268.623(3)$	$-268.620(2)$

Table 2: Reference ground state energies of the the antiferromagnetic Heisenberg model on one and two-dimensional periodic lattices. ED computed with SpinED [48, 49], and QMC with Alps [50, 51]. For 2D we also report SSE results from Ref. [64] for comparison.

For completeness, in this appendix we provide the reference energies used for benchmarking purposes and and present several additional numerical experiments that we performed to further support the results in the main text.

In order to benchmark our method we computed reference energies with exact diagonalization (ED) using using the SpinED package [48, 49], and did Quantum Monte Carlo (QMC) simultations using the loop algorithm from the Alps package [50, 51]. For the

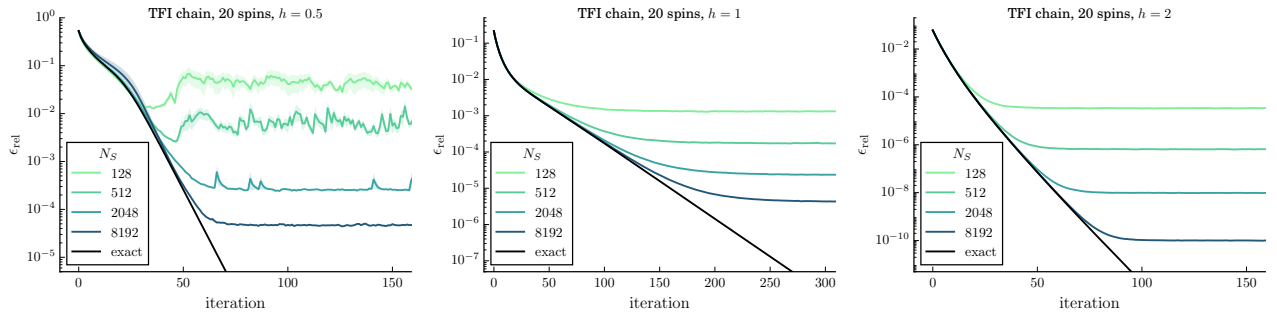

Figure 5: Convergence of the Self-learning power method for the TFI model on a one-dimensional chain of $N = 20$ spins. Relative error of the predicted energy with the true ground state energy as a function of the number of iterations n, compared to the power method for $h = 0.5$ (left panel), $h = 1$ (central panel) and $h = 2$ (right panel), taking the average over 100 runs. The central panel is equal to the left panel of Fig. 2.

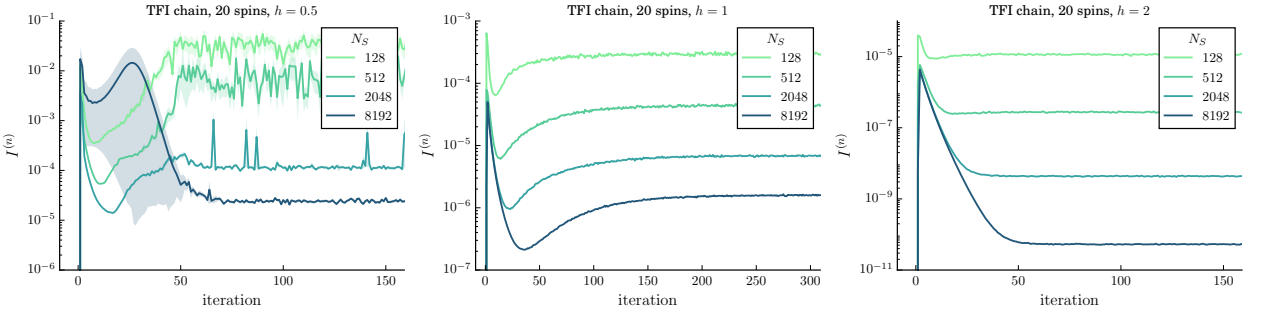

Figure 6: Step infidelity of learning the states visited along the SLPM for the TFI model on a one-dimensional chain of $N = 20$ spins. (left panel): $h = 0.5$, (central panel): $h = 1$, (right panel): $h = 2$. Shown are averages over 100 runs.

QMC simulations we fixed the inverse temperature at $\beta = 1000$ and ran 10^5 thermalization steps followed by 10^6 sweeps. In Table 1 we provide energies for the TFI model in two dimensions and in Table 2 for the AFH model in one and two dimensions.

TFI model in one dimension at fixed system size

We start with a few additional results for the 20-spin Ising chain. In Fig. 5 we provide a plot for the relative error as a function of the number of iterations, for $h = 0.5$ and $h = 2$ in addition to $h = 1$ as already plotted in the left panel of Fig. 2 in the main text. We observe comparable behaviour in all three regimes, except for $h = 0.5$ when the number of samples is low and fluctuations occur. This can be explained by the large noise introduced in this case as can be seen by our study of the step infidelity in the following plot. In Fig. 6 we plot step infidelity $I^{(n)}$ as a function of the step n. We observe that it is not constant for all the steps of the self-learning power method, but varies before finally leveling off when a steady state is reached, as can be seen by comparing to Fig. 5. In the case of $h = 0.5$ and a number of samples $N_S \leq 512$ the step infidelity becomes higher over time, causing the error to increase. We would like to point out that in this case the condition on the step infidelity mentioned in the discussion of Eq. (11) ($\varepsilon < 1/2$) is not satisfied, and therefore Theorem 1 cannot be applied. Nevertheless, by increasing the number of samples we can alleviate these fluctuations.

As pointed out in the main text, it is possible to generalize the theoretical bounds to varying noise (quantified by ε in Theorem 1). One straightforward way to do so is to apply the theorem several times in a row. Starting from an initial state with possibly exponentially small overlap with the ground state, we fix $\varepsilon = \frac{1}{2}$ and run enough steps until a infidelity of $\varepsilon^2 = 1/4$ is reached. In this regime the first assumption of Theorem 1 (Eq. (9)) dominates, as the initial fidelity is much smaller than ε. Now we are in a state with a finite fidelity of 3/4, which we use as initial state to apply Theorem 1 again, choosing a smaller value of ε, e.g. as a function of the step infidelity according to Eq. (11). Now the dominant assumption of Theorem 1 is the second one (Eq. (10)) as the initial fidelity is much larger than ε. We note that this argument also justifies starting the SLPM with a lower number of samples until a steady state is reached, and increasing afterwards, resulting in a lower compuational cost.

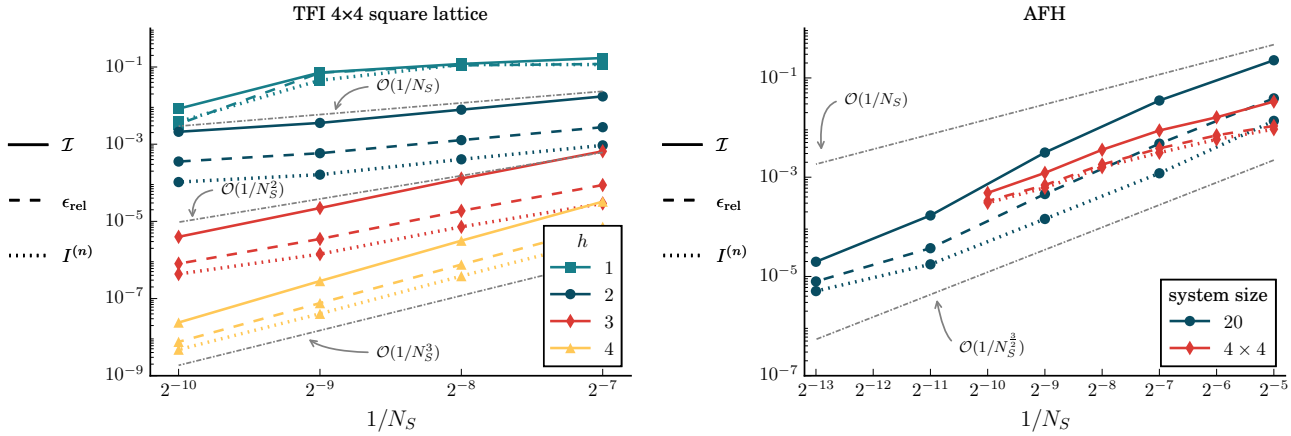

Figure 7: Final state convergence for the TFI model in 2D (left panel) and for the AFH model in 1D and 2D (right panel), in analogy to the right panel of Fig. 2 in the main text (TFI in 1D). Plotted are $I^{(n)}$: step infidelity of learning the final state (see Definition 1), $\mathcal{I}$: infidelity of the final state with the true ground state (defined in Eq. (12)), and $\epsilon_{\rm rel}$: relative error of the predicted energy of the final state (defined in Eq. (13)) after convergence of the self-learning power method, as a function of the number of samples in the data-set N_S, taking averages over 100 runs.

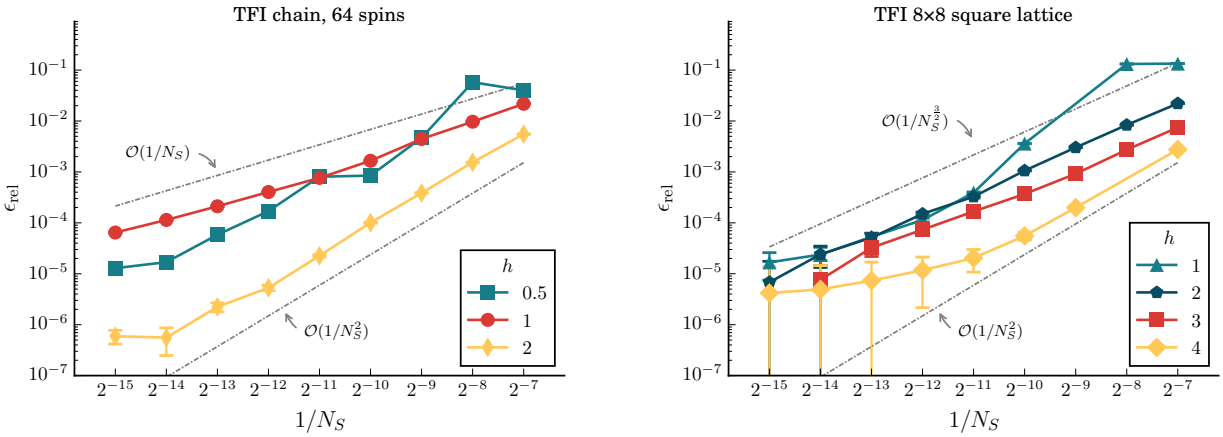

Figure 8: Scaling of the SLPM ground-state energy relative error as a function of the number of samples in the data-set size for a 1D chain of 64 spins **(left panel)** and 2D 8×8 **(right panel)** periodic lattice of the TFI Hamiltonian. Estimates and reference values computed as in Fig. 3.

Numerical verification of the efficient-learning assumption

In the main text we numerically investigated the efficient-learning assumption for the TFI model on a 1D chain with 20 spins, showing that the step infidelity after a fixed number of steps is compatible with a power-law $I^{(n)} \propto N_S^{-\alpha}$. In Fig. 7 we provide additional evidence that this is also verified for the TFI model in two dimensions and for the AFH model, plotting $I^{(n)}$ as a function of the number of samples after $n = 200$ steps of the SLPM. In the left panel we consider a 4×4 square lattice of the TFI model for different values of h, and in the right panel a 20 spin chain for the AFH in 1D, and a 4×4 square lattice in 2D. Furthermore we observe that the final Infidelity $\mathcal{I}$ and relative energy error $\epsilon_{\rm rel}$ follow similar power laws with the same exponent as $I^{(n)}$ confirming that Corollary 1 is valid for the these systems.

TFI and AFH model in one and two dimensions

In the main text we studied the scaling of the SLPM ground-state energy for the TFI model with the system size for a fixed number of samples, and the scaling with the number of samples for the AFH model. In Figs. 8 and 9 we provide the respective other plot for the two models. In Fig. 8 we study the TFI for 64 spins, in a 1-dimensional chain in the left panel and on a 8×8 square lattice on the right panel. In both cases we find a power law-like scaling of the error with the number of samples, further corroborating our results. In Fig. 9 we investigate the scaling

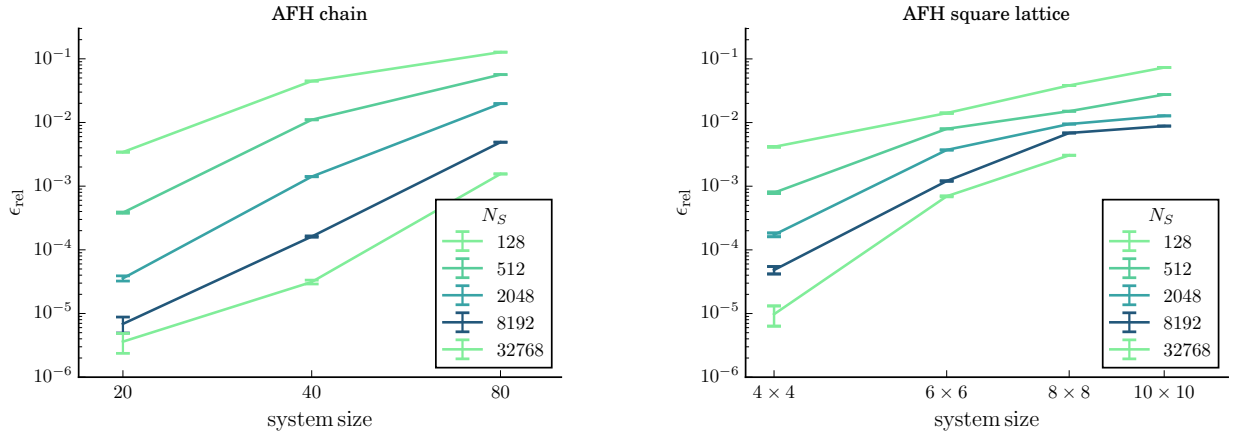

Figure 9: Scaling of the SLPM ground-state energy relative error as a function of the system size for 1D **(left panel)** and 2D **(right panel)** periodic lattices of the AFH Hamiltonian for different data-set sizes N_S. Estimates and reference values computed as in Fig. 3.

of the SLPM error for the AFH in the system size. For the one-dimensional systems in the left panel we find scaling compatible with a power law, similar to what we found for the TFI in the main text. For the two-dimensional systems we observe that for the largest system and higher number if samples levels off. It might be possible to attribute this to finite-size effects on the smallest system, given that the number of samples becomes of the order of the effective Hilbert space size.

D The Kernel of a symmetrized Restricted Boltzmann machine

Neural networks, like simple restricted Boltzmann machines (RBM) have been shown to be able to learn the ground states of the systems studied in this article, with an accuracy which increases with the width of the hidden layer, given by the hidden-layer density α [23]. Recently it has been discovered that the training of neural networks is governed by a kernel, the neural tangent kernel (NTK)[47]. It has been shown that in the infinite hidden-layer width limit, on average the prediction of randomly initialized neural networks, which are fully trained with gradiend descent using the least-squares loss, is equal to the prediction of kernel ridge regression using the NTK. Therefore, this theory formally allows us to study RBM's in the $\alpha \to \infty$ limit using kernel methods.

In particular we are interested in the NTK of a symmetrized RBM, and the effect the symmetrization of the neural network has on the symmetry properties of the kernel. Let G be a permutation group. A symmetrized RBM consists of the following layers

(a) An initial dense symmetric layer projecting onto G

(b) element-wise $\log\cosh$ Nonlinearity

(c) Sum over the features of (a)

(d) Global AvgPool over the elements of G

which produce an invariant model. We can write it as

$$f(\mathbf{x}) = \underbrace{\frac{1}{|G|}\sum_{g\in G}}_{(d)} \underbrace{\frac{1}{\sqrt{n_1}}\sum_{i}^{n_1}}_{(c)} \underbrace{\sigma}_{(b)} \Big(\underbrace{\frac{1}{\sqrt{n_0}}\sum_{k=1}^{n_0}(L_g w_i)_k \, \mathbf{x}_k}_{(a)} \Big) \tag{43}$$

where $\sigma(\cdot) = \log(\cosh(\cdot))$, $w \in \mathbb{R}^{n_1 \times n_0}$, $w_{i,j} \sim \mathcal{N}(0,1)$ is the matrix containing the filters, $\mathbf{x} \in \mathcal{X} = \{-1, 1\}^{n_0}$ is the input of size n_0 and n_1 is the number of features in the dense symmetric layer of which we take the limit $n_1 \to \infty$. L_g denotes a concrete instance of an operator performing the symmetry operation g on an element of $\mathcal{X}$, and we are not adding any hidden or visible bias.

The NTK of a neural network $f_\theta(\mathbf{x})$ is defined as

$$\Theta(\mathbf{x}, \mathbf{y}) \coloneqq \sum_{p=1}^{P} \partial_{\theta_p} f_\theta(\mathbf{x}) \, \partial_{\theta_p} f_\theta(\mathbf{y}) \tag{44}$$

where θ_p are the parameters of the neural network, in our case given by w and $P \to \infty$ when $n_1 \to \infty$.

It can be shown that the NTK of the symmetrized RBM defined above in Eq. (43) is given by

$$\Theta(\mathbf{x}, \mathbf{y}) \frac{1}{|G|} \sum_{g\in G} \sigma\left(\frac{1}{n_0} (L_g \mathbf{x})^T \mathbf{y} \right) \tag{45}$$

where $\sigma(x) \coloneqq x \arcsin(\gamma x)$ for $\gamma \approx 0.5808$ which comes from approximating the nonlinearity. We note that in the main text, when presenting this kernel, under a slight abuse of notation we wrote g instead of L_g to simplify the expression.

In the remainder we provide a sketch of the derivation needed to arrive at this simplified expression for the kernel. We note that the NTK can also be computed with a library such as Ref. [65], using circular convolution to get translation symmetry (the full space group is not possible), and with increased computational cost with respect to the kernel presented above, as convolution in twice the number of dimensions as the input needs to be performed.

Derivation of the neural tangent kernel of a symmetrized restricted Boltzmann machine with ± 1 input values

The common way to determine the NTK Θ of a neural network is to use a recursive procedure computing it layer by layer. To compute Θ one also needs to compute the covariance Σ of the centered multivariate normal distribution describing the output of the neural network over random initialization of the weights, before training. For a simple feed-forward neural network it has been shown that both Θ and Σ are represented by a scalar kernels times an implicit identity $\otimes I_{n_l}$, as the features of the output of each layer are independent. For the symmetrized layer of the RBM however, the kernels are no longer scalar, as the same features for different elements of G are correlated. Therefore, we have to work with kernels $\Sigma_{g,g'}, \Theta g, g'$ which are of size $|G| \times |G|$ with $g, g' \in G$. The final scalar NTK is obtained only after the last pooling layer. We note that it is possible to show, for both the covariance and the NTK for a permutation group G, that $\Sigma_{g,g'} = \Sigma_{e,g'g^{-1}}$ and $\Theta_{g,g'} = \Theta_{e,g'g^{-1}}$ holds[9], where e is the identity element of G. Therefore instead of computing the whole kernel matrix it suffices that we work with a single row.

We apply the recursive procedure to compute the NTK of the symmetrized RBM.

For the first layer (a) we have

$$\begin{aligned} \Sigma^1_{e,g'}(\mathbf{x},\mathbf{y}) &= \frac{1}{n_0}(L_{g'}\mathbf{x})^T\,\mathbf{y} \\ \Theta^1_{g,g'}(\mathbf{x},\mathbf{y}) &= \Sigma^1_{g,g'}(\mathbf{x},\mathbf{y}) = \Sigma^1_{e,g'g^{-1}}(\mathbf{x},\mathbf{y}) \end{aligned} \tag{46}$$

where we indicate the layer with the superscript. After applying the nonlinearity (b) and summing all features (c) we have that the NTK after the second layer is given by

$$\Theta^2_{g,g'}(\mathbf{x},\mathbf{y}) = \dot{\Sigma}^1(\mathbf{x},\mathbf{y})\,\Theta^1_{g,g'}(\mathbf{x},\mathbf{y}) \tag{47}$$

where

$$\dot{\Sigma}^1(\mathbf{x},\mathbf{y}) = \mathbb{E}_{f\sim\mathcal{N}(0,\Sigma^1)}[\dot{\phi}(f(\mathbf{x}))\dot{\phi}(f(\mathbf{y}))] \tag{48}$$

and $\dot{\phi}(\cdot) = \tanh(\cdot)$ is the derivative of $\log\cosh$.

[9] To give a concrete example, in the case of circular translation with stride 1 this means that the kernel matrices are circulant.

We apply the pooling (d) to get the scalar NTK for the output of the symmetrized RBM, pooling over the whole group G:

$$\begin{aligned} \Theta^3(\mathbf{x},\mathbf{y}) &= \frac{1}{|G|^2}\sum_{k\in gG}\sum_{k'\in g'G}\Theta^2_{k,k'}(\mathbf{x},\mathbf{y}) \\ &= \frac{1}{|G|^2}\sum_{k\in G}\sum_{k'\in G}\Theta^2_{k,k'}(\mathbf{x},\mathbf{y}) \end{aligned} \tag{49}$$

where we chose $g, g' \in G$ arbitrarly, as $\forall g \in G : gG = G$.

We combine the individual components to find the equation for the NTK of the symmetrized RBM, and simplify to obtain the NTK of the output of the final layer:

$$\begin{aligned} \Theta^3(\mathbf{x},\mathbf{y}) &= \frac{1}{|G|^2}\sum_{k\in G}\sum_{k'\in G}\dot{\Sigma}^1(\mathbf{x},\mathbf{y})\,\Sigma^1_{e,k'k^{-1}}(\mathbf{x},\mathbf{y}) \\ &= \frac{1}{|G|}\sum_{k\in G}\dot{\Sigma}^1(\mathbf{x},\mathbf{y})\,\Sigma^1_{e,k}(\mathbf{x},\mathbf{y}) \end{aligned} \tag{50}$$

What is left is to find an expression for $\dot{\Sigma}^1(\mathbf{x},\mathbf{y}) = \mathbb{E}_{f\sim\mathcal{N}(0,\Sigma^1)}[\dot{\phi}(f(\mathbf{x}))\dot{\phi}(f(\mathbf{y}))$, which requires evaluating a two-dimensional gaussian integral. We are not aware of any analytical solutions for our choice of non-linearity, and opt to approximate the non-linearity with one for which analytical solutions are known. We can approximate

$$\frac{d}{dx}\log\cosh(x) = \tanh(x) \approx \operatorname{erf}(\alpha x) \tag{51}$$

where $\alpha \approx 0.8324$ (which can be found numerically by minimizing the L2 error in a suitable interval centered around 0). Then noticing that for all inputs $\mathbf{x} \in \{-1,1\}^{n_0}$ it holds that $\Sigma^1_{g,g}(\mathbf{x},\mathbf{x}) = \frac{\|\mathbf{x}\|^2}{n_0} \equiv 1$, and using Eqs. 10 and 11 of Ref. [66] we find

$$\dot{\Sigma}^1(\mathbf{x},\mathbf{y}) = \frac{2}{\pi}\arcsin\Big(\underbrace{\frac{2\alpha^2}{1+2\alpha^2}}_{\approx 0.5808 =: \gamma}\Sigma^1_{g,g'}(\mathbf{x},\mathbf{y})\Big) \tag{52}$$

Finally we obtain the expression for the kernel presented in the main text (Section 4.3)

$$\begin{aligned} \Theta^3(\mathbf{x},\mathbf{y}) &= \frac{1}{|G|}\sum_{k\in G}\dot{\Sigma}^1(\mathbf{x},\mathbf{y})\,\Sigma^1_{e,k}(\mathbf{x},\mathbf{y}) \\ &\approx \frac{2}{\pi}\frac{1}{|G|}\sum_{k\in G}\arcsin\big(\gamma\Sigma^1_{e,k}\big)\Sigma^1_{e,k} \\ &\propto \frac{1}{|G|}\sum_{k\in G}\sigma(\Sigma^1_{e,k}) = \frac{1}{|G|}\sum_{g\in G}\sigma\left(\frac{1}{n_0}(L_g\mathbf{x})^T\,\mathbf{y}\right) \end{aligned} \tag{53}$$

where we substituted $\sigma(x) \coloneqq x\arcsin(\gamma x)$.