Tomography and generative training with quantum Boltzmann machines

Mária Kieferová
Institute for Quantum Computing, University of Waterloo, Ontario, Canada
and Macquarie University, NSW, 2109, Australia

Nathan Wiebe
Microsoft Research, Washington, USA

(Received 14 January 2017; revised manuscript received 1 August 2017; published 22 December 2017)

The promise of quantum neural nets, which utilize quantum effects to model complex data sets, has made their development an aspirational goal for quantum machine learning and quantum computing in general. Here we provide methods of training quantum Boltzmann machines. Our work generalizes existing methods and provides additional approaches for training quantum neural networks that compare favorably to existing methods. We further demonstrate that quantum Boltzmann machines enable a form of partial quantum state tomography that further provides a generative model for the input quantum state. Classical Boltzmann machines are incapable of this. This verifies the long-conjectured connection between tomography and quantum machine learning. Finally, we prove that classical computers cannot simulate our training process in general unless BQP = BPP, provide lower bounds on the complexity of the training procedures and numerically investigate training for small nonstoquastic Hamiltonians.

DOI: 10.1103/PhysRevA.96.062327

I. INTRODUCTION

Quantum machine learning is formed out of the confluence of machine learning and quantum computation. The use of quantum computing allows speed-ups and improved models for data in machine learning algorithms such as support vector machines [1], nearest-neighbor classification [2,3], boosting [4,5], and many others [6–9]. Artificial neural networks play a prominent role in machine learning because of their wide array of application. Their quantum counterparts are a much newer concept with open questions about efficient learning methods, computational power, and utility.

Here we present several training methods for a class of fully quantum neural networks known as quantum Boltzmann machines. Our work firmly establishes a connection between quantum neural net training and quantum state estimation, which is colloquially referred to as tomography. Assuming an approximation of a Gibbs state for the Hamiltonian in question can be prepared efficiently, we show that training quantum analogs of Boltzmann machines can be used to estimate quantum states efficiently.

The Boltzmann machine is a physically motivated neural network capable of generating new examples similar to the training data [10]. The close connection to physical systems has made Boltzmann machines a natural fit for quantum annealing [11–13] and quantum computing [3], the latter showing polynomial speed-ups relative to classical training [14]. While these methods showed that quantum technologies can train Boltzmann machines more accurately and at a lower cost than classical methods, the question of whether transitioning from an Ising model to a quantum model for the data would provide substantial improvements remained open.

This question is addressed in Ref. [15], wherein a method for training Boltzmann machines is provided that utilizes transverse Ising models in thermal equilibrium to model data. While such models are trainable and can outperform classical Boltzmann machines, the training procedure proposed therein suffers two drawbacks. First, the transverse field cannot be learned from the classical data. These terms must be found through brute force techniques, which makes finding the full Hamiltonian much more difficult. Second, the transverse Ising models considered are widely believed to be efficiently simulatable using quantum Monte Carlo methods and therefore does not provide a clear quantum advantage.

Here we look at quantum Boltzmann machines (QBMs) in a much broader context and investigate their performance for models that are manifestly quantum. We introduce the ability to learn the quantum terms of the model as well as the classical ones and generalize the training set to include quantum data sets. We show that these freedoms now allow quantum Boltzmann machines to act as approximate cloners for quantum states. That is, given exposure to enough copies of a density operator, a QBM can be trained to produce copies of an input state. This is a quantum analog of generative training that cannot be replicated by classical Boltzmann machines. We also provide numerical evidence that quantum Boltzmann machines are also more powerful than equivalently sized classical Boltzmann machines for classical machine learning problems.

II. BOLTZMANN MACHINE

A Boltzmann machine represents data as a thermal state of an Ising model Hamiltonian. The Hamiltonian is defined on a graph where the vertices represent spins and edges interactions between them. The vertices can be either visible units, used as input and output, or hidden units, which provide extra degrees of freedom for the model. The faithfulness of this representation is measured by KL divergence. Formally, the KL divergence quantifies the information loss occurred if the distribution generated from the model Q replaces the underlying distribution of data P:

$$D_{\mathrm{KL}}(P\|Q) = \sum_i P_i \ln \frac{P_i}{Q_i}. \tag{1}$$

The goal in training is to minimize $D_{\mathrm{KL}}(P\|Q)$, which is equivalent to maximizing the log-likelihood $\sum_i P_i \ln Q_i$.

The generalization of Boltzmann machine training into quantum setting is not unique. We propose two methods that we refer to as POVM-based Golden-Thompson training and state-based relative entropy training. The two approaches present different quantum analogs of training set and objective function. In the classical setting, the training set is a set of vectors representing the data. The training set for QBM is expected to include quantum data as well. This can be achieved with a set of POVM or a density matrix. Another source of ambiguity is the quantum objective function. Amin *et al.* in Ref. [15] introduced a quantum version of log-likelihood and an approximation to its gradient. We also propose training using the relative entropy, which is a quantum equivalent of KL divergence.

III. RELATIVE ENTROPY TRAINING

The relative entropy is the quantum equivalent of KL divergence and as such represents a natural extension for learning of quantum states. It is defined as

$$S(\rho\|\sigma) = \mathrm{Tr}(\rho\ln\rho - \rho\ln\sigma), \tag{2}$$

where ρ is the data distribution and $\sigma = e^{-\beta H}/Z$ is the thermal state generated by QBM. This gives a generalization of the training set for quantum data. We do not impose rules on the structure of the Hamiltonian as far as it is a smooth function of the parameters used to describe the Boltzmann model.

The objective function that we wish to maximize is

$$\mathcal{O}_\rho(H;\lambda) = \mathrm{Tr}\left[\rho\ln\left(\frac{e^{-H}}{\mathrm{Tr}[e^{-H}]}\right)\right], \tag{3}$$

which is equivalent to minimizing $S(\rho\|e^{-H}/\mathrm{Tr}[e^{-H}])$. The quantum terms can also be regularized by including a term proportional to the sum of the squares of their coefficients to $\mathcal{O}_\rho$. This penalizes quantum terms in Hamiltonians unless they are needed to explain the data.

The derivatives of $\mathcal{O}_\rho$ are

$$-\mathrm{Tr}[\rho\partial_\theta H] + \mathrm{Tr}[e^{-H}\partial_\theta H]/\mathrm{Tr}[e^{-H}]. \tag{4}$$

Thus, we can systematically make the state generated by a simulator of e^{-H}/Z harder to distinguish from the state ρ by following a gradient given by the difference between expectations of the Hamiltonian terms in the data distribution ρ and the corresponding expectation values for e^{-H}/Z. $\mathcal{O}_\rho$ is motivated by the fact that $S(\rho||e^{-H}/Z) \geqslant \|\rho - e^{-H}/Z\|^2/2\ln(2)$ if ρ is positive definite. Thus if ρ has maximum rank, $S(\rho||e^{-H}/Z) \to 0$ implies $e^{-H}/Z \to \rho$.

There are two major advantages to this method. First, no approximations are needed to compute the gradient. Second, it directly enables a form of partial tomography wherein a model for the state ρ is provided by the Hamiltonian learned through the gradient ascent process. Note that this differs from traditional quantum state tomography in that it does not output an explicit representation of the state operator, but instead it gives a prescription for a quantum process that yields approximate copies of the state. This is further interesting because our procedure is efficient, given that an accurate and efficient approximation to the thermal state can be prepared for H, and thus it can be used to describe states in high dimensions such as ground states for complex molecules that would be beyond the reach of standard tomography. The procedure also provides an explicit procedure for generating copies of this inferred state, unlike conventional tomographic methods, which allows it to be used as a surrogate for QRAM in quantum machine learning algorithms.

IV. GOLDEN-THOMPSON TRAINING

Generalization of classical Boltzmann training inspired by Amin *et al.* seeks to find a quantum analog of average log-likelihood $L = \sum_v P_v \ln Q_v$ as in (1). The probability Q_i of observing $|v\rangle$ corresponding to a classical state on the visible units can be translated to quantum setting as $Q_v = \mathrm{Tr}[e^{-H}\Lambda_v]/\mathrm{Tr}[e^{-H}]$ where H is the Hamiltonian defining the Gibbs state. The projector $\Lambda_v = |v\rangle\langle v| \otimes \mathbb{1}$ corresponds to visible units clamped to a training binary state v and the training set is a set of such projectors.

The goal of this approach is to maximize the objective function

$$\mathcal{O}_\Lambda(H;\lambda) = \sum_{\mathbf{v}} P_{\mathbf{v}} \ln\left(\frac{\mathrm{Tr}[e^{-H_v}]}{\mathrm{Tr}[e^{-H}]}\right), \tag{5}$$

where $H_v = H - \ln\Lambda_v$. This approximation is a lower bound on L and is tight when $[H,\Lambda_v] = 0$.

While this objective function is unlikely to be generically computable because the calculation of $\mathrm{Tr}[e^{-H}]$ is a #P-hard problem, the gradients of the objective function are in practice not hard to estimate. The components of the gradient of $\mathcal{O}_\Lambda$ are:

$$\sum_{\mathbf{v}} P_{\mathbf{v}}\left(-\frac{\mathrm{Tr}[e^{-H_v}\partial_\theta H]}{\mathrm{Tr}[e^{-H_v}]} + \frac{\mathrm{Tr}[e^{-H}\partial_\theta H]}{\mathrm{Tr}[e^{-H}]}\right). \tag{6}$$

Note that this training method does not allow gradients to be computed for terms where $\mathrm{Tr}[e^{-H_v}\partial_\theta H] = 0$. As a consequence, Amin *et al.* in Ref. [15] were not able to use this form of training to learn nondiagonal terms of the Hamiltonian.

Our contribution is the realization that POVMs provide the natural way to express the training set for this type of training. Our formalism avoids the problem of $\mathrm{Tr}[e^{-H_v}\partial_\theta H] = 0$ by explicitly including POVM elements that are nondiagonal. This is always possible for classical distribution because any classical probability distribution over training vectors can be viewed as the distribution of measurements over pure states. This freedom grants us the ability to always pick nondiagonal terms.

As a clarifying example, consider the following training set. Let us imagine that we wish to train a model that generates even numbers between 1 and 16. Then a sensible training set would be

$$\Lambda_n = |2n\rangle\langle 2n| \text{ for } 1 \leqslant n \leqslant 8$$
$$\Lambda_0 = \mathbb{1} - \sum_{n=1}^{8}\Lambda_n, \qquad P_v = (1-\delta_{v,0})/8. \tag{7}$$

The following equivalent training set can also be used

$$\Lambda_1 = \tfrac{1}{8}(|2\rangle + \cdots + |16\rangle)(\langle 2| + \cdots + \langle 16|),$$
$$\Lambda_0 = \mathbb{1} - \Lambda_1, \qquad P_v = \delta_{v,1}. \tag{8}$$

This ambiguity about the form of the training set reveals that POVM for quantum Boltzmann training can be nontrivial even when a single training vector is used. This allows us to choose a POVM that circumvents problems faced when $\mathrm{Tr}[\partial_\theta H e^{-H_v}] = 0$. The example of the transverse-Ising model considered in Ref. [15] implicitly uses (7). The model expectation term in the gradient of the transverse terms in this example satisfies $\mathrm{Tr}[X e^{-[H-\ln(|n\rangle\langle n|)]}] = 0$ with that choice, whereas if (8) were used then $\mathrm{Tr}[X e^{-[H-\ln(\Lambda_1)]}] \neq 0$. Thus choosing the training set (8) avoids the problems seen in Ref. [15].

Formally, we define the training set to be the following. Let $\mathcal{H} := \mathcal{V} \otimes \mathcal{L}$ be a finite-dimensional Hilbert space let $\mathcal{V}$ and $\mathcal{L}$ be subsystems corresponding to the visible and latent units of the QBM. The probability distribution P_v and POVM $\Lambda = \{\Lambda_v\}$, comprise a training set for QBM training if (i) there exists a bijection between the domain of P_v and Λ and (ii) the domain of each Λ_v is $\mathcal{H}$ and it acts nontrivially only on subsystem $\mathcal{V}$.

V. COMPLEXITY ANALYSIS

With the expressions for the gradients of the training objective functions in hand, we can now proceed to bound the complexity of training the Boltzmann machine by gradient ascent.

Let us start by explaining the cost model. We assume that we have an oracle, $F_H(\epsilon_H)$, that is capable of taking the weights and biases of the quantum Boltzmann machine (or equivalently a parametrization of H) and outputs the state σ such that $\|\sigma - e^{-H}/Z\|_{tr} \leqslant \epsilon_H$ for $\epsilon_H \geqslant 0$. We manifestly assume that the state preparation is not exact because any computational model that grants the ability to prepare exact Gibbs states for arbitrary Hamiltonians is likely to be more powerful than quantum computing under reasonable complexity theoretic assumptions. For relative entropy training, we also assume that the training data ρ is provided by a query to an auxiliary oracle F_ρ. We cost both oracles equivalently. Finally, we assume for POVM training that the POVM elements can be prepared with a constant sized circuit and do not assign a cost to implementing such a term. We do this for two reasons. First, for most elementary examples the POVM elements are very simple projectors and are not of substantially greater complexity than implementing a Hamiltonian term. The second is that incorporating a cost for them would necessitate opening the black-box F_H, which would substantially complicate our discussion and force us to specialize to particular state preparation methods.

The first result that we show is a lower bound based on tomographic bounds that shows that quantum Boltzmann training cannot be efficient in general if we wish to provide a highly accurate generative model for the training data.

Lemma 1. The number of queries to F_ρ, which yields copies of rank r state operator $\rho \in \mathbb{C}^{D\times D}$ required to train an arbitrary quantum Boltzmann machine using relative entropy such that the quantum state generated by the Boltzmann machine are within trace distance $\epsilon \in (0,1)$ of ρ, and with failure probability $\Theta(1)$, is in $\Omega(Dr/[\epsilon^2 \ln(D/r\epsilon)])$.

Proof. The proof follows by contradiction. Since we have assumed an arbitrary quantum Boltzmann machine we will consider a Boltzmann machine that has a complete set of Hamiltonian terms. If we do not make this assumption then there will be certain density operators that cannot be prepared within error ϵ for all $\epsilon > 0$. Let us assume that ρ is rank D if this is true then there exists $H \in \mathbb{C}^{D\times D}$ such that $\rho \propto e^{-H}$ because the matrix logarithm is well defined for such systems.

Now let us assume that ρ has rank less than D. If that is the case then there does not exist $H \in \mathbb{C}^{D\times D}$ such that $\rho \propto e^{-H}$, but ρ can be closely approximated by it. Let P_0 be a projector onto the null space of ρ, which we assume is $D-r$ dimensional. Then let $\tilde{\rho} \in \mathbb{C}^{r\times r}$ be the projection of ρ onto the orthogonal compliment of its null space. Since ρ is maximum rank within this subspace, there exists $\tilde{H} \in \mathbb{C}^{r\times r}$ such that $\tilde{\rho} \propto e^{-\tilde{H}}$. After a trivial isometric extension of $\tilde{H}$ to $\mathbb{C}^{D\times D}$, we can then write $\rho \propto (\mathbb{1} - P_0)e^{-\tilde{H}}(\mathbb{1} - P_0)$. By construction $[\tilde{H}, (\mathbb{1} - P_0)] = 0$, and thus $\rho \propto (\mathbb{1} - P_0)e^{-\tilde{H}} = (\mathbb{1} - P_0)e^{-(\mathbb{1}-P_0)\tilde{H}(\mathbb{1}-P_0)}$.

The definition of the trace norm implies that for any $\gamma > 0$, $\|(\mathbb{1} - P_0) - e^{-\gamma P_0}\|_1 \in O([D-r]e^{-\gamma})$. Thus because $e^{-(\mathbb{1}-P_0)\tilde{H}(\mathbb{1}-P_0)}/Z$ has trace norm 1

$$\begin{aligned}\rho &= e^{-\gamma P_0} e^{-(\mathbb{1}-P_0)\tilde{H}(\mathbb{1}-P_0)}/Z + O([D-r]e^{-\gamma}) \\ &= e^{-(\mathbb{1}-P_0)\tilde{H}(\mathbb{1}-P_0)-\gamma P_0}/Z + O([D-r]e^{-\gamma}). \end{aligned} \tag{9}$$

Thus ρ can be approximated within error less than ϵ, regardless of its rank, by a Hermitian matrix whose norm scales at most as $O(\|\tilde{H}\| + \ln(D/\epsilon))$. Thus for every $\epsilon > 0$ there exists a quantum Boltzmann machine with a complete set of Hamiltonian terms that can approximate ρ within trace distance less than ϵ using a bounded Hamiltonian.

Haah, Harrow *et al.* show in Theorem 1 of Ref. [16] that $\Omega(Dr/[\epsilon^2 \ln(D/r\epsilon)])$ samples are needed to tomographically reconstruct a rank r density operator $\rho \in \mathbb{C}^{D\times D}$ within error ϵ in the trace distance. Since training a Boltzmann machine can provide a specification of an arbitrary density matrix, to within trace distance ϵ, if this training process required $o(Dr/[\epsilon^2 \ln(D/r\epsilon)])$ samples we would violate their lower bound on tomography. The result therefore follows. ■

Lemma 2. There does not exist a general purpose POVM-based training algorithm for quantum Boltzmann machines on a training set such that $|\{P_v : P_v > 0\}| = N$ can prepare a thermal state such that $\mathrm{Tr}([\sum_v P_v \Lambda_v]e^{-H}/Z) \geqslant 1/\Delta$, which requires M queries to P_v where $\Delta \in o(\sqrt{N})$ and $M \in o(\sqrt{N})$.

Proof. The proof is a reduction of Grover's search to Boltzmann training. We aim to use queries to the black-box oracle to learn a white-box oracle that we can query to learn the marked state without actually querying the original box. To be clear, let us pick $\Lambda_0 = |0\rangle\langle 0|$ and $P_1 = 1$ and for $v > 1$, $\Lambda_v = |v\rangle\langle v|$ with $P_v = 0$. These elements form a POVM because they are positive and sum to the identity.

In the above construction the oracle that gives the P_v is equivalent to the Grover oracle. This implies that a query to this oracle is the same as a query to Grover's oracle.

Now let us assume that we can train a Boltzmann machine such that $\mathrm{Tr}(\Lambda_0 e^{-H}/Z) \in \omega(1/\sqrt{N})$ using $o(\sqrt{N})$ queries to the black box. This implies that $o(\sqrt{N})$ queries are needed on average to prepare $|0\rangle$ by drawing samples from the BM and verifying them using the oracle. Since the cost of learning the BM is also $o(\sqrt{N})$, this implies that the number of queries needed in total is $o(\sqrt{N})$. Thus we can perform quantum search

under these assumptions using $o(\sqrt{N})$ queries and hence from lower bounds this implies $o(\sqrt{N}) \subseteq \Omega(\sqrt{N})$, which is a contradiction. ■

The above lemmas preclude general efficient Boltzmann training without further assumptions about the training data, or without making less onerous requirements on the precision of the BM model output by the training algorithm. This means that we cannot expect even quantum Boltzmann machines to have important limitations that need to be considered when we examine the complexity of quantum machine learning algorithms.

Theorem 1. Let $H = \sum_{j=1}^{M} \theta_j H_j$ with $\|H_j\|_2 = 1 \forall j$ be the Hamiltonian for a quantum Boltzmann machine where for notational simplicity the Λ_v present in POVM-based training are included into the Hamiltonian and that the thermal states of all such models are accessed only through querying a quantum subroutine $F_H(\epsilon_H)$ such that $\|e^{-H}/Z - F_H(\epsilon_H)\|_{\mathrm{tr}} \leqslant \epsilon_H$. Further let G be an approximation to $\nabla\mathcal{O}$ where $\mathcal{O}$ is the training objective function for either POVM based or relative entropy training. If $\epsilon > \sqrt{M}\epsilon_H$ then there exist training algorithms that yield a gradient $\mathbb{E}(\|G - G_{\mathrm{true}}\|_2^2) \leqslant \epsilon^2$ and query $F_H(\epsilon_H)$ and the training set

$$O\left(\frac{M}{\epsilon^2 - M\epsilon_H^2}\right)$$

times per epoch.

Proof. We show the proof by considering the approximate gradients given by the methods in the main body. The algorithm estimates the gradient by sampling the expectation values of local Hamiltonians in the approximate thermal states yielded by $F_H(\epsilon_H)$. The true gradient is the vector of expectation value of local Hamiltonians measured in the thermal state $G_{\mathrm{true}} = \sum_{j=1}^{M} \mathrm{Tr}(H_j e^{-H})/Z$. Thus

$$\begin{aligned}
&\mathbb{E}\left(\|G - G_{\mathrm{true}}\|_2^2\right) \\
&\quad = \sum_{j=1}^{M} \mathbb{E}\left(\left(G^j - G^j_{\mathrm{true}}\right)^2\right) \\
&\quad = \sum_{j=1}^{M} \mathbb{E}((G^j)^2) - 2\mathbb{E}\left(G^j G^j_{\mathrm{true}}\right) + \mathbb{E}\left(\left(G^j_{\mathrm{true}}\right)^2\right) \\
&\quad = \sum_{j=1}^{M} \mathbb{V}(G_j) + \mathbb{E}(G^j)^2 - G^j_{\mathrm{true}} 2\mathbb{E}(G^j) + \left(G^j_{\mathrm{true}}\right)^2 \\
&\quad = \sum_{j=1}^{M} \mathbb{V}(G^j) + \left(\mathbb{E}(G^j) - G^j_{\mathrm{true}}\right)^2. \qquad (10)
\end{aligned}$$

The expectation value of the gradient component $\mathbb{E}(G^j) = \mathrm{Tr}(H_j\sigma)$ where σ is approximation of the thermal state e^{-H}/Z such that $\|\sigma - e^{-H}/Z\|_{\mathrm{tr}} \leqslant \epsilon_H$ can be bounded using standard properties of the trace norm as

$$\|H_j e^{-H}/Z - \sigma H_j\|_{\mathrm{tr}} \leqslant \|e^{-H}/Z - \sigma\|_{\mathrm{tr}} \|H_j\|_2 \leqslant \epsilon_H. \quad (11)$$

Thus $|G^j_{\mathrm{true}} - \mathbb{E}(G^j)| \leqslant \epsilon_H$ under the assumption that $\|H_j\|_2 \leqslant 1$ for all j.

For relative entropy training the variance can be estimated

$$\mathbb{V}(G_j) \in O(\max\{\mathrm{Tr}(\rho H_j), \mathrm{Tr}(H_j\sigma)\}) \in O(1/n), \qquad (12)$$

where ρ is the density matrix corresponding to the ensemble of training vectors.

Similarly for POVM training

$$\begin{aligned}
\mathbb{V}(G_j) &\in O\left(\max\left\{\mathrm{Tr}\left(H_j \sum_v P_v \sigma\right), \mathrm{Tr}(H_j\sigma)\right\} \Big/ n\right) \\
&\in O(1/n). \qquad (13)
\end{aligned}$$

Note that in this context we have implicitly allowed the POVM elements to be considered as Hamiltonian terms in the Boltzmann machine. Thus we can prepare the clamped Gibbs states e^{-H_v}/Z_v within trace distance ϵ_H using one query to $F_H(\epsilon_H)$. Thus in both cases the sample variance of each coordinate of the gradient vector has the same upper bound.

We can plug these results back into (10) and bound the error

$$\mathbb{E}\left(\|G - G_{\mathrm{true}}\|_2^2\right) \leqslant M\left(\frac{1}{n} + \epsilon_H^2\right). \qquad (14)$$

Thus if we wish to take the overall variance to be ϵ^2 it suffices to take $n = M/(\epsilon^2 - M\epsilon_H^2)$. This also places a bound on the precision of gradient estimation in terms of precision of the density matrix preparation as $\epsilon > \sqrt{M}\epsilon_H$. ■

While the above analysis provides an asymptotic upper bound on the scaling of the number of state preparations needed to estimate the components of the gradient within constant error with respect to the max-norm. This gives the complexity of performing one epoch of the training process. However, we do not provide an estimate of the number of epochs required for the algorithm to converge. This number is unknown and depends sensitively on the training data, as we show in previous Appendixes, as well as the learning rate. Further work will be needed to provide good empirical, and theoretical bounds, on the number of training epochs that are needed in practice to train the Boltzmann machine within constant error.

While the above result gives an estimate of the query complexity of the algorithm, in order to assess the practicality of the algorithm the cost of the Gibbs state oracle $F_H(\epsilon_H)$ needs to be discussed. Since the exact preparation is NP-hard, one needs to rely on approximations such as contrastive divergence [10] in the classical case. There are several proposals for approximating a Gibbs state of a local Hamiltonian notably in Refs. [17,18]. However, neither of these methods can *a priori* be guaranteed to be efficient without making promises about either the fidelity of the thermal state with an efficiently preparable ansatz or without making assumptions about the spectral gap of a Markov process that operates on the qubits. We discuss properties of these methods in the Appendix.

Alternatively, one can achieve an approximation of the thermal state with a quantum annealer [11,15]. While it is difficult to argue about how close the Gibbs state output by such an annealer is to the true thermal state, such approaches are significant because they allow our training algorithms to be executed on present-day hardware.

Even if the gradient is efficiently computable the number of training epochs required to learn H may be large. It is difficult to bound the number of epochs, however, we provide

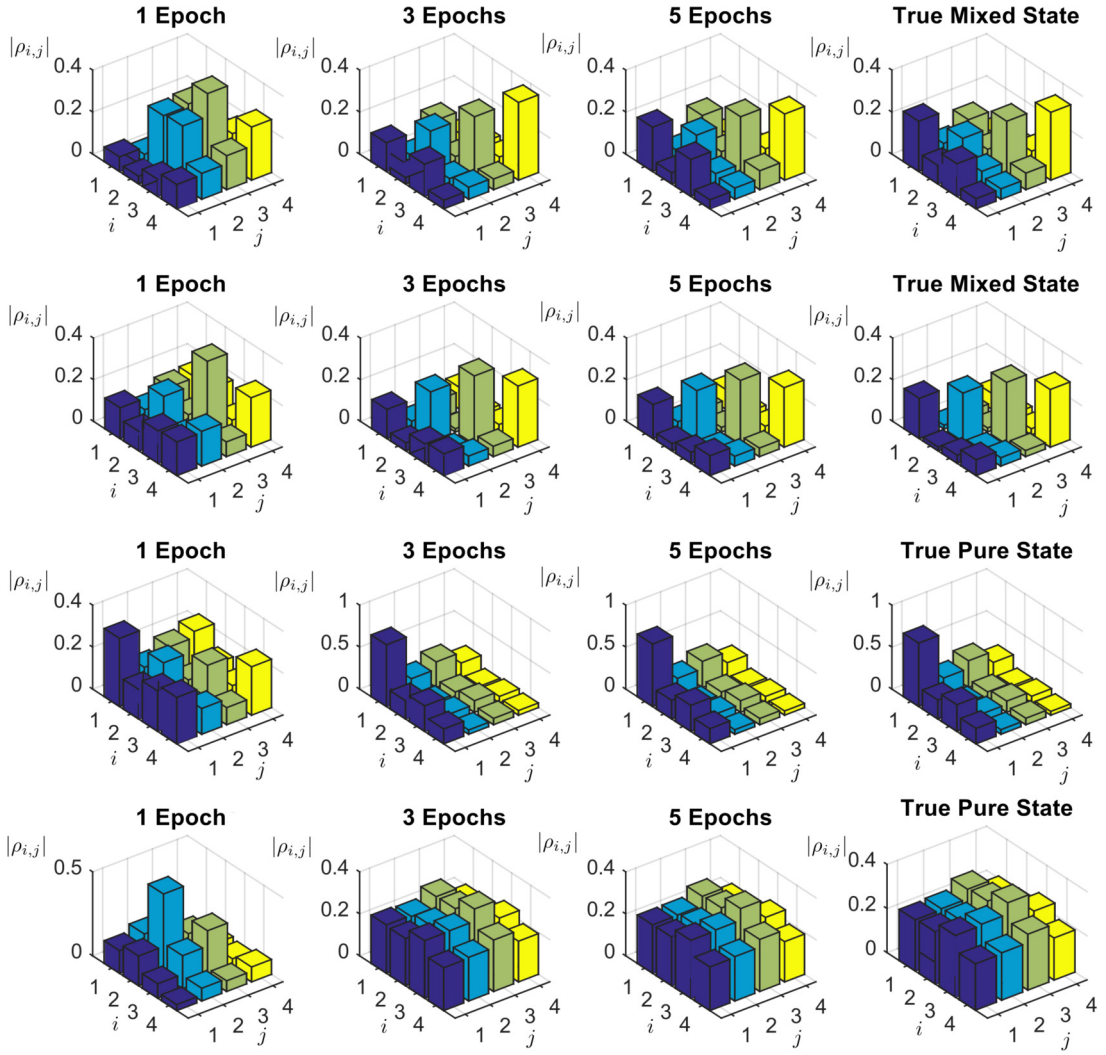


FIG. 1. Absolute values of tomographic reconstructions of two-qubit Haar-random pure states and two-qubit mixed states formed out of a uniform mixture of four Haar-random pure state operators using relative entropy training with $\eta = 1$.

below two lemmas that state the limitations of both relative entropy training and POVM training by respectively reducing tomography and Grover's search to them.

These results show that in general we cannot expect the number of training epochs to be polynomially large in generality. However, we have no reason to suspect that either problem is characteristic of the complexity of typical machine learning tasks on a quantum computer. We provide numerical evidence for this conjecture below.

VI. NUMERICAL RESULTS

We demonstrate the ability of quantum Boltzmann training to learn ensembles of two-qubit states that are either Haar-random pure states or mixed states that are convex combinations of columns vectors of Haar-random unitary matrices with uniformly distributed weights. For simplicity, we choose our Hamiltonian to consist of every two-qubit Pauli operator. Since this set is complete, every possible state can be generated using an appropriate Hamiltonian. We provide data to this effect in Fig. 1, wherein as few as five training epochs suffice to learn these states within graphical accuracy. We provide further details of the error versus epoch tradeoff in the Appendix. We next examine the performance of our algorithm for generative training using a small number of visible and hidden units and compare the result to classical training. Since we can only simulate small quantum computers classically, we choose a simple training set composed of step functions where the step occurs at each possible value with 5% noise added to each component of the vectors. For Golden-Thompson training we use a fermionic Hamiltonian from [19] plus a particle nonconservative term (see Appendix). This introduction of Fermionic operators makes the Hamiltonian nonstoquastic and thus hard to simulate using classical methods. Its detailed form can be found in the Appendix. Taking $\Lambda_1 = |\psi\rangle\langle\psi|$, $\Lambda_0 = \mathbb{1} - |\psi\rangle\langle\psi|$ for POVM training where $|\psi\rangle$ is a pure state constructed in the above fashion.

The data in Fig. 2 shows that the quantum model consistently outperforms the classical model in terms of accuracy. We observe that increasing the number of hidden units gives the classical methods a substantial advantage, but we do not notice that adding hidden units substantially improves $\mathcal{O}_\Lambda$ here. This

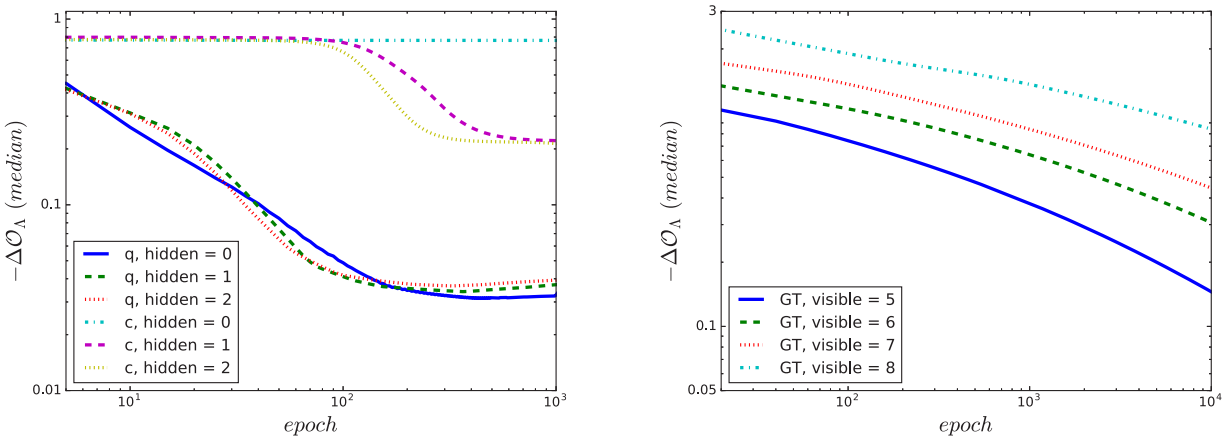


FIG. 2. Simulation of QBM with POVM training. We compare the difference between the optimal objective function and the computed one. We are able to compute the objective function because of the small size of the QBM. (a) and (b) show $\Delta\mathcal{O}_\Lambda := \mathcal{O}_{\Lambda,\max} - \mathcal{O}_\Lambda$ for (a) five visible units and varying numbers of hidden units and (b) for all relative entropy training with all visible Boltzmann machines. We take $\lambda = 0$ for all data considered and max $\mathcal{O}_{\Lambda,\max}$ is the maximum value of the training objective function attainable for the training data. In (a) we clearly see an improvement for the QBM (q) compared to its classical counterpart (c). (b) depicts the performance of QBMs with no hidden units and varied number of visible units.

is likely because the training data is sufficiently simple that the fermionic Hamiltonian gives enough freedoms to fit the data without the need for hidden units.

This data, along with further data in the Appendix, suggests that fermionic QBMs may be superior models for data; however, further study is needed to ensure that these models do not overfit the data.

VII. BQP-HARDNESS

It is straightforward to show that if our Boltzmann machine training algorithm were efficiently simulatable then classical computers could solve a BQP-hard decision problem, meaning that if a classical computer could simulate the protocol efficiently then classical computers would have to be at least as powerful as quantum computers. This strongly suggests classical intractability of our algorithm. To see this let $\{H_j\}$ consist only of the Pauli-Z operator acting on the zeroth qubit. Let U be an efficient quantum circuit that solves a decision problem with probability at least $2/3$ and encodes the answer in the zeroth qubit. Finally, let $\rho = e^{-\beta(\mathbb{1}-2U|0\rangle\langle 0|U^\dagger)}/\mathrm{Tr}(e^{-\beta(\mathbb{1}-U|0\rangle\langle 0|U^\dagger)})$. We then have

$$\begin{aligned} F(\rho, U|0\rangle\langle 0|U^\dagger) &= \mathrm{Tr}(\sqrt{\rho}[U|0\rangle\langle 0|U^\dagger]\sqrt{\rho}) \\ &= \mathrm{Tr}(e^{-\beta(\mathbb{1}-|0\rangle\langle 0|)}|0\rangle\langle 0|)/\mathrm{Tr}(e^{-\beta(\mathbb{1}-|0\rangle\langle 0|)}). \\ &= \frac{1}{(2^n-1)e^{-\beta}+1}, \end{aligned} \tag{15}$$

which is at least $\sqrt{1-\epsilon_H^2}$ if

$$\beta \geqslant n\ln(2) + \ln\left(\frac{1-\epsilon_H^2}{\left(\sqrt{1-\epsilon_H^2}-1\right)^2}\right).$$

Since U is by construction an efficient circuit, it is clear that $U|0\rangle$ provides an efficient ϵ_H approximation to ρ. The computation of $\mathrm{Tr}(\rho H_j)$ in this case corresponds to preparation of the state $U|0\rangle$ and then measuring the zeroth qubit in the computational basis within a prescribed error. Thus, if we could perform Boltzmann training efficiently for every $\{H_j\}$ and ρ we could also approximate any efficient quantum computation within bounded error probability. We thereby conclude that quantum Boltzmann machine training cannot be simulated classically, for arbitrary H_j, unless BPP = BQP. Thus quantum Boltzmann training offers the potential for exponential speedups relative to classical machine learning methods.

VIII. CONCLUSION

We proposed an approach to training QBM and eliminate the drawbacks presented by previous schemes. In particular, we see that we can learn a full Hamiltonian through either our POVM-based Golden-Thompson training approach or by training according to the relative entropy. The latter approach enables a form of partial tomography, which allows learning of Hamiltonian models for complex quantum states that cannot be probed using conventional tomographic approaches.

While our work demonstrates the viability of quantum Boltzmann training for broad classes of nonstoquastic Hamiltonians, subsequent work will be needed to establish whether it provides more generalized classical data than classical Boltzmann machines do. This will be necessary to understand the extent to which quantum models are prone to overfit the data.

Since we require only approximations of Gibbs states and computation of expectation values, these algorithms are near ideally suited for near future experiments. Furthermore, unlike many other proposals for machine learning that rely on QRAM, this approach can potentially offer exponential advantages and can be run on important problems using neural networks that are composed of fewer than 1000 qubits [20].

Perhaps the most exciting avenue of future work is the strong link between quantum state estimation and quantum neural network training. We hope that combining ideas from

quantum machine learning, quantum Hamiltonian learning, and state estimation will lead to even more powerful and efficient methods.

ACKNOWLEDGMENTS

We would like to thank K. M. Svore, A. Kapoor, F. Brandao, and V. Kliuchnikov for useful comments and feedback.

APPENDIX A: PREPARING THERMAL STATES

An essential part of Boltzmann machine training is sampling from the thermal distribution. Sadly, preparing the thermal state is NP-hard. Classical algorithms circumvent this problem by approximating it using contrastive divergence [10]. Analogous quantum solutions have been proposed in Refs. [14,18,21]. A high-precision approximation can be obtained using the methods from Ref. [17].

The method of Chowdhury and Somma is strongly related to the methods in Refs. [3,18,21]. The main difference between these methods is that their approach uses an integral transformation to allow the exponential to be approximated as a linear combination of unitaries. These operators are then simulated using Hamiltonian simulation ideas as well as ideas from simulating fractional queries. The complexity of preparing a Gibbs state $\rho \in \mathbb{C}^{N\times N}$ within error ϵ, as measured by the 2-norm, is from [17]

$$O\left[\sqrt{\frac{N}{Z}}\,\mathrm{polylog}\left(\frac{1}{\epsilon}\sqrt{\frac{N}{Z}}\right)\right], \tag{A1}$$

for inverse temperature $\beta = 1$ and cases where H is explicitly represented as a linear combination of Pauli operators. This is roughly quadratically better than existing approaches for preparing general Gibbs states if constant ϵ is required, but constitutes an exponential improvement if $1/\epsilon$ is large. This approach is further efficient if $Z \in \Theta(N/\mathrm{polylog}(N))$. This is expected if roughly a constant fraction of all eigenstates have a meaningful impact on the partition function. While this may hold in some cases, particularly in cases with strong regularization [3,14], it is not expected to hold generically.

An alternative method for preparing thermal states is proposed by Yung and Aspuru-Guzik. The approach works by using a Szegedy walk operator whose transition amplitudes are given by the Metropolis rule based on the energy eigenvalue difference between the two states. These eigenvalues are computed using phase estimation. A coherent analog of the Gibbs state is found by using phase estimation on the walk operator, W, which follows these transition rules. The number of applications of controlled W required in the outer phase estimation loop is

$$O\left[\frac{\|H\|^2}{\epsilon\sqrt{\delta}}\ln\left(\frac{\|H\|^2}{\epsilon^2}\right)\right], \tag{A2}$$

where δ is the gap of the transition matrix that defines the quantum walk, ϵ is the error in the preparation of the thermal state. Since each application of the walk operator requires estimation of the eigenvalues of H, this complexity is further multiplied by the complexity of the quantum simulation. Provided that the Hamiltonian is a sum of at most m one-sparse Hamiltonians with efficiently computable coefficients then the cost is multiplied by a factor of $m\ln(m)/\ln\ln(m)$ to $m^{2+o(1)}$ depending on the quantum simulation algorithm used within the phase estimation procedure.

These features imply that it is not clear *a priori* which algorithm is preferable to use for preparing thermal states. For cases where the partition function is expected to be large or highly accurate thermal states are required, Eq. (A1) is preferable. If the spectral gap of the transition matrix is small, quantum simulation is inexpensive for H and low precision is required then Eq. (A2) will be preferable.

APPENDIX B: RELATIVE ENTROPY TRAINING

In this Appendix, we provide further numerical experiments that probe the performance of quantum relative entropy training. The first that we consider is in Fig. 3, which shows the performance of this form of training for learning randomly chosen two-qubit pure and mixed states. In particular, we choose the pure states uniformly with respect to the Haar measure and pick the mixed states by generating the eigenvectors of Haar-random unitaries and choosing our mixed states to be convex combinations of such states with weights that are uniformly distributed.

We see from these experiments that the median performance of relative entropy training on mixed states is quite good. The quantum relative entropy is observed to shrink exponentially with the number of training epochs. After as few as 35 training epochs with $\eta = 1$, the error is limited by numerical precision. However, a glance at the 95% confidence interval in this figure reveals that many of the examples yield much larger errors than these. Specifically after 60 epochs with the same learning rate the 97.5th percentile of the data in Fig. 3 only has a relative entropy of 10^{-5} and is decaying much slower than the median.

The origin of this problem can be seen from the plot of the relative entropy for pure states in Fig. 3. Pure states are observed to require many more training epochs to achieve the same accuracy as highly mixed states. This is expected because pure states are only possible in the limit as $\|H\| \to \infty$. The need to have large weights in the Hamiltonian not only means that more epochs will be needed to allow the weights to reach the magnitudes needed to approximate a pure state, but it also means that the training landscape is expected to be much more rough as we approach this limit. This is what makes learning such pure states difficult. Similarly, the fat tails of the error distribution for the mixed state case makes sense given that some of the data will come from nearly pure states.

The narrowing of error bars in these plots can be understood, approximately, from Levy's lemma. Levy's lemma states that for any Lipschitz continuous function mapping the unit sphere in $2N-1$ dimensions (on which the pure states in $\mathbb{C}^N$ can be embedded) the probability that $f(x)$ deviates from its Haar expectation value by ϵ is in $e^{-O(N\epsilon^2)}$. Thus if we take $f(x) = \langle x|\sigma|x\rangle$, as we increase N we expect almost all initial states x chosen uniformly at random according to the Haar measure to have the widths of their confidence intervals in $O(1/\sqrt{N}) \subseteq O(2^{-n/2})$, where n is the number of qubits. This means that the we expect the width of the confidence intervals to shrink exponentially with the number of qubits for cases where the target state is pure. We do not necessarily expect

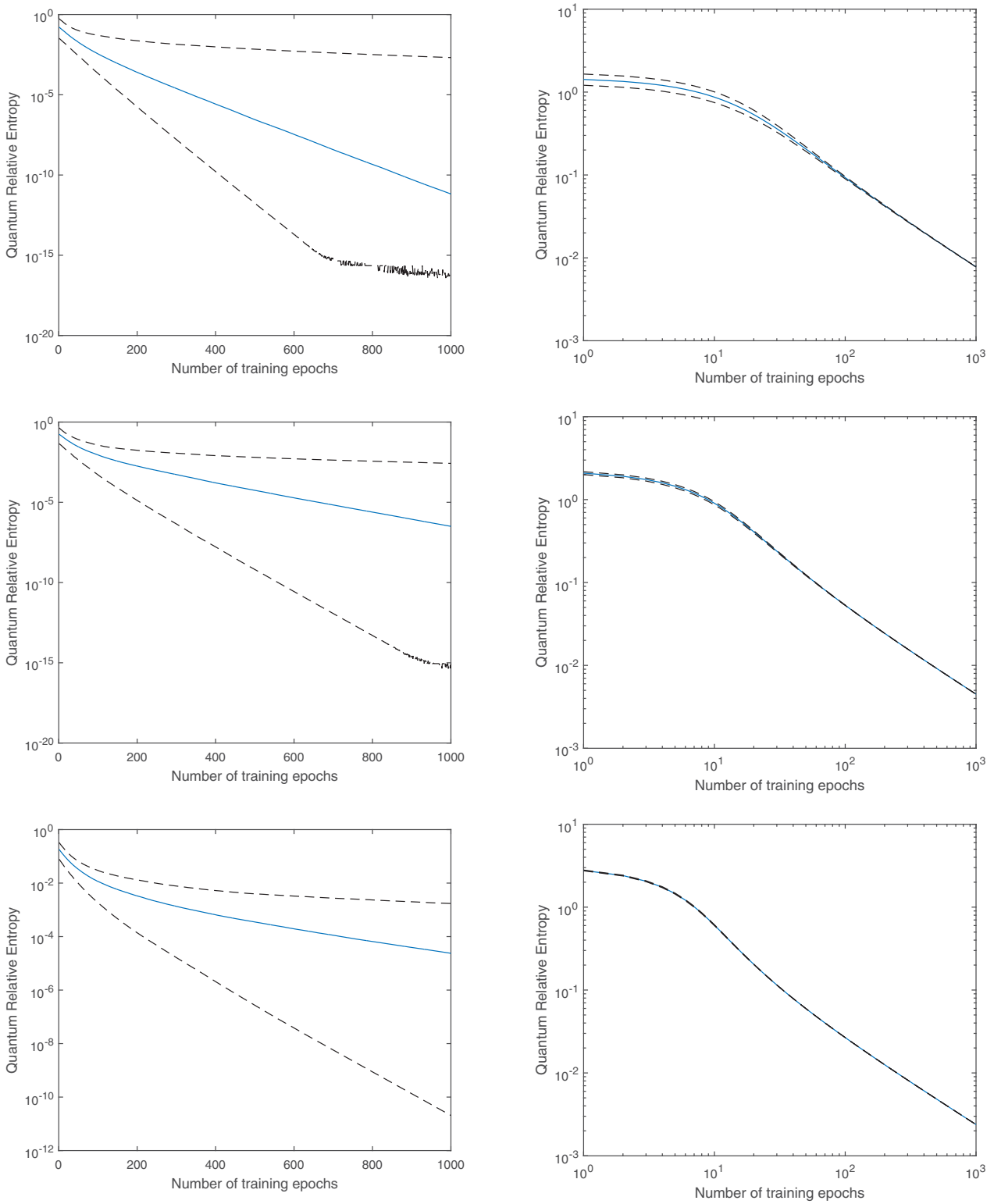


FIG. 3. Distribution of quantum relative entropies between randomly chosen mixed (left) and pure (right) states as a function of the number of training epochs for two- (top), - (middle), and -qubit (bottom) tomography with $\eta = 0.025$. Dashed lines represent a 90% confidence interval and the solid line denotes the median.

similar concentrations to hold for mixed states because Levy's lemma does not directly apply in such cases.

When we consider relative entropy training, we note that the value of the objective function seems to systematically grow with the size of the Boltzmann machine. This is expected because the complexity of the training data grows as we increase the number of visible units. We see that qualitatively that training continues to improve the value of the objective

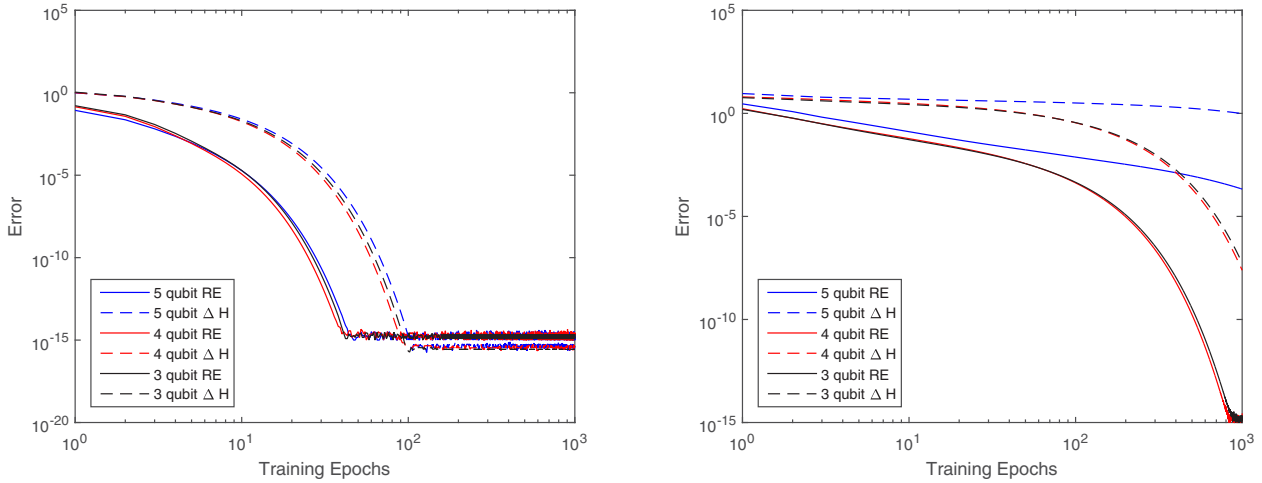


FIG. 4. Relative entropies and Hamiltonian errors for learning transverse Ising models. The left figure shows data for a TI Hamiltonian with Gaussian random terms that is rescaled to unit norm. The right figure shows the analogous situation but without normalizing the Hamiltonian. Here $\Delta H = \|H_{\text{true}} - H_{\text{est}}\|_2$.

function here and given the computational resources at our disposal, we were unable to see the learning stop despite training with the relative entropy objective rather than the reported objective function $\mathcal{O}_\Lambda$.

APPENDIX C: APPLICATIONS TO HAMILTONIAN LEARNING

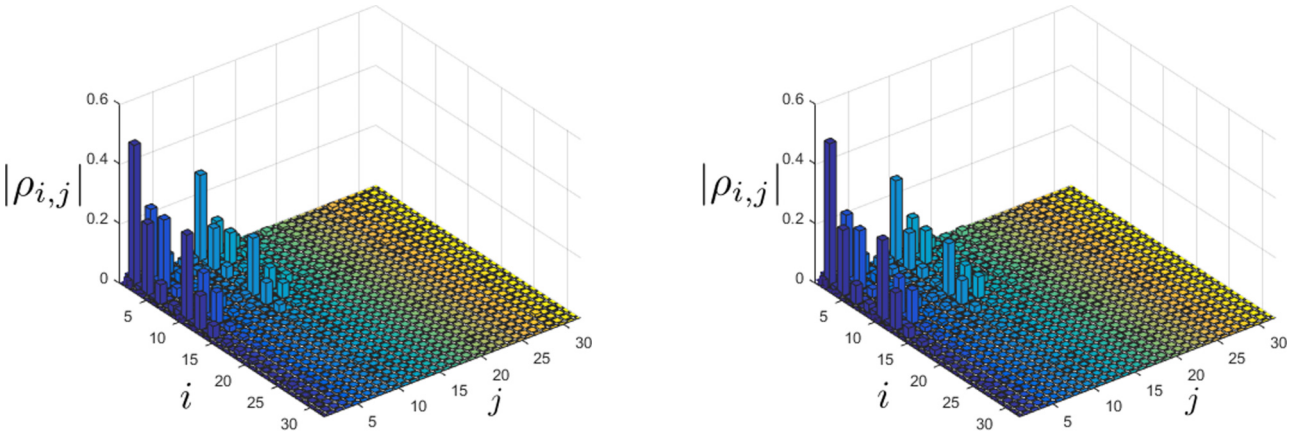
In all of the above applications our aim is to learn a Hamiltonian that parameterizes a thermal state model for the training data. However, in some cases our aim may not be to learn a particular input state but to learn a system Hamiltonian for a thermalizing system. Relative entropy training then allows such a Hamiltonian to be learned from the thermal expectation values of the Hamiltonian terms via gradient ascent and a simulator. Here we illustrate this by moving away from a Hamiltonian model that is composed of a complete set of Pauli operators, to a local Hamiltonian model that lacks many of these terms. Specifically, we choose a transverse Ising model on the complete graph:

$$H = \sum_j \alpha_j Z^j + \sum_j \beta_j X^j + \sum_{<i,j>} \gamma_{i,j} Z^i Z^j. \qquad \text{(C1)}$$

We then test the ability of our training algorithm to reconstruct the true Hamiltonian given access to the requisite expectation values.

Apart from the simplicity of the transverse Ising model, it is also a useful example because in many cases these models can be simulated efficiently using quantum Monte Carlo methods. This means that quantum computers are not necessary for estimating gradients of models for large quantum systems.

Figure 4 shows that the ability to learn such models depends strongly on the norm of the Hamiltonian, or equivalently the inverse temperature of the thermal state. It is much easier for us to learn a model using this method for a high-temperature



(a) Thermal state for transverse Ising Model (b) Mean-field approximation

FIG. 5. Absolute value of mean-field approximations to the thermal state of a five-qubit random TI Hamiltonian where each Hamiltonian term was chosen by sampling from a Gaussian with zero mean and unit variance at $\beta = 1$. The learning rate was taken to be $\eta = 1$ and 100 training epochs were used. (a) Thermal state for transverse Ising model. (b) Mean-field approximation.

state than a low-temperature thermal state. The reason for this is similar to what we observed previously. Gradient ascent takes many steps before it can get within the vicinity of the correct thermal state. This is especially clear when we note that the error changes only modestly as we vary the number of qubits, however, it changes dramatically when we vary the norm of the Hamiltonian. This means that it takes many more training epochs to reach the region where the errors shrink exponentially from the initially chosen random Hamiltonian. In cases where a good ansatz for the Hamiltonian is known, this process could be sped up.

Mean-field approximations

Mean-field approximations are ubiquitous in condensed matter physics. They are relatively simple to compute for some quantum systems such as Ising models, but can be challenging for fully quantum models. Here we provide a method to find a mean-field Hamiltonian for a system given the ability to compute moments of the density operator ρ. The approach exactly follows the previous discussion, except rather than taking Eq. (C1) we use

$$H = \sum_j H_j,$$
$$H_j := \alpha_j Z^j + \beta_j X^j + \gamma_j Y^j. \tag{C2}$$

Our aim is then to find vectors, α, β and γ such that the correlated state ρ is approximated by the uncorrelated mean-field state:

$$\rho \approx e^{-H}/Z = \left[\prod_j e^{-H_j}\right]/Z. \tag{C3}$$

We see from the data in Fig. 5 that relative entropy training on a thermal state that arises from a five-qubit transverse-Ising Hamiltonian on a complete graph for 100 training epochs yields a mean-field approximation that graphically is very close to the original state. In fact if ρ is the TI thermal state and σ is the mean-field approximation to it then $\mathrm{Tr}(\rho\sigma) \approx 0.71$. This shows that our method is a practical way to compute a mean-field approximation.

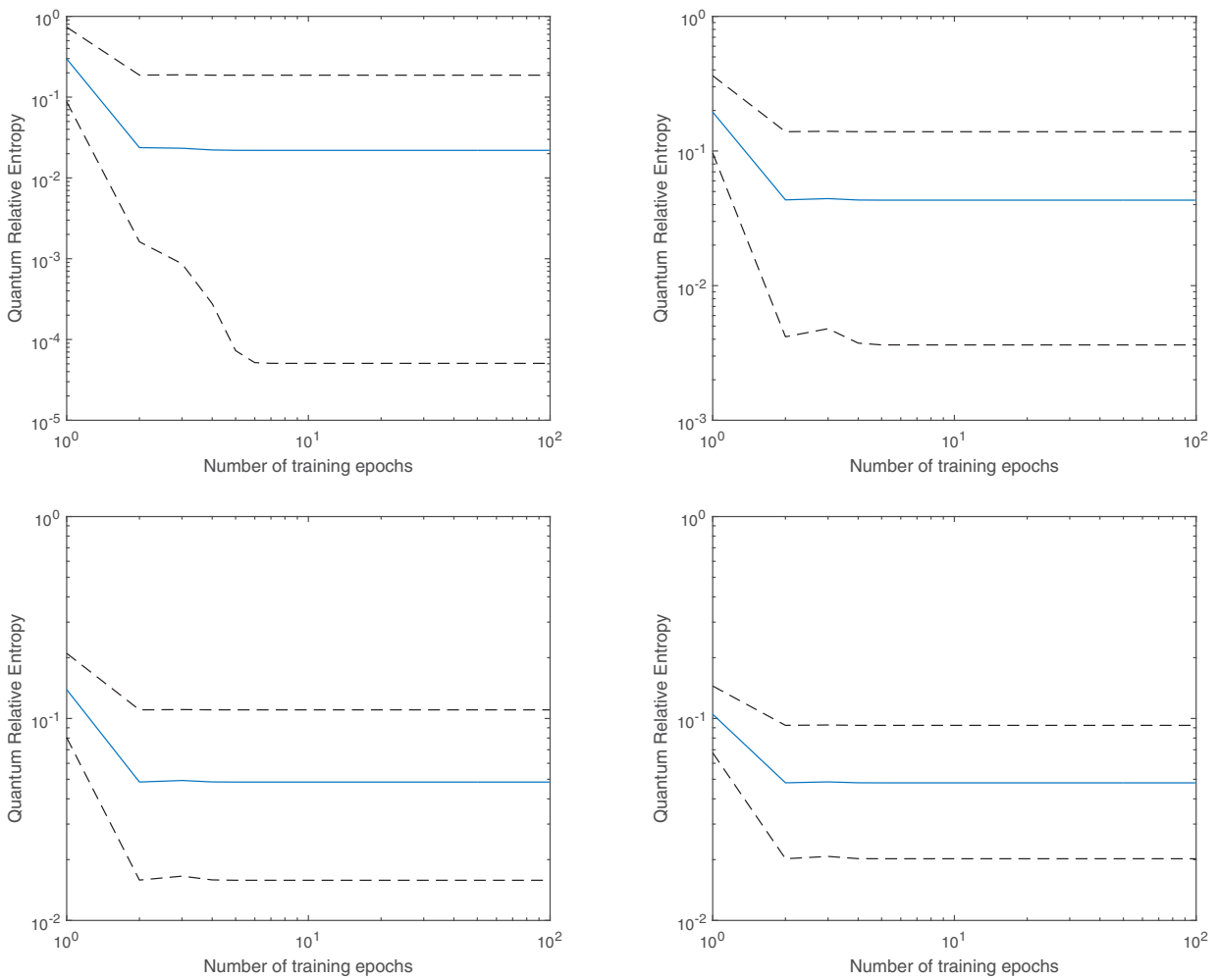


FIG. 6. Median relative entropies for mean-field and true distributions for thermal states generated by transverse Ising models on the complete graph with Gaussian random coefficients chosen with zero mean and unit variance for two (top left), three (top right), four (bottom left), and five (bottom right) qubits and $\eta = 1$ was taken for each datum. Dashed lines give a 95% confidence interval.

In order to assess how many epochs it takes in order to converge to a good mean-field approximation, we see in Fig. 6 that after only a single training epoch, the median relative entropy, over 1000 randomly chosen instances, approximately reaches its optimal value. Furthermore, we note that the relative entropy at which the system saturates tends to rise with the number of qubits. This is in part due to the fact that the Hamiltonian is on the complete graph and the weights are chosen according to a Gaussian distribution. We therefore expect more correlated Hamiltonians as the number of qubits grows and in turn expect the mean-field approximation to be worse, which matches our observations.

If we turn our attention to learning mean-field approximations to n-local Hamiltonians for $n = 2, \ldots, 5$ we note that the mean-field approximation fails both qualitatively and quantitatively to capture the correlations in the true distribution. This is not surprising because such states are expected to be highly correlated and mean-field approximations should fail to describe them well. These discrepancies continue even when we reduce the norm of the Hamiltonian. This illustrates that the ability to find high-fidelity mean-field approximations depends less on the number of qubits than the properties of the underlying Hamiltonian.

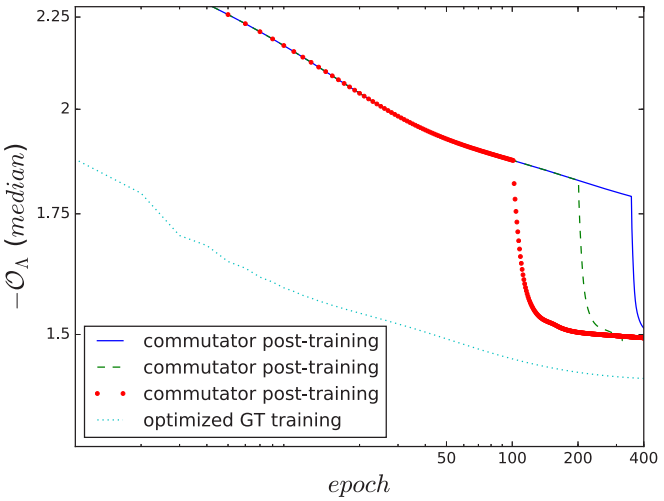


FIG. 7. Plot showing the efficacy of commutator training for all-visible Boltzmann machines with four visible units. The top lines depict training with Golden-Thompson at first and then switching to commutator training where we see a sudden increase in accuracy. We picked the parameters such that the commutator training is stable. The bottom line (dotted) shows the performance of Golden-Thompson training with optimized learning rate and momentum.

APPENDIX D: COMMUTATOR TRAINING

Commutator training. A second approach to POVM training avoids the use of the Golden-Thompson inequality. The idea behind this approach is to approximate the series in the derivative of (5) as a commutator series using Hadamard's lemma. The derivative of exponential can be expressed using the Duhamel's formula

$$\mathrm{Tr}[\Lambda_v \partial_\theta e^{-H}] = \mathrm{Tr}\left[\int_0^1 \Lambda_v e^{sH}[\partial_\theta H] e^{(1-s)H} ds\right]. \tag{D1}$$

If Λ_v commuted with H, then we would recover an expression for the gradient that strongly resembles the classical case. In general, the expectation value can be written as a commutator series. In particular, if the Hamiltonian is a sum of bounded Hamiltonian terms then we have $\mathrm{Tr}[Ce^{-H}]$ for

$$C := \Lambda_v\left(\partial_\theta H + \frac{[H,\partial_\theta H]}{2!} + \frac{[H,[H,\partial_\theta H]]}{3!} + \cdots\right). \tag{D2}$$

Thus the gradient of the average log-likelihood becomes

$$\sum_{\mathbf{v}} P_{\mathbf{v}}\left(-\frac{\mathrm{Tr}[e^{-H}C]}{\mathrm{Tr}[e^{-H}]} + \frac{\mathrm{Tr}[e^{-H}\partial_\theta H]}{\mathrm{Tr}[e^{-H}]}\right) - \lambda h_\theta \delta_{H_\theta \in H_Q}. \tag{D3}$$

This commutator series can be made tractable by truncating it at low order, which will not incur substantial error if $\|[H,\partial_\theta H]\| \ll 1$. Commutator training is therefore expected to outperform Golden-Thompson training in the presence of L_2 regularization on the quantum terms, but is not as broadly applicable.

We see in Fig. 7 that for a fixed learning rate that the gradients returned from a Golden-Thompson expansion are inferior to those returned from a high-order commutator expansion. This in turn illustrates the gap between the exact gradients and Golden-Thompson gradients. We examine this by performing Golden-Thompson trainings for an all-visible Boltzmann machine with four visible units. We train for a fixed number of epochs using the Golden-Thompson gradients and then switch to a fifth-order commutator expansion. We see a dramatic improvement in the objective function as a result. This shows that in some circumstances much better gradients can be found with the commutator method than with Golden-Thompson; albeit at a higher price due to the fact that more expectation values need to be measured.

A drawback of the commutator method is that we find in numerical experiments that it is much less stable than Golden-Thompson. In particular, commutator training does not fail gracefully when the expansion does not converge or when the learning rate is too large. This means that the optimal learning rate for this form of training can substantially differ from the optimal learning rate for Golden-Thompson training. When we optimize the learning rate for Golden-Thompson training we find that the training objective function increases by a factor of roughly 1.5, falling in line with the results seen using commutator training. This shows that while commutator training can give more accurate gradients, it does not necessarily require fewer gradient steps. In practice, the method is likely to be used in the last few training epochs after Golden-Thompson training, or other forms of approximate training to reach a local optima.

APPENDIX E: ADDITIONAL EXPERIMENTS FOR GENERATIVE TRAINING

While the numerics in the main body provided a glimpse of the ability of Golden-Thompson and relative entropy training to learn general Hamiltonian models, we provide a few additional experiments here to look at the performance of the training algorithm for different sizes of fermionic Boltzmann

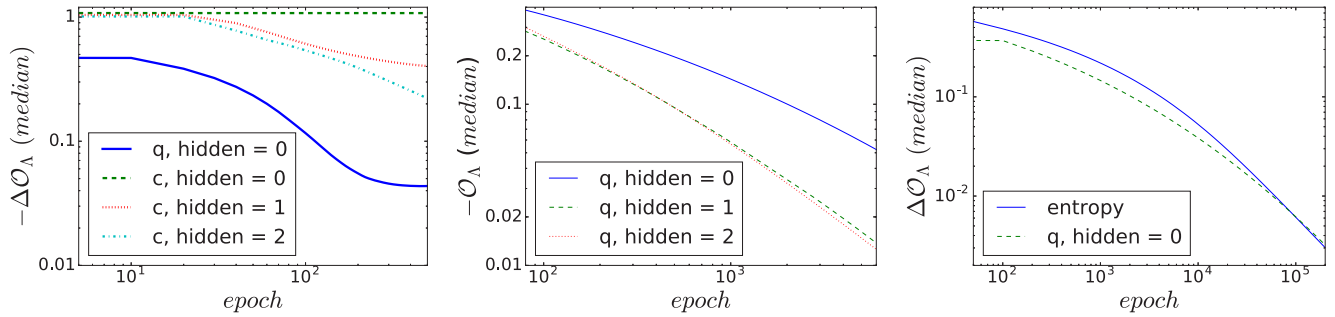


FIG. 8. (a) shows the value of the Golden-Thompson objective function in terms of $\mathcal{O}_\rho$ for all-visible quantum Boltzmann machines with Golden-Thompson POVM training and other parameters optimized for performance. In (b), we compare Boltzmann machines with four visible units and different number of hidden units using POVM based training. (c) compares the convergence of training for relative entropy and Golden-Thompson approaches.

machines. The Hamiltonian we consider is of the form

$$H = H_p + \tfrac{1}{2}H_{pq} + \tfrac{1}{2}H_{pqrs}, \tag{E1}$$

where

$$H_p = \sum_p h_p(a_p + a_p^\dagger), \tag{E2}$$

$$H_{pq} = \sum_{pq} h_{pq}(a_p^\dagger a_q + a_q^\dagger a_p), \tag{E3}$$

$$H_{pqrs} = \sum_{pqrs} h_{pqrs}(a_p^\dagger a_q^\dagger a_r a_s + \text{H.c.}). \tag{E4}$$

Here a_p and $a_p^\dagger$ are Fermionic creation and annihilation operators, which create and destroy Fermions at unit p. They have the properties that $a^\dagger|0\rangle = |1\rangle$, $a^\dagger|1\rangle = 0$ and $a_p^\dagger a_q + a_q a_p^\dagger = \mathbb{1}\delta_{pq}$. The Hamiltonian here corresponds to the standard Hamiltonian used in quantum chemistry modulo the presence of the nonparticle conserving H_p term.

We first examine the performance of the algorithm as a function of the number of hidden units for a six visible unit example in Fig. 8. We note here that while we can increase the number of hidden units in the classical model to help improve the objective function,

We see from Fig. 8 that the inclusion of hidden units can have a dramatic improvement on the classical model's ability to learn. In the quantum case we see that even the all-visible model outperforms each of the classical cases considered. Adding a single hidden unit does substantially help for a four visible unit model in the quantum case, but additional hidden units do not provide the quantum Boltzmann machine with much greater power for this training set. This vindicates that the idea of deep learning still has a role for these quantum models despite the fact that the POVM is a projector onto a pure state and its compliment. However, the lack of systematic improvements observed for larger instances suggest that the correlations present in the training data can be easily represented using the H_{pqrs} terms present in the fermionic Hamiltonian, since the impact of such terms is greatly diminished in the four visible unit case. More work will be needed in order to systematically study the role that hidden units play in deep learning for fermionic Boltzmann machines and related models.

Lastly, we train the fermionic Boltzmann machine with relative entropy training. Figure 8 shows that relative entropy and Golden-Thompson perform similarly.

[1] P. Rebentrost, M. Mohseni, and S. Lloyd, Quantum Support Vector Machine for Big Data Classification, Phys. Rev. Lett. **113**, 130503 (2014).
[2] E. Aïmeur, G. Brassard, and S. Gambs, *Advances in Artificial Intelligence* (Springer, Berlin, 2006), pp. 431–442.
[3] N. Wiebe, A. Kapoor, and K. M Svore, Quantum deep learning, Quantum Inf. Comput. **16**, 0541 (2016).
[4] K. L. Pudenz and D. A. Lidar, Quantum adiabatic machine learning, Quantum Inf. Process. **12**, 2027 (2013).
[5] H. Neven, V. S. Denchev, G. Rose, and W. G. Macready, arXiv:0811.0416.
[6] P. Rebentrost, M. Schuld, F. Petruccione, and S. Lloyd, arXiv:1612.01789.
[7] N. Wiebe, A. Kapoor, and K. M. Svore, arXiv:1602.04799.
[8] I. Kerenidis and A. Prakash, arXiv:1603.08675.
[9] A. Monràs, G. Sentís, and P. Wittek, Inductive Supervised Quantum Learning, Phys. Rev. Lett. **118**, 190503 (2017).
[10] G. E. Hinton, Training products of experts by minimizing contrastive divergence, Neural Comput. **14**, 1771 (2002).
[11] M. Denil and N. De Freitas, Toward the implementation of a quantum RBM, In NIPS Deep Learning and Unsupervised Feature Learning Workshop, Vol. 5, 2011.
[12] S. H Adachi and M. P Henderson, arXiv:1510.06356.
[13] M. Benedetti, J. Realpe-Gómez, R. Biswas, and A. Perdomo-Ortiz, Phys. Rev. A **94**, 022308 (2016).
[14] N. Wiebe, A. Kapoor, C. Granade, and K. M. Svore, arXiv:1507.02642.

[15] M. H. Amin, E. Andriyash, J. Rolfe, B. Kulchytskyy, and R. Melko, arXiv:1601.02036.

[16] J. Haah, A. W. Harrow, Zhengfeng Ji, Xiaodi Wu, and Nengkun Yu, arXiv:1508.01797.

[17] A. N. Chowdhury and R. D. Somma, arXiv:1603.02940.

[18] Man-Hong Yung and A. Aspuru-Guzik, A quantum-quantum Metropolis algorithm, Proc. Natl. Acad. Sci. **109**, 754 (2012).

[19] J. D. Whitfield, J. Biamonte, and A. Aspuru-Guzik, Simulation of electronic structure Hamiltonians using quantum computers, Mol. Phys. **109**, 735 (2011).

[20] Yann LeCun, The mnist database of handwritten digits, http://yann.lecun.com/exdb/mnist/, 1998.

[21] D. Poulin and P. Wocjan, Sampling from the Thermal Quantum Gibbs State and Evaluating Partition Functions with a Quantum Computer, Phys. Rev. Lett. **103**, 220502 (2009).