

# Physical implementation of quantum nonparametric learning with trapped ions

Dan-Bo Zhang,<sup>1</sup> Shi-Liang Zhu,<sup>1,2,\*</sup> and Z. D. Wang<sup>3,1,†</sup>

<sup>1</sup>*Guangdong Provincial Key Laboratory of Quantum Engineering and Quantum Materials,  
GPETR Center for Quantum Precision Measurement, SPTE,  
South China Normal University, Guangzhou 510006, China*

<sup>2</sup>*National Laboratory of Solid State Microstructures,  
School of Physics, Nanjing University, Nanjing 210093, China*

<sup>3</sup>*Department of Physics and Center of Theoretical and Computational Physics,  
The University of Hong Kong, Pokfulam Road, Hong Kong, China*

Nonparametric learning is able to make reliable predictions by extracting information from similarities between a new set of input data and all samples. Here we point out a quantum paradigm of nonparametric learning which offers an exponential speedup over the sample size. By encoding data into quantum feature space, similarity between the data is defined as an inner product of quantum states. A quantum training state is introduced to superpose all data of samples, encoding relevant information for learning its bipartite entanglement spectrum. We demonstrate that a trained state for prediction can be obtained by entanglement spectrum transformation, using quantum matrix toolbox. We further workout a feasible scheme to implement the quantum nonparametric learning with trapped ions, and demonstrate the power of quantum superposition for machine learning with a state-of-the-art technology.

*Introduction.*— Machine learning extracts useful information from data for prediction. The extraction can be categorized into parametric and nonparametric learning [1, 2]. Parametric learning distills knowledge of data into parameters of a function, e.g., neural networks. However, the form of function may set a model bias or a limitation. Without a predetermined form of a function, nonparametric learning can make predictions by extracting information of similarities between new data and all samples, with the appropriate weighting of each sample being related to correlation of samples. This can utilize a self-defined kernel that may better capture the similarity between data, while on the other hand, it requires a large number of samples and the runtime is polynomial with the sample size, which is time-consuming for big data.

In quantum setting, machine learning can be enhanced with quantum information processing [3–13]. For nonparametric learning, quantum properties can be exploited in several aspects. First, encoding classical data  $\mathbf{x}$  into quantum state  $|\psi_{\mathbf{x}}\rangle$  can take advantages of quantum-enhanced feature spaces for highly nonlinear feature map [12–14], which is desirable for complicated machine learning tasks. The similarity between two data then is an inner product of two corresponding quantum states, which is classically intractable when using an instantaneous quantum polynomial [15] or continuous variable instantaneous quantum polynomial circuit [16]. Second, all data of samples can be superposed, and querying of similarities can be achieved in a quantum parallel way. Moreover, correlations of data can be extracted and transformed more efficiently with quantum matrix toolbox [5, 7, 17], including density matrix exponentiation and matrix inversion.

In this Letter, we illustrate a quantum paradigm for nonparametric learning by elaborating on a regression

task and its physical implementation. With a superposition of all samples into a quantum training state  $|\psi_{\mathbf{A}}\rangle$  defined later, we show that relevant important information for learning is represented by the bipartite entanglement spectrum of  $|\psi_{\mathbf{A}}\rangle$ , and different kinds of regression can be proposed by choosing different types of entanglement spectrum transformation. The transformation involves quantum algorithm for matrix inversion using auxiliary qumodes (continuous variables). We further propose a feasible scheme to implement this quantum nonparametric learning with trapped ions, which has well controllable internal states and motional states [18–20]. With a state-of-the-art technology [21–24], the power of quantum superposition for machine learning can be demonstrated. Our work provides a new insight for machine learning by exploiting entanglement structure of quantum superposed training data, which may inspire other new machine learning algorithms.

*Nonparametric regression.*— Let us first introduce nonparametric learning. Given a training dataset of  $M$  points  $\{\mathbf{x}^{(m)}, y^{(m)}\}$  (with  $m = 1, 2, \dots, M$ ), where  $\mathbf{x}^{(m)} \in R^N$  is a vector of  $N$  features and  $y^{(m)} \in R$  is the target value, the goal is to learn an input-output function, which can be used to predict  $\tilde{y}$  for new data  $\tilde{\mathbf{x}}$ . A parametric regression is to find a function  $f(\mathbf{x})$ , e.g., a linear model,  $f(\mathbf{x}) = \mathbf{w}^T \mathbf{x}$ , parametrized by a matrix  $\mathbf{w}$ . A nonparametric learning, instead, directly establishes a prediction based on a weighted average over the similarity between new data  $\tilde{\mathbf{x}}$  and each training data, namely,

$$\tilde{y} = \sum_{m=1}^M \alpha_m \kappa(\mathbf{x}^{(m)}, \tilde{\mathbf{x}}), \quad (1)$$

where  $\kappa(\mathbf{x}^{(m)}, \tilde{\mathbf{x}})$  defines the similarity between data and can be chosen beforehand. The weighting  $\alpha =$

$(\alpha_1, \dots, \alpha_M)^T$ , for instance, can be determined by minimizing the least-square loss function

$$L(\boldsymbol{\alpha}) = \sum_{m=1}^M (\tilde{y}^{(m)} - y^{(m)})^2 + \chi \sum_{m=1}^M \alpha_m^2. \quad (2)$$

Here the  $\chi$  term is a  $L_2$  regularization term and it makes a constraint on the weighting of each sample, which is necessary for avoiding over fitting. The combination of Eq. (1) and Eq. (2) is a kernel ridge regression. The solution turns to be  $\boldsymbol{\alpha} = (K + \chi I)^{-1} \mathbf{y}$ , where  $K$  is the covariance matrix with elements  $K_{m_1, m_2} = \kappa(\mathbf{x}^{(m_1)}, \mathbf{x}^{(m_2)})$ , and  $\mathbf{y} = (y^{(1)}, \dots, y^{(M)})^T$ . The prediction can be written as  $\tilde{y} = \mathbf{y}^T (K + \chi I)^{-1} \boldsymbol{\kappa}$ , where  $\kappa_m = \kappa(\mathbf{x}^{(m)}, \tilde{\mathbf{x}})$ .

Nonparametric regression on a quantum computer can be reformulated to exploit quantum properties. First, classical data  $\mathbf{x}$  is encoded into a quantum state  $|\psi_{\mathbf{x}}\rangle$ , which exploits the representation power of feature Hilbert space with highly nonlinear feature map [12–14]. The similarity between two data is defined as  $K_{m_1, m_2} = \langle \psi_{\mathbf{x}^{(m_1)}} | \psi_{\mathbf{x}^{(m_2)}} \rangle$ . Second, the prediction can be conducted in a quantum parallel manner. To illustrate this idea, we take a superposition of the training dataset  $\{\mathbf{x}^{(m)}\} \rightarrow M^{-\frac{1}{2}} \sum_m |m\rangle |\psi_{\mathbf{x}^{(m)}}\rangle \equiv |\psi_{\mathbf{A}}\rangle$ ,  $\{y^{(m)}\} \rightarrow |\mathbf{y}\rangle^{-\frac{1}{2}} \sum_m y^{(m)} |m\rangle \equiv |\mathbf{y}\rangle$ . The prediction is done by evaluating an overlapping between two states [25, 26]: the query state for a set of new data,  $\mathbf{y} \otimes \tilde{\mathbf{x}} \rightarrow |\psi_R\rangle \equiv |\mathbf{y}\rangle |\psi_{\tilde{\mathbf{x}}}\rangle$ , and a trained state  $|\psi_{\mathbf{A}+}\rangle$  that evolves from  $|\psi_{\mathbf{A}}\rangle$ , i.e.,

$$\tilde{y} = \langle \phi_R | \psi_{\mathbf{A}+} \rangle. \quad (3)$$

A formal derivation is shown in Supplemental Material [27]. The above equation represents a quantum version of nonparametric learning. The prediction is made by extracting information from a superposition of training data. Therein, learning is manifested in a proper trained state  $|\psi_{\mathbf{A}+}\rangle$ . A naive choice of  $|\psi_{\mathbf{A}+}\rangle = |\psi_{\mathbf{A}}\rangle$  means all training data has equal weighting, which fails to take correlations between training data into consideration. A wisdom from quantum information is to investigate the entanglement structure of the biparticle state  $|\psi_{\mathbf{A}}\rangle$ . Correlations between data reflect in a Schmidt decomposition of the training state,  $|\psi_{\mathbf{A}}\rangle = \sum_i \lambda_i |u_i\rangle |v_i\rangle$ . A larger entanglement spectrum  $\lambda_i$  corresponds to a more redundancy of information and thus should be assigned a smaller weight for the prediction. In general, we can expect that the trained state takes the formula,  $|\psi_{\mathbf{A}+}\rangle = c \sum_i g(\lambda_i) |u_i\rangle |v_i\rangle$ , where  $g(\lambda_i) \leq g(\lambda_j)$  for  $\lambda_i > \lambda_j$ . For a least square loss in Eq. (2), we can take  $g(\lambda) = \frac{\lambda}{\lambda^2 + \chi}$  [26]. Other choices can also be adopted. For instance,  $g(\lambda) = 1/\lambda$  for  $\lambda > \lambda_c$  while components of  $\lambda < \lambda_c$  are discarded, corresponding to principal component regression [2].

*Matrix inversion.*— An efficient quantum algorithm can be developed to obtain  $|\psi_{\mathbf{A}+}\rangle$  from  $|\psi_{\mathbf{A}}\rangle$ . Note that the covariance matrix can be evaluated as  $\rho_K =$

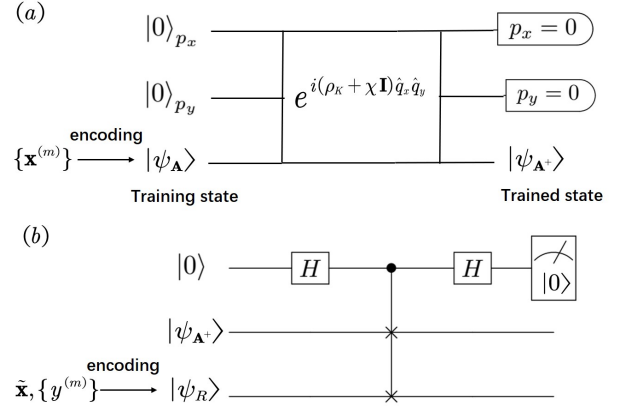


FIG. 1. Illustration of the quantum algorithm. (a). Matrix inversion algorithm for a matrix  $B = \rho_K + \chi \mathbf{I}$  that transform  $|\psi_{\mathbf{A}}\rangle$  into  $|\psi_{\mathbf{A}+}\rangle = B^{-1} |\psi_{\mathbf{A}}\rangle$ , using two auxiliary qumodes that are post-selected into zero momentum. (b). A swap test that evaluates the inner product between  $|\mathbf{A}^+\rangle$  and  $|\psi_R\rangle$ , which can be used to infer the prediction for input data  $\tilde{\mathbf{a}}$ .

$K/\text{Tr}K = \text{Tr}_1 |\psi_{\mathbf{A}}\rangle \langle \psi_{\mathbf{A}}|$  (partial trace of the addressing registers  $|m\rangle$ ) and  $\rho_K |\psi_{\mathbf{A}}\rangle = \lambda_i^2 |\psi_{\mathbf{A}}\rangle$ . Then, the required operator for evolution is given by

$$|\psi_{\mathbf{A}+}\rangle = \mathbf{B}^{-1} |\psi_{\mathbf{A}}\rangle. \quad (4)$$

where  $\mathbf{B} = \rho_K + \chi \mathbf{I}$ .

The non-unitary operator  $\mathbf{B}^{-1}$  is a matrix inversion and its quantum algorithm can exhibit exponentially speed-up. We take an approach for the matrix inversion of  $\mathbf{B}$  by writing it into a combination of unitary operators [28, 29]. Inspired by  $b^{-1} = \int_{-\infty}^{\infty} dx \delta(bx) = \int_{-\infty}^{\infty} dx dy \exp(ibxy)$ , we consider  $\mathbf{B}|b\rangle = b|b\rangle$ , then we have

$$\begin{aligned} \mathbf{B}^{-1} &= \int_{-\infty}^{\infty} dq_x dq_y \exp(i\mathbf{B} q_x q_y) \\ &\propto \langle 0_{p_x} | \langle 0_{p_y} | \exp(i\mathbf{B} \hat{q}_x \hat{q}_y) | 0_{p_x} \rangle | 0_{p_y} \rangle, \end{aligned} \quad (5)$$

where  $|0\rangle_p$  is zero momentum eigenstate. It can be seen that  $\mathbf{B}^{-1}$  can be written as an average of unitary operator  $\exp(i\mathbf{B} \hat{q}_x \hat{q}_y)$  over the infinite squeezing state  $|0_{p_x}\rangle |0_{p_y}\rangle$  of momentums  $p_x$  and  $p_y$ .

The state transformation  $|\psi_{\mathbf{A}}\rangle \rightarrow |\psi_{\mathbf{A}+}\rangle$  can be implemented as follows:  $\mathbf{B}^{-1}$  performs on the initial state  $|\psi_{\mathbf{A}}\rangle |0_{p_x}\rangle |0_{p_y}\rangle$ , and then project two qumodes onto  $|0_{p_x}\rangle |0_{p_y}\rangle$ . To implement  $\mathbf{B}^{-1}$ , we can write  $\exp(i\mathbf{B} \hat{q}_x \hat{q}_y) = \exp(i\rho_K \hat{q}_x \hat{q}_y) \exp(i\chi \hat{q}_x \hat{q}_y)$ . The first part  $\exp(i\rho_K \hat{q}_x \hat{q}_y)$  can be generated by density matrix exponentiation by sampling from multiple copies of quantum software state  $|\psi_{\mathbf{A}}\rangle$  [7, 30]. The second part is just a basic two-qumode gate.

*Quantum algorithm.*— We now turn to work out a quantum algorithm for nonparametric regression. The main steps are show in Fig. 1, where steps 1 – 4 illustrated in

Fig. 1a transform  $|\psi_{\mathbf{A}}\rangle$  to  $|\psi_{\mathbf{A}+}\rangle$ , and step 5 illustrated in Fig. 1b implements the prediction.

1. *State preparation.* Prepare the data state  $|\psi_{\mathbf{A}}\rangle$ , the query state  $|\psi_{\mathbf{R}}\rangle$ , and a two-qumode state  $|s\rangle_{p_x}|s\rangle_{p_y}$ , where  $|s\rangle_p = s^{1/2}\pi^{-1/4} \int dp e^{-s^2 p^2/2} |p\rangle_p$ . In general,  $|\psi_{\mathbf{A}}\rangle$  can be prepared efficiently with a quantum random access memory [31]. It uses the addressing state  $\sum_m |m\rangle$  to access the memory cells storing quantum states  $|\phi_{\mathbf{x}(m)}\rangle$  in training data registers. Also, two qumodes are initialized in a finite squeezing state  $|s\rangle_{p_x}|s\rangle_{p_y}$ .

2. *Quantum phase estimation.* Perform  $U = \exp(i\rho_K \hat{q}_x \hat{q}_y)$  on  $|\psi_{\mathbf{A}}\rangle|s\rangle_{p_x}|s\rangle_{p_y}$ . Here,  $U$  is constructed with the density matrix exponentiation method [7, 30, 32], which requires to access  $\varepsilon^{-1}$  copies of density matrix  $\rho_K$  when the error is  $\varepsilon$ .

3. *Regularization.* Perform  $e^{i\chi \hat{q}_1 \hat{q}_2}$  on two qumodes. Here  $\chi$  is a preset hyperparameter.

4. *Singular-value transformation.* Project two qumodes into the squeezing state  $|s\rangle_{p_x}|s\rangle_{p_y}$ , and the state turns to be  $|\psi'_{\mathbf{A}+}\rangle = \sum_i f(\lambda_i, s, \chi) |u_i\rangle |v_i\rangle$ , approximating  $|\psi_{\mathbf{A}+}\rangle$ , where  $f(\lambda_i, s, \chi) = \frac{\lambda_i}{\sqrt{\frac{4}{s^4} + (\lambda_i^2 + \chi)^2}}$ .

5. *Prediction.* For a new data  $\tilde{\mathbf{x}}$ , the prediction  $\tilde{y} \propto \langle \psi_{\mathbf{R}} | \psi'_{\mathbf{A}+} \rangle$  can be accessed with a swap test. After the conditional swap operation, an entangled state is obtained,  $|\Psi\rangle = \frac{1}{\sqrt{2}}(|0\rangle \otimes |\psi'_{\mathbf{A}+}\rangle |\Psi_{\mathbf{R}}\rangle + |1\rangle \otimes |\Psi_{\mathbf{R}}\rangle |\psi'_{\mathbf{A}+}\rangle$ . Then, a Hadamard gate is performed on the qubit, followed with a projection into  $|0\rangle$ , whose success rate  $p = \frac{1}{2}(1 + |\langle \psi'_{\mathbf{A}+} | \psi_{\mathbf{R}} \rangle|^2)$  is used to infer the prediction  $\tilde{y} \propto \sqrt{2p-1}$ , up to a sign.

*Quantum advantages.*— We now elaborate that the above algorithm has an exponential speed-up. Using quantum random access memory  $|\psi_{\mathbf{A}}\rangle$  can be prepared in a runtime of  $O(\log M)$ . It takes  $O(\varepsilon^{-1})$  copies of  $|\psi_{\mathbf{A}}\rangle$ , thus a runtime of  $O(\varepsilon^{-1} \log M)$  to perform  $\exp(i\rho_K \hat{q}_x \hat{q}_y)$  [26, 32], for a desired accuracy  $\varepsilon$ . The success rate of homodyne detection is  $O(s^{-4})$  and this procedure thus requires  $O(s^4)$  (see Supplemental Material [27]). In total, the runtime scales as  $O(s^4 \varepsilon^{-1} \log M)$ . The exponential speed-up relies on the capacity of superposition. If randomly chosen  $M' < M$  training data is superposed for each copy [33], then the number of copies should be increased  $O(\frac{M}{M'})$  times. To retain exponential speed-up requires  $M'/M \sim O(1)$ .

Another potential quantum advantage comes from quantum feature map when encoding  $\mathbf{x}$  into  $|\phi_{\mathbf{x}}\rangle$ . Remarkably, continuous variable provides infinite dimension Hilbert space with highly nonlinear feature maps. For instance, encoding into a Gaussian state, such as  $|\phi_{\mathbf{x}}\rangle = \otimes_i |\mathbf{x}_i\rangle_c$  ( $|\mathbf{x}_i\rangle_c$  denotes a coherent state with a displacement  $\mathbf{x}_i$ ), corresponds to a Gaussian kernel, since  $\langle \phi_{\mathbf{u}} | \phi_{\mathbf{v}} \rangle = e^{-|\mathbf{u}-\mathbf{v}|^2/2}$ . Classically intractable instantaneous quantum polynomial or continuous variable instantaneous quantum polynomial circuits are pursued [15, 16]. Moreover, a promising direction is to find encoding schemes that can better represent similarities

between data for specified tasks, and thus require less training data and better generalization, such as predicting ground state energies for molecules [34, 35].

*Quantum operations required in trapped ions.*— Implementing the above quantum algorithm requires hybrid discrete and continuous variable quantum computing. Some promising candidates for quantum computation, such as superconducting qubits in a circuit-QED and trapped ions, have this property. Here we take trapped ions as the platform [18–20] to illustrate the details. We consider trapped ions in a Paul trap, and take  $L$  internal levels of each ions as a qudit to encode the discrete variables and local transverse phonon modes (along  $x$  and  $y$  directions) [36, 37] to encode the continuous variables, while the longitudinal collective modes along  $z$  direction serves as the bus modes to connect any two ions. Notably both internal states and phonon modes are well controllable in trapped ions [37–44].

We outline quantum operations required for the proposed algorithm (see Supplemental Material [27]). We first address the operations acting on single ion, denoting as the  $j$ -th ion. A single qubit gate  $\mathcal{R}(\theta, \mathbf{n}) = e^{i\theta \sigma_{j\mathbf{n}}}$  acting on any two internal levels of the  $j$ -th ion with high fidelity is realizable, where  $\sigma_{j\mathbf{n}}$  is a Pauli matrix along the direction  $\mathbf{n}$ . Operations on a motional mode include  $\mathcal{P}_{\alpha}(\theta) = e^{i\theta a_{j\alpha}^{\dagger} a_{j\alpha}}$ , displacement operator  $\mathcal{D}_{\alpha}(h) = e^{h a_{j\alpha} - h^* a_{j\alpha}^{\dagger}}$  and squeezing operator  $\mathcal{S}_{\alpha}(s) = e^{-\frac{\ln s}{2}(a_{j\alpha}^2 - a_{j\alpha}^{\dagger 2})}$  with  $\alpha = x, y$  [18, 41, 45–47], where  $a_{j\alpha}^{\dagger}$  ( $a_{j\alpha}$ ) is the create (annihilation) operator of the  $\alpha$  phonon mode. A controlled phase gate  $\mathcal{C}_q = e^{ix \hat{q}_j^x \hat{q}_j^y}$  coupling both motional modes can be realized by manipulating the trap potential. By using red and blue side excitations induced by lasers, internal and motional states can be coupled, e.g., obtaining Dirac type operators  $H_1 = g \hat{q}_j^x \sigma_j^x$  and  $H_2 = g \hat{q}_j^y \sigma_j^y$  [39, 40]. Then the hybrid operator  $\mathcal{W}(\eta) = e^{i\eta \sigma_{jz} \hat{q}_j^x \hat{q}_j^y}$ , which is important for quantum phase estimation, can be constructed by repeatedly applying  $1/(g^2 \delta t^2)$  times of the quantum evolution  $e^{iH_2 \delta t} e^{iH_1 \delta t} e^{-iH_2 \delta t} e^{-iH_1 \delta t} = e^{-[H_1, H_2] \delta t^2} + O(\delta t^3)$ .

As for two ions, besides the standard controlled-NOT gate [38], a beam-splitter defined as  $\mathcal{B}(\theta) = e^{i\theta(a_{j\alpha}^{\dagger} a_{j+1\alpha} + a_{j+1\alpha}^{\dagger} a_{j\alpha})}$  is needed, and it was theoretically proposed [37] and then experimentally achieved recently [48]. These two operators thus couple qubit states or qumodes from different ions. Furthermore, a coupling of one qubit from an ion and a qumode from another ion is possible with Dirac type Hamiltonians where spin and momentum (position) come from different ions. Necessary quantum operations on three ions includes controlled swap operators, for which one ion provides a qubit to control a swap for other two ions, either on internal states or motional states. The former has been realized experimentally in trapped ions [49]. On the other hand, precision measurement can be implemented for

both qubits [18] and qumodes [50]. Those unitary operators and measurements serve as building blocks for the quantum algorithm of nonparametric regression as well as other hybrid quantum information processing tasks.

*Physical implementation with trapped ions.*— We illustrate the implementation with a simple example. We just take one ion to encode the training dataset, that is, using only one ion to represent one copy of state  $|\psi_{\mathbf{A}}\rangle$ . To this end, we choose  $L(=M)$  internal levels of the ion as a qudit to encode the  $M$  points dataset and two local transverse phonon modes (along  $x$  and  $y$  directions) to encode the continuous variables ( $N=2$ ).

The implementation needs four types of ions which we denote as  $a, b, c, d$ — ions. 1). An  $a$ -ion provides a qubit and two qumodes as auxiliary modes. 2). A  $b$ -ion is used to store the state  $|\psi_{\mathbf{A}}\rangle$  that encodes all data.  $L$  internal levels and two local motional modes along the  $x, y$  directions are used. On this ion the state will be transformed into the target state  $|\psi_{\mathbf{A}+}\rangle$ . 3). Several  $c$ -ions, the number of which depends on the accuracy required for the algorithm, are used for constructing the unitary operator  $U$  on the  $b$ -ion. Each  $c$ -ion is initialized in the state  $|\psi_{\mathbf{A}}\rangle$ . 4). A  $d$ -ion encodes input data for prediction into quantum state  $|\psi_{\mathbf{R}}\rangle$ .

The scheme for nonlinear regression is schematically shown in Fig. 2a. In the state preparation, the generalized Schrodinger cat states  $|\psi_{j\mathbf{A}}\rangle = \sum_{m=0}^M |m_j\rangle \otimes |\phi_j(\mathbf{x}^{(m)})\rangle$ , where  $|\phi_j(\mathbf{x}^{(m)})\rangle = |\mathbf{x}_{jx}^{(m)}\rangle |\mathbf{x}_{jy}^{(m)}\rangle$  for both  $j = b, c$ —ions, can be generated with Dirac type operations (see Supplemental Material [27]). Also two qumodes of the  $c$ -ion are prepared in a squeezing state  $|s\rangle_{p_x}|s\rangle_{p_y}$ . For the quantum phase estimation, the unitary operation  $U = \exp(i\rho_K \hat{q}_x \hat{q}_y)$  is constructed with the density matrix exponentiation method [7, 26, 30, 32],

$$\begin{aligned} & \text{Tr}_{\rho}(e^{i\delta t \hat{q}_x \hat{q}_y} S_{cv} \rho \otimes \rho' e^{-i\delta t \hat{q}_x \hat{q}_y} S_{cv}) \\ &= e^{i\rho \delta t \hat{q}_x \hat{q}_y} \rho' e^{i\rho \delta t \hat{q}_x \hat{q}_y} + O(\delta t^2). \end{aligned} \quad (6)$$

Here  $\rho = \rho_K$  is a mixed state encoded in the motional states of  $c$ -ion (the internal states are traced out), and  $\rho' \equiv |\psi_{\mathbf{A}}\rangle\langle\psi_{\mathbf{A}}|$  is a state on  $b$ -ion. The conditional swap operator  $e^{i\delta t \hat{q}_x \hat{q}_y} S_{cv}$  is constructed from  $C_{S_{cv}} H_a W_a(\delta t) H_a C_{S_{cv}}$  [35], where  $C_{S_{cv}}$  swaps motional states of  $b$ -ion and  $c$ -ion, conditioned on the qubit state of  $a$ -ion initialized in  $|+\rangle$  state. The one-qubit-two-qumodes coupling  $W_a(\delta t)$  performs on the  $a$ -ion, and  $H_a$  is a Hadamard gate acting on the  $a$ -ion. Multiple copies of  $c$ -ion are required and each is encoded with mixed state  $\rho$  in the internal states. Conditional swap operations are sequentially performed on  $b$ -ion and a new  $c$ -ion and swap their motional states, effectively giving a  $U$  operation on  $b$ -ion.

After applying  $U$  on the  $b$ -ion, a regularization can be realized by applying  $C_q = e^{i\chi \hat{q}_x \hat{q}_y}$  on the two motional modes of  $c$ -ion. A measurement projects two qumodes of the  $c$ -ion onto  $|s\rangle_{p_x}|s\rangle_{p_y}$ . The  $b$ -ion is on target state

$|\psi_{\mathbf{A}+}\rangle$ . After an evolution  $U_R^\dagger$ , where  $U_R|0\rangle = |\psi_{\mathbf{R}}\rangle$ , a projective measure on  $|g\rangle|0\rangle_c$  with the success probability  $p = |\langle\psi_{\mathbf{R}}|\psi_{\mathbf{A}+}\rangle|^2$  can infer the prediction for new data  $\tilde{x}$ .

This implementation scheme can demonstrate a remarkable quantum-enhanced property. The above density matrix exponentiation can use partial training dataset for each time [33], e.g., use  $\rho \sim \sum_{m \in \mathcal{R}_M} |\phi_j(\mathbf{x}^{(m)})\rangle\langle\phi_j(\mathbf{x}^{(m)})|$ , where  $\mathcal{R}_M$  represents to randomly choose  $R_M$  samples in the training dataset, and we thus choose  $L = R_M$  internal levels to represent state  $|\psi_{\mathbf{A}}\rangle$ . Therefore, in the experiments, we can compare the results of  $R_M = 1, 2, \dots, M$  randomly chosen data from the dataset for each copy. We calculate the prediction errors as a function of the number of the  $c$ -ions, and the results are shown in Fig. (2)b. Under the condition of the same accuracy, the number of  $c$ -ions increase with the decrease of  $R_M$ ; Similarly, the prediction errors decrease for a large  $R_M$ . Therefore, it is a clear evidence to demonstrate the power of superposition for quantum nonparametric learning. So a remarkable result shown here is that, a Paul trap with around ten ions, which has been realized in several groups [21–24], can demonstrate the quantum-enhanced property for quantum machine learning.

*Scalability.*— We continue to address how to scale the algorithm up to general  $M$  and  $N$ . Instead of using qudit as discussed in above example, we here consider one qubit for each ion. As each ion provides one qubit and two qumodes, the required ions for encoding  $|\psi_{\mathbf{A}}\rangle$  is  $N_{\text{ion}} = \text{Max}\{\log_2 M, N/2\}$ . Thus  $b, c$  or  $d$ -ions consist of a group of  $N_{\text{ion}}$  ions.

The main procedures of the quantum algorithm are the same with the previous example. A particular concern is how to efficiently prepare many copies of state  $|\psi_{\mathbf{A}}\rangle$ . A direct quantum circuit can sequentially apply  $|m\rangle\langle m| \otimes \mathcal{D}(\mathbf{x}^{(m)})$ , which can be decomposed into a generalized Toffoli gate and a hybrid gate involving only one qubit and one qumode (see Supplemental Material [27]). This scheme to encode  $|\psi_{\mathbf{A}}\rangle$  into  $N_{\text{ion}}$  ions is feasible but not efficient, since the time complexity is  $M$ . We may speed up the state preparation of  $|\psi_{\mathbf{A}}\rangle$  to a runtime of  $\log M$  using quantum random access memory. Assuming this efficient state preparation, the quantum algorithm has an exponential speed-up with the number of sample  $M$ , and is independent of the number of features  $N$ .

We wish to pinpoint that a set of classical data  $\mathbf{x} \in R^N$  can be encoded into a general  $N$ -mode Gaussian state with other Gaussian gates such as beam splitter, squeezing operator and so on [51]. This may be useful as interactions between different original features are taken into account.

To conclude, we have illustrated a quantum paradigm of nonparametric learning that can fully exploit quantum advantages with realistic physical implementation. The above-proposed experimental scheme has paved the way for quantum machine learning.



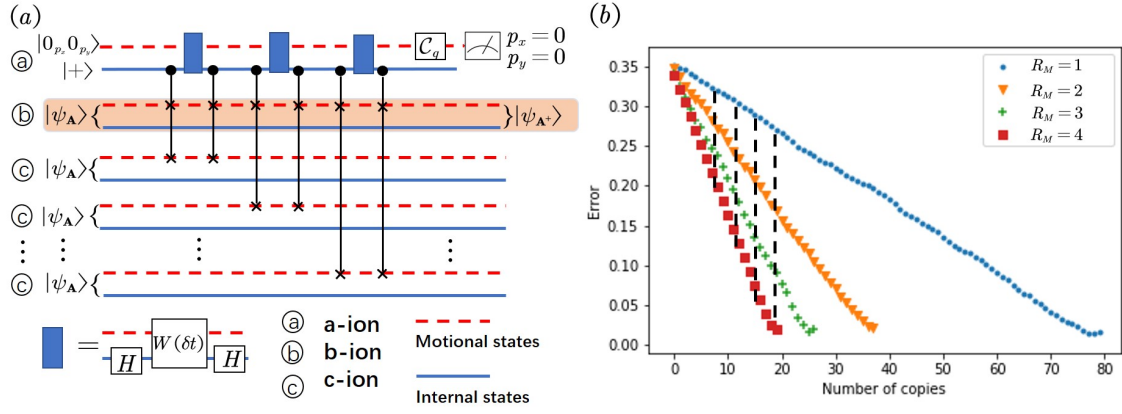


FIG. 2. (a). A quantum procedure that transforms  $|\psi_A\rangle$  into  $|\psi_{A+}\rangle$  on *b*-ion, assisted by an *a*-ion providing a qubit for control and two qumodes for matrix inversion, and many *c*-ions initialized in  $|\psi_A\rangle$  serving as quantum software states for quantum phase estimation. Note the swap only performs on motional states (red dash lines). (b). Density matrix exponentiation where partial training dataset is superposed. Here  $R_M = 1, 2, 3, 4$  stands for the number of randomly chosen samples for each copy. The region with dashed lines represents accessible zone for a trade-off between error and the number of ions involved, constraint by the maximum available *c*-ion  $N_t = 20$ .

This work was supported by the National Key Research and Development Program of China (Grant No. 2016YFA0301800), the National Natural Science Foundation of China (Grants No. 91636218, No.11474153, and No. U1801661), the Key Project of Science and Technology of Guangzhou (Grant No. 201804020055).

\* slzhu@nju.edu.cn

† zwang@hkucc.hku.hk

- [1] C. M. Bishop, *Pattern recognition and machine learning*, Vol. 1 (Springer, 2006).
- [2] Trevor Hastie, Robert Tibshirani, and Jerome Friedman, *The Elements of Statistical Learning*, 2nd ed. (Springer, 2009).
- [3] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, and S. Lloyd, “Quantum machine learning,” *Nature* **549**, 195–202 (2017).
- [4] Sankar Das Sarma, Dong-Ling Deng, Lu-Ming Duan, Machine learning meets quantum physics, *Physics Today* **72**, 3, 48 (2019).
- [5] A. W. Harrow, A. Hassidim, and S. Lloyd, “Quantum algorithm for linear systems of equations,” *Phys. Rev. Lett.* **103**, 150502 (2009).
- [6] N. Wiebe, D. Braun, and S. Lloyd, “Quantum algorithm for data fitting,” *Phys. Rev. Lett.* **109**, 050505 (2012).
- [7] S. Lloyd, M. Mohseni, and P. Rebentrost, “Quantum principal component analysis,” *Nat. Phys.* **10**, 631–633 (2014).
- [8] P. Rebentrost, M. Mohseni, and S. Lloyd, “Quantum support vector machine for big data classification,” *Phys. Rev. Lett.* **113**, 130503 (2014).
- [9] V. Dunjko, J. M. Taylor, and H. J. Briegel, “Quantum-enhanced machine learning,” *Phys. Rev. Lett.* **117**, 130501 (2016).
- [10] S. Lloyd, S. Garnerone, and P. Zanardi, “Quantum al-

gorithms for topological and geometric analysis of data,” *Nature Communications* **7**, 10138 (2016).

- [11] Seth Lloyd and Christian Weedbrook, “Quantum generative adversarial learning,” *Phys. Rev. Lett.* **121**, 040502 (2018).
- [12] Vojtech Havlek, Antonio D. Croles, Kristan Temme, Aram W. Harrow, Abhinav Kandala, Jerry M. Chow, and Jay M. Gambetta, “Supervised learning with quantum-enhanced feature spaces,” *Nature* **567**, 209–212 (2019).
- [13] Maria Schuld and Nathan Killoran, “Quantum machine learning in feature hilbert spaces,” *Phys. Rev. Lett.* **122**, 040504 (2019).
- [14] K. Mitarai, M. Negoro, M. Kitagawa, and K. Fujii, “Quantum circuit learning,” *Phys. Rev. A* **98**, 032309 (2018).
- [15] Michael J. Bremner, Ashley Montanaro, and Dan J. Shepherd, “Average-case complexity versus approximate simulation of commuting quantum computations,” *Phys. Rev. Lett.* **117**, 080501 (2016).
- [16] T. Douce, D. Markham, E. Kashefi, E. Diamanti, T. Coudreau, P. Milman, P. van Loock, and G. Ferrini, “Continuous-variable instantaneous quantum computing is hard to sample,” *Phys. Rev. Lett.* **118**, 070503 (2017).
- [17] Andrs Gilyn, Yuan Su, Guang Hao Low, and Nathan Wiebe, “Quantum singular value transformation and beyond: exponential improvements for quantum matrix arithmetics,” arXiv preprint arXiv:1806.01838 (2018).
- [18] D. Leibfried, R. Blatt, C. Monroe, and D. Wineland, “Quantum dynamics of single trapped ions,” *Rev. Mod. Phys.* **75**, 281–324 (2003).
- [19] H. Haeffner, C. F. Roos, R. Blatt, Quantum computing with trapped ions, *Phys. Rep.* **469**, 155 (2008).
- [20] C. Monroe and J. Kim, Scaling the Ion Trap Quantum Processor, *Science* **339**, 1164 (2013).
- [21] J. Zhang, G. Pagano, P. W. Hess, A. Kyprianidis, P. Becker, H. Kaplan, A. V. Gorshkov, Z. X. Gong, and C. Monroe, “Observation of a many-body dynamical phase transition with a 53-qubit quantum simulator,”

- Nature* **551**, 601 (2017).
- [22] Nicolai Friis, Oliver Marty, Christine Maier, Cornelius Hempel, Milan Holzpfel, Petar Jurcevic, Martin B. Plenio, Marcus Huber, Christian Roos, Rainer Blatt, and Ben Lanyon, “Observation of entangled states of a fully controlled 20-qubit system,” *Physical Review X* **8**, 021012 (2018).
  - [23] C. Kokail, C. Maier, R. van Bijnen, T. Brydges, M. K. Joshi, P. Jurcevic, C. A. Muschik, P. Silvi, R. Blatt, C. F. Roos, and P. Zoller, “Self-verifying variational quantum simulation of lattice models,” *Nature* **569**, 355–360 (2019).
  - [24] K. Wright, K. M. Beck, S. Debnath, J. M. Amini, Y. Nam, N. Grzesiak, J. -S. Chen, N. C. Pienti, M. Chmielewski, C. Collins, K. M. Hudek, J. Mizrahi, J. D. Wong-Campos, S. Allen, J. Apisdorf, P. Solomon, M. Williams, A. M. Ducre, A. Blinov, S. M. Kreikemeier, V. Chaplin, M. Keesan, C. Monroe, J. Kim, Benchmarking an 11-qubit quantum computer, arXiv:1903.08181.
  - [25] M. Schuld, I. Sinayskiy, and F. Petruccione, “Prediction by linear regression on a quantum computer,” *Phys. Rev. A* **94**, 022342 (2016).
  - [26] Dan-Bo Zhang, Zheng-Yuan Xue, Shi-Liang Zhu, and Z. D. Wang, “Realizing quantum linear regression with auxiliary qumodes,” *Phys. Rev. A* **99**, 012331 (2019).
  - [27] Supplemental Material.
  - [28] Andrew M. Childs, Robin Kothari, and Rolando D. Somma, “Quantum algorithm for systems of linear equations with exponentially improved dependence on precision,” *Siam. J. Comput.* **46**, 1920–1950 (2017).
  - [29] Juan Miguel Arrazola, Timjan Kalajdziewski, Christian Weedbrook, and Seth Lloyd, “Quantum algorithm for non-homogeneous linear partial differential equations,” arXiv:1809.02622 (2018).
  - [30] Shelby Kimmel, Cedric Yen-Yu Lin, Guang Hao Low, Maris Ozols, and Theodore J. Yoder, “Hamiltonian simulation with optimal sample complexity,” *npj Quantum Information* **3**, 13 (2017).
  - [31] V. Giovannetti, S. Lloyd, and L. Maccone, “Architectures for a quantum random access memory,” *Phys. Rev. A* **78** (2008).
  - [32] H. K. Lau, R. Pooser, G. Siopsis, and C. Weedbrook, “Quantum machine learning over infinite dimensions,” *Phys. Rev. Lett.* **118**, 080501 (2017).
  - [33] Patrick Rebentrost, Thomas R. Bromley, Christian Weedbrook, and Seth Lloyd, “Quantum hopfield neural network,” *Phys. Rev. A* **98**, 042308 (2018).
  - [34] Matthias Rupp, Alexandre Tkatchenko, Klaus-Robert Müller, and O. Anatole von Lilienfeld, “Fast and accurate modeling of molecular atomization energies with machine learning,” *Phys. Rev. Lett.* **108**, 058301 (2012).
  - [35] Dan-Bo Zhang, Shi-Liang Zhu, and Z. D. Wang, “Non-linear regression based on a hybrid quantum computer,” arXiv:1808.09607 (2018).
  - [36] Shi-Liang Zhu, C. Monroe, and L. M. Duan, “Trapped ion quantum computation with transverse phonon modes,” *Phys. Rev. Lett.* **97**, 050505 (2006).
  - [37] C. Shen, Z. Zhang, and L. M. Duan, “Scalable implementation of boson sampling with trapped ions,” *Phys. Rev. Lett.* **112**, 050504 (2014).
  - [38] J. I. Cirac and P. Zoller, “Quantum computations with cold trapped ions,” *Phys. Rev. Lett.* **74**, 4091–4094 (1995).
  - [39] L. Lamata, J. Leon, T. Schatz, and E. Solano, “Dirac equation and quantum relativistic effects in a single trapped ion,” *Phys. Rev. Lett.* **98**, 253005 (2007).
  - [40] R. Gerritsma, G. Kirchmair, F. Zähringer, E. Solano, R. Blatt, and C. F. Roos, “Quantum simulation of the Dirac equation,” *Nature* **463**, 68–U72 (2010).
  - [41] Hoi-Kwan Lau and Daniel F. V. James, “Proposal for a scalable universal bosonic simulator using individually trapped ions,” *Phys. Rev. A* **85**, 062329 (2012).
  - [42] Luis Ortiz-Gutierrez, Bruna Gabrielly, Luis F. Muñoz, Kain T. Pereira, Jefferson G. Filgueiras, and Alessandro S. Villar, “Continuous variables quantum computation over the vibrational modes of a single trapped ion,” *Opt. Commun.* **397**, 166–174 (2017).
  - [43] Junhua Zhang, Mark Um, Dingshun Lv, Jing-Ning Zhang, Lu-Ming Duan, and Kihwan Kim, “Noon states of nine quantized vibrations in two radial modes of a trapped ion,” *Phys. Rev. Lett.* **121**, 160502 (2018).
  - [44] C. Flühmann, T. L. Nguyen, M. Marinelli, V. Negnevitsky, K. Mehta, and J. P. Home, “Encoding a qubit in a trapped-ion mechanical oscillator,” *Nature* **566**, 513–517 (2019).
  - [45] J. I. Cirac, A. S. Parkins, R. Blatt, and P. Zoller, ““dark” squeezed states of the motion of a trapped ion,” *Phys. Rev. Lett.* **70**, 556–559 (1993).
  - [46] D. Kienzler, H. Y. Lo, B. Keitch, L. de Clercq, F. Leupold, F. Lindemeyer, M. Marinelli, V. Negnevitsky, and J. P. Home, “Quantum harmonic oscillator state synthesis by reservoir engineering,” *Science* **347**, 53 (2015).
  - [47] S. C. Burd, R. Srinivas, J. J. Bollinger, A. C. Wilson, D. J. Wineland, D. Leibfried, D. H. Slichter, and D. T. C. Allcock, “Quantum amplification of mechanical oscillator motion,” arXiv:1812.01812 (2018).
  - [48] Kenji Toyoda, Ryoto Hiji, Atsushi Noguchi, and Shinji Urabe, “Hongoumandel interference of two phonons in trapped ions,” *Nature* **527**, 74 (2015).
  - [49] N. M. Linke, S. Johri, C. Figgatt, K. A. Landsman, A. Y. Matsuura, and C. Monroe, “Measuring the Rényi entropy of a two-site fermi-hubbard model on a trapped ion quantum computer,” *Phys. Rev. A* **98**, 052334 (2018).
  - [50] J. F. Poyatos, R. Walser, J. I. Cirac, P. Zoller, and R. Blatt, “Motion tomography of a single trapped ion,” *Phys. Rev. A* **53**, R1966–R1969 (1996).
  - [51] Christian Weedbrook, Stefano Pirandola, Ral Garca-Patrón, Nicolas J. Cerf, Timothy C. Ralph, Jeffrey H. Shapiro, and Seth Lloyd, “Gaussian quantum information,” *Rev. Mod. Phys.* **84**, 621–669 (2012).