1-1-2018

Unsupervised machine learning account of magnetic transitions in the Hubbard model

Kelvin Ch'ng
San Jose State University

Nick Vazquez
San Jose State University

Ehsan Khatami
San Jose State University, ehsan.khatami@sjsu.edu

Recommended Citation

Kelvin Ch'ng, Nick Vazquez, and Ehsan Khatami. "Unsupervised machine learning account of magnetic transitions in the Hubbard model" *Physical Review E* (2018). https://doi.org/10.1103/PhysRevE.97.013306

Unsupervised machine learning account of magnetic transitions in the Hubbard model

Kelvin Ch'ng, Nick Vazquez, and Ehsan Khatami[*]

Department of Physics and Astronomy, San José State University, San José, California 95192, USA

(Received 10 August 2017; published 16 January 2018)

We employ several unsupervised machine learning techniques, including autoencoders, random trees embedding, and t-distributed stochastic neighboring ensemble (t-SNE), to reduce the dimensionality of, and therefore classify, raw (auxiliary) spin configurations generated, through Monte Carlo simulations of small clusters, for the Ising and Fermi-Hubbard models at finite temperatures. Results from a convolutional autoencoder for the three-dimensional Ising model can be shown to produce the magnetization and the susceptibility as a function of temperature with a high degree of accuracy. Quantum fluctuations distort this picture and prevent us from making such connections between the output of the autoencoder and physical observables for the Hubbard model. However, we are able to define an indicator based on the output of the t-SNE algorithm that shows a near perfect agreement with the antiferromagnetic structure factor of the model in two and three spatial dimensions in the weak-coupling regime. t-SNE also predicts a transition to the canted antiferromagnetic phase for the three-dimensional model when a strong magnetic field is present. We show that these techniques cannot be expected to work away from half filling when the "sign problem" in quantum Monte Carlo simulations is present.

DOI: 10.1103/PhysRevE.97.013306

I. INTRODUCTION

Machine learning has emerged as an unconventional tool to gain insight into properties of many-body physics. In 2014, Louis-François *et al.* [1] used support vector machines, a type of supervised learning models, to obtain Green's function of the Anderson impurity model. Supervised machine learning techniques based on artificial neural networks were used later in a groundbreaking work to classify phases of models in statistical mechanics and condensed matter physics [2]. Shortly after, other groups expanded the application of these techniques to identify phase transitions, including to topological phases, in quantum many-body systems at zero or finite temperatures [3–7].

Parallel efforts demonstrated the power of restricted Boltzmann machines, simple artificial neural networks with one visible layer corresponding to the physical system and one hidden layer, in learning thermodynamics of Ising models [8], producing starting points for variational quantum Monte Carlo that are superior to those from conventional methods [9], performing tomography for many-body quantum states [10], and constructing topological states [11]. Interesting connections between artificial neural networks and more conventional methods in condensed matter physics have also been uncovered [12,13]

Unsupervised machine learning techniques, on the other hand, have so far been mostly used to classify phases of classical model in many-body physics. For example, t-distributed stochastic neighboring ensemble (t-SNE) technique [14–16] was used in Ref. [2] to cluster spin configurations and visualize the phase transition of the two-dimensional (2D) Ising model. Later, Lei Wang [17] applied principal component analysis (PCA) [18] to obtain low-dimensional representations of Ising spin configurations and make connections between principal components and physical observables such as the magnetization and the susceptibility, conventionally used to determine critical phenomena. His work was recently followed up by other groups who provided a more detailed examination of the PCA and other techniques applied to various classical models, including those on frustrated geometries [19–21]. PCA has also been applied to quantum systems [5]; however, the 2D visualization of the spin configuration for the random-field Heisenberg model did not produce any useful features. A proposal for a different type of unsupervised machine learning for quantum many-body systems, which combines two-point function calculations with convolutional neural networks, is also recently put forth [22].

Here we employ several nonlinear unsupervised machine learning methods, including fully connected and convolutional autoencoders [23–25], random trees embedding [26–28], and the t-SNE, to reduce the dimensionality of raw auxiliary spin (also known as auxiliary field) configurations generated during quantum Monte Carlo simulations of the two- and three-dimensional (3D) Fermi-Hubbard models [29]. We focus on the finite-temperature magnetic phase transitions of the model in three or the corresponding crossover in two spatial dimensions. Therefore, most of the data are generated at half filling, where there are on average one fermion per lattice site. We visualize the outcomes and look for features in the dimensionally-reduced configurations that may correlate with physical observables or signal phase transitions and crossovers. We work with unlabeled data during learning; we do not use any knowledge of the model parameters or temperature each configuration corresponds to in our analysis, nor do we use any information about the location of the phase transitions or crossovers during learning.

We start, however, with the classical Ising model on a 3D cubic lattice and benchmark the outcome of our convolutional autoencoder using spin configurations generated in a Monte Carlo simulation in a range of temperatures

[*] Corresponding author: ehsan.khatami@sjsu.edu

on both sides of the phase transition. We are able to define indicators that closely resemble the magnetization or the susceptibility. We then generalize the neural network to accommodate for the additional imaginary time axis in the auxiliary spin configurations of the Hubbard models and show that quantum fluctuations as well as the O(3) symmetry of the model in this case lead to low-dimensional visualizations that are fuzzier than their classical counterparts, although useful indicators signaling magnetic phase changes can still be defined. Next, we find that a *fully connected* autoencoder, combined with random trees embedding produces a more or less temperature-resolved image of the configurations in two dimensions.

t-SNE emerges as a clear winner among all the techniques, or combinations of techniques we have used, producing low-dimensional representations of data with clearly distinguishable patterns above and below the Néel temperature for the 3D model or the crossover temperature for the 2D model. We define temperature-dependent indicators and show that, in the weak-coupling regime, they correlate perfectly with the antiferromagnetic (AFM) structure factors, and can capture a transition in the presence of a magnetic field. Finally, we apply t-SNE to configurations generated for the 3D Hubbard model away from half filling in the presence of the "sign problem" [30,31] in quantum Monte Carlo simulations and discuss the risks of using dimensional-reduction techniques in sign-problematic regions.

In the following section, we briefly discuss the models we have considered in this study. Then, in Sec. III, we provide an overview of the various machine learning techniques we employ. The results are discussed in Sec. IV, followed by concluding remarks.

II. MODELS

A. 3D Ising model

We first consider the classical Ising model on the 3D cubic lattice,

$$H = -J \sum_{\langle ij \rangle} \sigma_i \sigma_j, \tag{1}$$

where $\sigma_i = \pm 1$, $\langle .. \rangle$ denotes nearest neighbors, and J is the strength of the corresponding exchange interaction (we set $J = 1$ as the unit of energy whenever the Ising model is discussed). The system undergoes a second-order phase transition as the temperature is lowered below the critical value of $4.5J$ [32].

We perform Monte Carlo simulations based on the Metropolis algorithm [33] on a $N = 8 \times 8 \times 8$ lattice and generate spin configurations at different temperatures. Each configuration is an array of size N with ±1 as elements, and can be thought of as a points in a N-dimensional space.

B. The Fermi-Hubbard model

We are mainly interested in how quantum fluctuations affect our ability to locate phase transitions or crossovers with unsupervised machine learning techniques. Therefore, we also consider strongly-correlated fermions in 3D cubic and 2D square lattices, described by the Hubbard model,

$$H = -t \sum_{\langle ij \rangle \sigma} c^{\dagger}_{i\sigma} c_{j\sigma} + U \sum_i \left(n_{i\uparrow} - \frac{1}{2} \right) \left(n_{i\downarrow} - \frac{1}{2} \right) - \mu \sum_{i\sigma} n_{i\sigma} - \frac{h}{2} \sum_i (n_{i\uparrow} - n_{i\downarrow}), \tag{2}$$

where $c^{\dagger}_{i\sigma}$ ($c_{i\sigma}$) creates (annihilates) a fermion with spin σ on site i, $n_{i\sigma} = c^{\dagger}_{i\sigma} c_{i\sigma}$ is the number operator, t is the nearest-neighbor hopping integral (we set $t = 1$ as the unit of energy whenever the Hubbard model is discussed), U is the on-site Coulomb interaction, μ is the chemical potential, and h is the magnitude of the magnetic field. By symmetry, $\mu = 0$ leads to half filling (average density of one fermion per site) and hole doping is achieved by decreasing μ.

For any $U > 0$, the 3D model displays a second-order transition from an unordered phase at high temperatures to a long-range Néel ordered phase below a U-dependent critical temperature T_N. Theoretical and numerical analysis [34–40] have shown that after an exponential increase from zero by turning on U in the weak-coupling regime [34], T_N peaks around $U = 9$ and eventually goes to zero as $1/U$ by further increasing the interaction strength in the strong-coupling regime. The latter can be understood from the Néel transition of an antiferromagnetic spin$-1/2$ Heisenberg model [41], which provides the low-energy description of the half-filled Hubbard model in the strong-coupling regime where double occupancy is largely suppressed and fermions interact predominantly through the spin exchange interaction $\mathcal{J} = \frac{4t^2}{U}$. Long-range antiferromagnetic correlations of the model have recently been observed in an experimental realization using ultracold fermionic atoms on optical lattices [42].

We use the determinantal quantum Monte Carlo (DQMC) [43,44] method to simulate the model on a $N = 4 \times 4 \times 4$ lattice for three different values of the interaction strength, $U = 4$, 9, and 14, in the weak-, intermediate-, and strong-coupling regimes, respectively. We generate and save auxiliary spin configurations that can be thought of as points in a $(N\mathcal{L})$-dimensional space, where $\mathcal{L} = 200$ is the number of imaginary time slices, for a range of temperatures (see Ref. [4] for details of our DQMC simulations).

We also generate auxiliary spin configurations for the model in two dimensions with $N = 10 \times 10$ and $U = 4$ and the same number of time slices as in the 3D case. The 2D model does not have a finite-temperature phase transition to a long-range order, rather, a crossover to a region with strong antiferromagnetic correlations. The onset of this region, which is associated with the formation of a peak in the uniform susceptibility is estimated to be around $T = 0.25$ for $U = 4$ [45,46].

III. METHODS

A. Autoencoders

Autoencoder refers to a particular set of architectures of artificial neural networks [47] that can be trained to extract features or reduce the dimensionality of big data without the specific prior knowledge of features or distinctions (in an *unsupervised* fashion). They are made up of multiple fully connected and/or convolutional layers, similar to what is used in supervised neural network machine learning algorithms. An example of the autoencoder architecture is shown in the top panel of Fig. 1. Similarly to supervised learning, such a feed-forward network can be trained by minimizing a cost function, which is defined based on the difference between neuron outputs at the output (right-most) layer and the desired output. However unlike in the supervised learning, the autoencoder is supposed to reproduce the input data [what is fed to the network in the input (left-most) layer] at the output layer; the desired output is simply the same as the input.

The hidden layers on the left half of the autoencoder are called the *encoding* layers, where, for example, an image is gradually deconstructed and represented using smaller and smaller number of pixels, whereas the hidden layers on the right half are called the *decoding* layers, where the network tries to reconstruct the image from the knowledge in the low-dimensional space. The middle layer, also known as the *coding* layer, provides the most dimensionally reduced representation of the input data after the network has been trained, allowing other clustering or machine learning methods to extract meaningful information more easily. It is known that a single-layer autoencoder with linear activations and the PCA are very similar [48]. Here, we have used deeper fully connected and convolutional autoencoders and have avoided the PCA as it is useful only for data with linear correlations. The latter is shown to have a poor performance for quantum systems relative to classical ones in classifying phases [5,49].

In our convolutional autoencoder, the network tries to extract features in encoding layers by convolving a shared filter (or kernel), which sweeps the previous layer, with small cubic subsections on the data in that layer. To further reduce the dimensionality before the next convolutional layer, a process called *maxpooling* is typically performed after each convolutional layer in which the resolution is reduced by taking the maximum value of a subsection of data and passing only that value to the next layer. The decoding is done through a general process called *upsampling*, which refers to random resampling and interpolation to put together the extracted features and increase the resolution of the data.

In fully connected autoencoders, however, each neuron in an encoding or decoding layer is connected to all the neurons in the neighboring layers, as opposed to convolutional autoencoders, where only subsections of data from the previous layer are connected to their corresponding neurons in the following layer via an adjustable filter, allowing for fewer parameters needing to be trained. We use both fully connected and convolutional autoencoders in our study using convolutional autoencoders specifically to allow spatially correlated features to be extracted more efficiently.

B. Random trees embedding

Random trees embedding transforms data in an unsupervised fashion to a high-dimensional space using tree graphs, resulting in sparser representation, for which the principal features can be extracted and mapped to a lower dimension [26–28]. Here, tree is referring to a graph with nodes repeatedly branching unidirectionally. The parent node, or the root of the tree, contains all the data and a node on a branch has a subset of the data. A node is associated with a global feature of the subset, and so, smaller branches, which contain smaller subsets farther away from the root, reveal more localized structures. Representing data in a metric space on a tree graph introduces distortions. This issue is overcome by embedding the data using an ensemble of randomized trees instead. That is, the metric space is divided into random sections with overlapping of these sections permitted. A given section is then further divided up into smaller subsets on a tree. An ensemble of such trees can make independent observations. The maximum branching depth of a tree and the number of trees are tunable parameters in the algorithm.

Once these random trees are grown, the pruning for features begins. The ensemble of trees votes for prominent features based on the density of the overlap between nodes from different trees at a given depth. Features with overlapping density lower than a certain threshold are discarded. This voting process reduces the bias and variance of those selected features.

Here, we use random trees embedding with 100 randomized trees and a maximum depth of 10 for the number of branchings. We apply the algorithm to a low-dimensional representation of the auxiliary field data for the Hubbard model obtained via a fully-connected autoencoder as outlined in Sec. IV.

C. t-SNE

PCA has been the go-to dimensionality reduction technique in condensed matter physics so far and has been successfully applied to classical models like the Ising model for extracting measures that closely resemble the order parameter or the susceptibility. However, PCA performs linear projection of the data from the high-dimensional space to a low-dimensional space by maximizing the variance of the projection, and so, local structures with non-linear correlations are not preserved. A simple measure such as the magnetization is evident to the linearity of the Ising model as each state can be projected onto a point in one dimension merely by summing its spins. The same linearity cannot be assumed for meaningful observables extracted from the auxiliary field configurations.

t-SNE is a powerful algorithm developed to preserve both global, and more importantly, local structures of data in low-dimensional space when projecting them from a high-dimensional space. Prior to t-SNE, SNE [50] was one of several attempts at achieving that. Although not very successful, it was the foundation for t-SNE. SNE employs stochastic gradient descent to minimize Kullback-Leibler divergences between pairwise conditional probability distributions that represent similarity of points, from the high- and low-dimensional spaces [14]. The distributions are obtained from Gaussian functions centered around each point. The effective number of neighbors, also known as the *perplexity*, which is provided by the user, is kept fixed by adjusting the width of the Gaussian distributions in different regions of the configuration space with different density of points. t-SNE uses a slightly different cost function and Student-t distribution, as opposed to Gaussian, in the low-dimensional space to mitigate some issues in the original SNE and provide a better performance [14].

Values between 5 and 50 are suggested for the perplexity [14,15]. We use a perplexity of about 27, which leads to the most physically interesting features across various U's in the low-dimensional representations of the field variables. The method is rather slow, scaling like $O(n^2)$, where n is the number of data [51].

D. k-means

k-means is an unsupervised clustering algorithm that locates the centroids of k clusters in the data, where k is provided by the user. The algorithm starts by k centroids distributed randomly. The next step is to calculate the euclidean distance ($L2$-norm) of each data point to each centroid and assign them to a cluster based on which centroid they are closest to. The new centroids of these clusters are then calculated and these two steps are repeated until one converges to a stable set of centroids. This type of clustering also allows for new points to be classified based on which cluster they fall closest to. It is, of course, most effective in cases where data points form well-separated clusters.

Here, we apply the k-means algorithm to the 2D output of our autoencoders or t-SNE to quantify the spread of data at various temperatures. We ask k-means to identify three centroids at each temperature. Then we find the center of the three centroids and define a temperature-dependent indicator, $\mathcal{D}$, which is the mean distance of the three centroids from their center. We note that our indicator is not unique, one may be able to come up with measures that more accurately capture the evolutions of features in the outputs as the temperature is varied.

IV. RESULTS

We start with spin configurations generated in a Monte Carlo simulation of the Ising model on a $N = 8 \times 8 \times 8$ cluster. We use a fine grid for the temperature ranging from $T = 3$ to $T = 6$ in increments of 0.01 and work with a total of about 23 000 configurations across all temperatures. Similarly to what is done in Refs. [17,20], we treat each configuration as a point in the N-dimensional space and try to deduce any features corresponding to the phase transition by reducing the dimensionality of the configuration space and visualizing it in one or two dimensions.

We use a 3D convolutional autoencoder with four encoding and decoding layers and one fully connected coding layer with either one or two neurons. The architecture is shown in Fig. 1. The input and output ($\odot$ and $\otimes$) have the same structure as the physical system. The filter used in the convolutions is $2 \times 2 \times 2$. A maxpooling layer is used after each convolutional layer. We shuffle the configurations among all temperatures, then use 70% of them to train the neural network until the accuracy, defined as one minus the mean square error between the input and the output, saturates to a value around 75%. We then feed the autoencoder with the remaining 30% of the configurations it has not seen during the training and plot the values of neurons in the coding layer (also known as the *latent variables*).

The main panel in Fig. 1 shows the output of the coding layer with two neurons. The color gradient of points represents the temperature gradient. There is a clear distinction between how data points cluster at high and low temperatures. At temperatures above the transition (red dots), we detect only one cluster. However, below the critical temperature, T_c, (blue dots) two clusters separated along both axes clearly emerge. This duality is a result of the spin inversion symmetry in the model that manifests itself in the ordered phase.

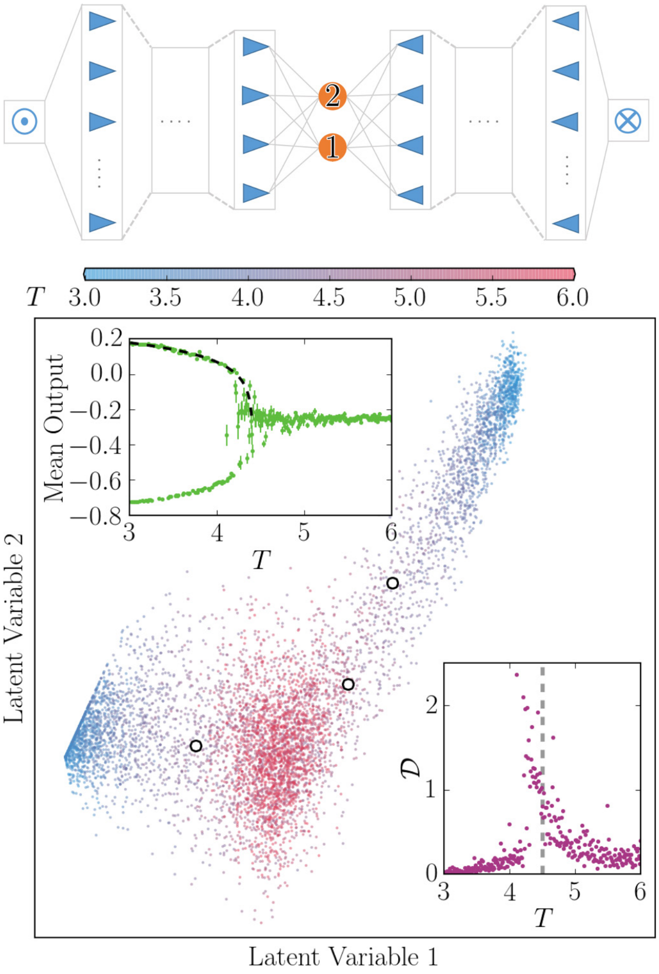

FIG. 1. Values of neurons in the coding layer of a trained autoencoder that takes Monte Carlo spin configurations of a 3D Ising model on a $N = 8 \times 8 \times 8$ lattice as the input. The architecture of our 3D convolutional autoencoder is depicted on top with neurons in the coding layer as filled (orange) circles. The encoder (decoder) part consists of four convolutional (upsampling) layers with 32, 8, 4, and 4 feature maps. Different symbol colors (red middle cluster and blue clusters on the sides) in the scatter plot correspond to different temperatures. Top inset: The output of a similar autoencoder, in which the coding layer consists of only a single neuron, as a function of temperature. The dashed line is a fit to $A(B - T)^\beta + C$ with $A = 0.38$, $B = 4.55$, $\beta = 0.34$, and C kept fixed at -0.25, which is the average output over all T. Bottom inset: Temperature dependence of the spread of the data as measured through k-means. The vertical dashed line marks the location of T_c.

At the lowest temperatures, the two clusters have the largest separation, which seems to reach a saturation value, analogous to the behavior of the order parameter (magnetization). However, we cannot properly quantify this separation as a function of temperature as the broken time reversal symmetry at a given $T < T_c$ forces all configurations to fall in one or the other low-temperature cluster, but not both. Having also predominantly one cluster of points formed at very high temperatures, suggests that in the critical temperature region, the points are the most spread out. We quantify the spread of data by applying the k-means clustering technique and requiring it to identify three clusters and their centroids (shown in Fig. 1 at $T = 4.5J$ as white circles). We plot $\mathcal{D}$ as a function of temperature in the bottom inset. $\mathcal{D}$ bears a striking resemblance to the magnetic susceptibility of the Ising model. Interestingly, thermodynamics of the system, encoded in the distribution of configurations in the importance sampling, are preserved during the dimension reduction.

In the top inset of Fig. 1, we also plot the autoencoder output in the case where we have only one neuron in the coding layer as a function of temperature. We observe a bifurcation of the neuron output as we decrease the temperature below T_c. Similar results were shown in Ref. [20] for the 2D Ising model. The neuron output looks almost exactly like magnetization of the model as a function of temperature, except for a seemingly arbitrary shift. Therefore, we fit the neuron outputs in the top branch at $T < 4.5J$ to a function proportional to $A(B - T)^{\beta}$ after a shift, where A, B and β are constants (fitted function is shown as a black dashed line in the inset of Fig. 1), and obtain $B = 4.55J$ as the critical temperature, which agrees well with $4.54J$, estimated for a system of the same size [32], and $\beta = 0.34$, close to 0.33, the critical exponent of the 3D Ising model.

Inspired by these findings, we ask if one can use a similar dimension-reduction recipe to deduce critical temperatures in quantum mechanical systems? In this work, we focus on magnetic phase transitions. We know that the perfect antiferromagnetic alignment of spins in the z direction can no longer describe the Néel state, and so, quantum fluctuations will likely blur the clear image we observe in reduced dimensions for the classical Ising model. However, to what extent can the information be still useful?

We set to answer this question by using the auxiliary fields and modifying our convolutional autoencoder such that the $\mathcal{L} = 200$ imaginary time slices are treated as different "color" channels in the input and output layers ($\mathcal{L}$ $\odot$ and $\otimes$), each with $L \times L \times L$ neurons, the same as the single channel we used for the Ising model. The architecture of the remaining network is also modified to suit the smaller L (=4) we use for the Hubbard model. We use one less hidden layer and different number of feature maps in hidden layers than in the Ising autoencoder. The output in this case is sensitive to the number of feature maps and we have chosen a set that results in the largest accuracy [52].

The results are shown in Fig. 2 for $U = 4$, 9, and 14. The first three main panels show the outcome in the case where the coding layer has two neurons. Unlike the classical case, lowering the temperature does not lead to the formation and contraction of two distinguishable clusters. Instead, the data points seem to spread out mostly along one of the two dimensions, at least for $U = 4$ and $U = 9$. Quantum fluctuations and the fact that our auxiliary field is represented along a particular direction in the spin vector space seem to play a significant role in blurring the picture. As a result, in this case, our measure of the spread of data obtained from k-means, $\mathcal{D}$, behaves more similarly to an order parameter than to the susceptibility. In Fig. 2(d) we show this quantity, along with the AFM structure factor [53], calculated in the DQMC, as a function of temperature for $U = 4$ and 9. $\mathcal{D}$ has been multiplied by a constant that minimizes the mean square distance of its values from the structure factor data (weighted by the error bars in the latter) over all temperatures. We do not find a good agreement between the two at low temperatures, however, $\mathcal{D}$ is a relatively smooth function for $U = 4$ and displays the fastest rise around $T = 0.20$, where we expect the critical temperature to be for this cluster. For $U = 9$, we observe large fluctuations in $\mathcal{D}$, which appear to grow significantly larger below $T = 0.35$, the expected Neél temperature. For $U = 14$, the indicator is dominated by noise and is not shown. Our approach seems to be most efficient in locating the critical behavior in the weak coupling regime of the Hubbard model.

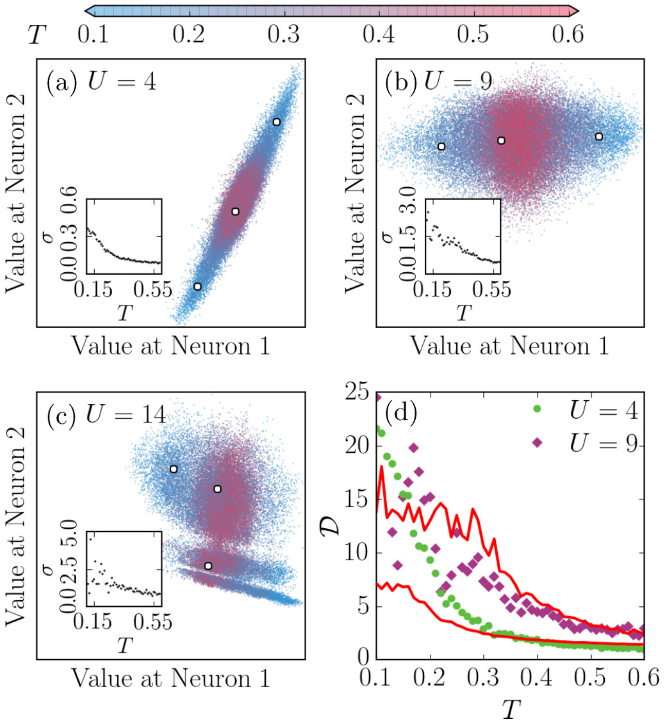

FIG. 2. The output of the 3D convolutional autoencoder for the 3D Hubbard model with (a) $U = 4$, (2) 9, and (3) $U = 14$ at half filling. Each of the $\mathcal{L} = 200$ imaginary time slices is treated as a different "color" channel. The input is a four-dimensional array of size $\mathcal{L} \times N$ with $N = 4^3$. Similarly to the output for the Ising model, the character of the output in the space of the neuron outputs of the coding layer changes as a function of temperature. However, we do not find a formation of distinct clusters below the expected Néel temperatures. In (d), we show $\mathcal{D}$ for the data in (a) and (b), normalized to fit the antiferromagnetic structure factors (shown as red solid lines) in each case, as a function of temperature. For $U = 14$, this indicator is dominated by noise, and therefore, not shown.

The same conclusion can be drawn considering the output of an autoencoder that has a single neuron in the coding layer. Unlike for the Ising model, we do not find a bifurcation as a function of temperature when projecting the data to one dimension. However, we find that the fluctuations in the data increase rapidly as the temperature decreases. So, in the insets of Figs. 2(a), 2(b), and 2(c), we show the standard deviation of the latent variable as a function of temperature. For $U = 4$ and

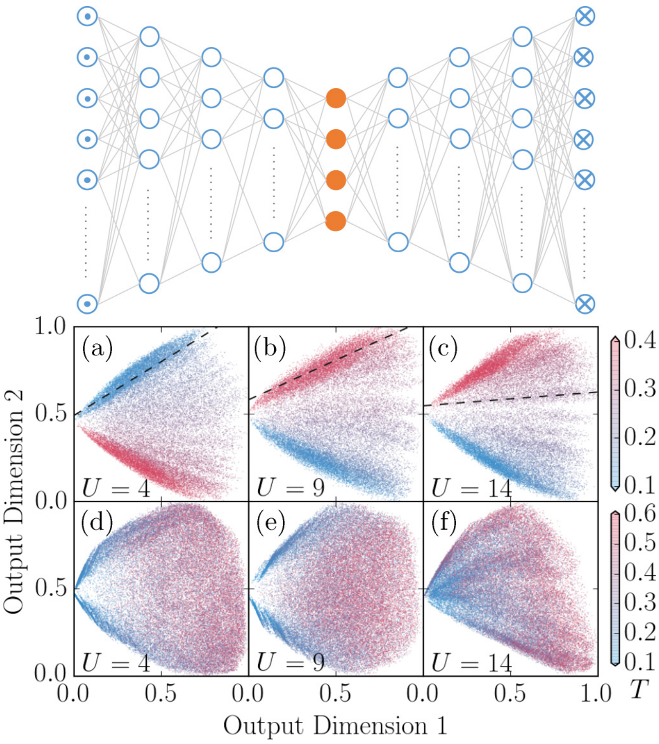

FIG. 3. The output of random trees embedding algorithm trained on the four latent variables of a fully-connected autoencoder for the 3D Hubbard model with (a) $U = 4$, (b) $U = 9$, and (c) $U = 14$ at half filling. The input layer of the autoencoder (shown on top) is a 1D array of size $\mathcal{L} \times N$. The hidden layers in the encoder and decoder parts have 80, 30, and 10 neurons. The random trees embedding algorithm further reduces the dimension of the data from four to two. In this case, the outputs clearly separate data points from different temperatures. The dashed lines are line fits to data at the estimated Néel temperature for each U. (d)–(f) Same as in (a)–(c), except that the inputs are the latent variables of the convolutional autoencoder used for Fig. 2, modified to have four neurons in the coding layer. In this case, the temperature gradient is larger along output 1 and clustering of points at low temperatures is visible for $U = 4$ and $U = 9$.

$U = 9$, it behaves similarly to $\mathcal{D}$, e.g., rapidly rises around T_N for $U = 4$. For $U = 14$, the fluctuations dominate at $T < 0.3$.

Next, we explore the possibility of having more than two neurons in the coding layer (more than two latent variables) and further reducing the dimensionality of data using another technique such as random trees embedding. We use the same convolutional autoencoder architecture as we used for Fig. 2, except that we choose to have four neurons in the coding layer. Then, we feed the output of the autoencoder to the random trees embedding algorithm and obtain a representation in two dimensions. The results are summarized in Figs. 3(d)–3(f). The emerging fan shape not only creates an approximate temperature resolution, but for $U = 4$ and 9 also exhibits two low-temperature clusters near small values along the first dimension, separated along the second dimension, reminiscent of the Ising picture. For $U = 14$, the latter feature is mostly washed away.

The convolutions in our autoencoder appear to be crucial for the low-temperature clustering in the output of the random trees embedding, but not for the temperature resolution. We try the same combination of autoencoders and random trees embedding, except that this time, we use a fully-connected autoencoder, shown in the top part of Fig. 3. The results for the latter are shown in Figs. 3(a)–3(c). We find that despite the better temperature resolution than with the convolutional autoencoder, which extends even to $U = 14$, there is no peculiarity that can point to different phases at high and low temperatures. Interestingly, data points corresponding to a given temperature spread along straight lines in this reduced space with the slope of the line correlating with temperature. We have also explored applying random trees embedding to data extracted not from the coding layer, but from various hidden layers of our fully connected autoencoder. We find that the coding layer yields the best picture in terms of temperature resolution.

Techniques like random trees embedding do not scale well with the dimension of the original configuration space and so cannot easily be directly applied to the raw auxiliary field configurations for dimension reduction. Other clustering techniques, such as the PCA or the t-SNE, can better handle larger dimensions. As we discuss below, we find that the direct application of t-SNE to the raw data yields a superior distinction between clustering patterns at different temperatures.

We apply the t-SNE algorithm [16] with perplexity of 27.07 to the half-filled 3D Hubbard configurations at our three interaction strengths. We reduce the dimensionality to 37 after preprocessing using PCA by a batch size of 500 data at a time within the t-SNE algorithm for greater speed without inducing severe distortions and to filter out some noise [14,54]. These are the same configurations as the ones we used for the autoencoder. The two-dimensional visualizations are shown in Figs. 4. For $U = 4$ in Fig. 4(a), not only the data points spread out by decreasing the temperature, similarly to the autoencoder outcome in Fig. 2, but also two distinct clusters emerge, analogous to what we observed in Fig. 3 from random trees embedding, or for the 3D Ising model. We point out, however, that unlike in the Ising case, there is a significant number of points that are scattered between the two centers at the lowest temperatures.

A similar picture emerges for $U = 9$ in Fig. 4(b), however for $U = 14$, we find that the data points stick together and form worm-like figures. As can be seen in Fig. 4(c), they mostly gather around two centers at low temperatures. Interestingly, we find that they are formed by data points that belong to the same temperature to a great extent. We attribute their formation to Mott physics. In the Mott region at large U, the freezing of charge degrees of freedom manifests itself in a significant increase in the autocorrelation time in the single spin-flip scheme of our DQMC algorithm, and the simulations become less ergodic than in the weak- or intermediate-coupling regimes. We observe this behavior despite the fact that we have attempted to mitigate the problem by performing ten different simulations of the model for $U = 14$ using different random number seeds and shuffling the configurations from those simulations before applying t-SNE.

We find a remarkable correlation of the indicator $\mathcal{D}$, calculated for the 2D t-SNE outputs, and the AFM structure factor of the model in the weak-coupling regime. The two are plotted in Fig. 4(d) for $U = 4$ after normalizing $\mathcal{D}$ to best fit the structure factor (red solid curve). They show a very

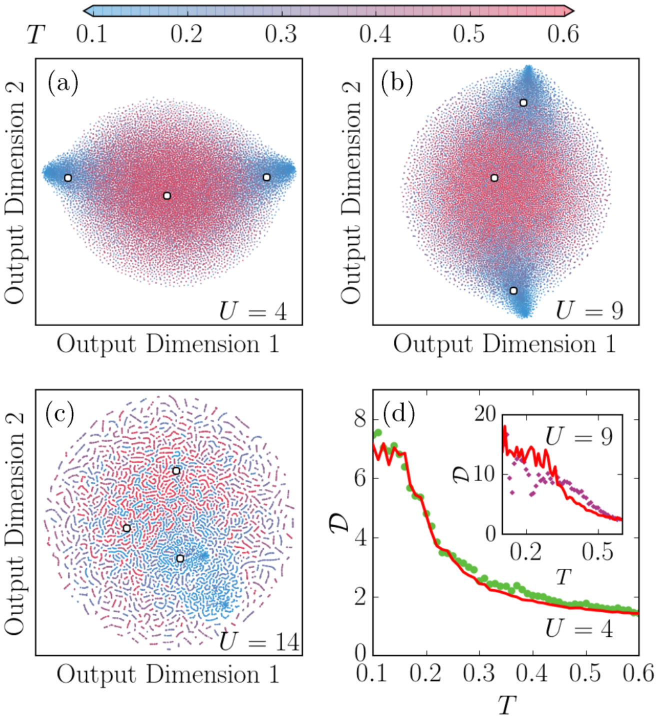

FIG. 4. Output of the t-SNE algorithm in two dimensions for the 3D Hubbard model at half filling for three different values of the interaction strength, (a) $U = 4$, (b) $U = 9$, and (c) $U = 14$. We use the raw auxiliary field configurations from DQMC simulations of the model at 51 temperatures on a uniform grid that extends from $T = 0.10$ to $T = 0.60$ as input. Different symbol colors (densities in the greyscale) in (a)–(c) correspond to different temperatures. (d) Same indicators as in Fig. 2(d) as a function of temperature calculated for the t-SNE outputs. For $U = 4$, the indicator follows the AFM structure factor very closely.

good agreement across the entire range of temperatures shown. The development of long-range correlations and the growing dissimilarities in the configurations due to the breaking of the time-reversal symmetry as we lower the temperature can be directly mapped to the increase in the structure factor.

We have also applied t-SNE to the auxiliary spin configurations for visualizations in 1D and 3D (not shown). We do not find any meaningful features in the 1D visualization. Projection of the configurations to 3D produces outputs that resemble volumetric versions of the 2D scatter plots.

Breaking the SU(2) symmetry of the Hubbard model changes the low-temperature physics and the picture obtained from unsupervised machine learning algorithms. We explore that by including a strong uniform magnetic field, $h = 1.0$, which mostly aligns the spins with the z axis and pushes the remnant AFM correlations to the xy plane. The canted AFM physics for the 3D Hubbard model is evidenced by the z and xy components of the AFM structure factor plotted in Fig. 5(d). The ferromagnetic correlations along z (not shown) are more than a factor of two larger than the xy component of the AFM correlations.

We study the effect of the magnetic field on the results from the convolutional autoencoder and the t-SNE. The output of the autoencoder with two latent variables, shown in Fig. 5(a), displays a temperature resolution; however, unlike for the original model with SU(2) symmetry, all of the low-temperature points appear to belong to a single cluster. As shown in Fig. 5(c), $\mathcal{D}$ for this case is a flat function of temperature, which misses the important phase changes in the xy plane. One can in principle define other indicators. For example, the distance between the high-temperature cluster and clusters at lower temperatures can presumably serve as a measure for the ferromagnetic correlations along z. The t-SNE, on the other hand, in addition to displaying the temperature resolution, assigns low-temperature points to a region that shrinks rapidly by lowering the temperature below $T \sim 0.3$, signaling a phase change similar to the case of zero magnetic field. This is more clearly captured by the corresponding indicator in Fig. 5(c), which now measures simply the concentration of points.

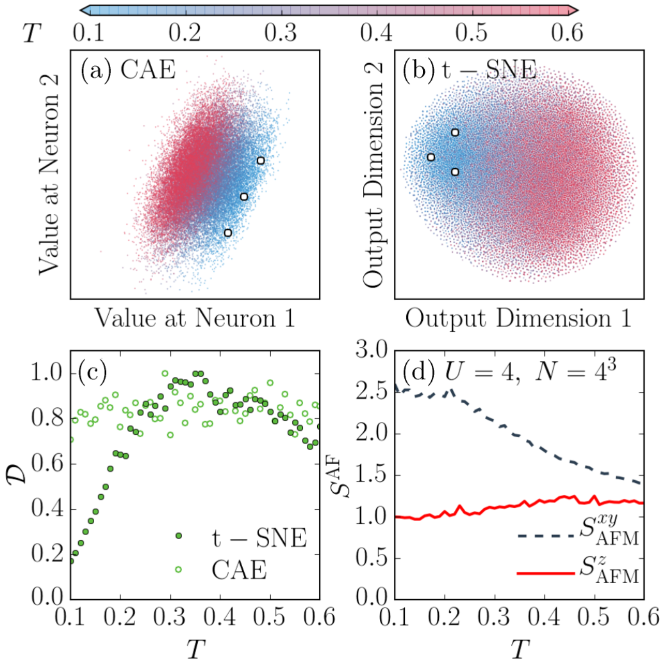

FIG. 5. (a) The convolutional autoencoder (CAE) and (b) t-SNE outputs for the 3D Hubbard model with $U = 4$ at half filling in the presence of a magnetic field $h = 1.0$. (c) The corresponding indicators as a function of temperature. (d) The AFM structure factors of the model for the z and xy components of the spin as a function of temperature.

Motivated by the ability of t-SNE to distinguish high-temperature configurations from the low-temperature ones below the critical temperatures of the 3D Hubbard model, at least for small U, we examine the 2D Hubbard model in the weak-coupling regime, also at half filling, using t-SNE with the same parameters as used for the 3D model. As shown in Fig. 6, the outcome of t-SNE in this case exhibits a similar extension of points in the space by lowering the temperature as for the model in 3D. We do not find any specific features that point to a crossover, as opposed to a phase transition expected in 3D. In fact, the indicator $\mathcal{D}$, which closely follows the AFM structure factor in this case too, exhibits an even sharper rise just below $T = 0.2$ than for the model in 3D.

One may be tempted to apply these techniques to configurations generated for the Fermi-Hubbard model away from half filling to, for example, study the fate of the

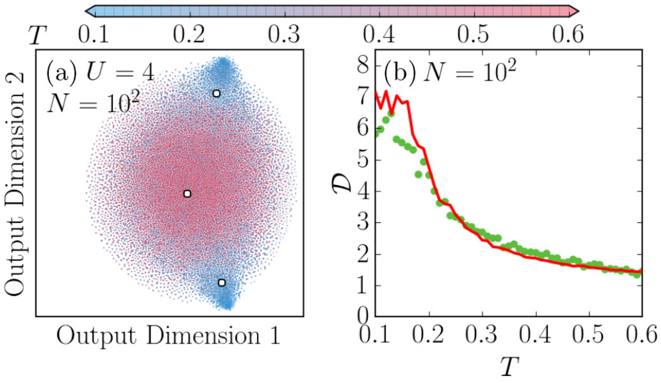

FIG. 6. Output of the t-SNE algorithm for the 2D Hubbard model with $N = 10^2$ and $U = 4$ at half filling. We use 800 raw auxiliary field configurations for each temperature in the uniform grid of T between 0.1 and 0.6. We use the same t-SNE parameters as in the training for the 3D Hubbard model in Fig. 4. (b) The indicator $\mathcal{D}$ (green symbols) and the AFM structure factor (red line) as a function of temperature.

antiferromagnetic phase of the model in 3D, or the pseudogap and superconducting properties in 2D, as the system is doped. However, any phase transition or crossover deduced from the dimensionally reduced data in that case, for example, through an indicator similar to $\mathcal{D}$, would be a transition or crossover not for the Fermi-Hubbard model, but for an alternative model whose statistics is described by the absolute value of the probability amplitudes in the DQMC simulation of the Fermi-Hubbard model. Therefore, unless it so happens that the phase boundaries of the two models are the same for the transition or crossover of interest, the results will not be of much use for the Fermi-Hubbard model.

Here, we demonstrate this concept using DQMC simulations of the 3D Hubbard model for $U = 9$ at nonzero hole doping ($\mu < 0$). We generate 2000 auxiliary spin configurations per μ that ranges from -0.2 to -4.0 in increments of 0.2 at $T = 0.16$, which is deep in the AFM phase at half filling. We first ignore the signs of the configurations and run t-SNE on 3/4 of the entire set. Then, in our visualization of the 2D output, we separate configurations with a negative sign ($\mathcal{S} = -1$) from those with a positive sign ($\mathcal{S} = 1$) at different values of μ. The results are shown in Figs. 7(a)–7(d) and Figs. 7(e)–7(h), respectively. We have seen an indicator can be defined based on the t-SNE output that behaves very similarly to a physical observable (e.g., the structure factor), although this was not exactly the case for our $\mathcal{D}$ at $U = 9$. If we assume this is valid away from half filling and separately for the positive and negative configurations, the indicators are expected to correlate with the corresponding structure factors calculated using positive only or negative only configurations [see Fig. 7(i)]. However, the physical structure factor in the presence of the sign problem in the DQMC is obtained from the difference between the above two structure factors divided by the average sign [the latter is shown in Fig. 7(j) as a function of μ]. Therefore, one cannot expect an indicator, when calculated using a mixture of negative and positive configurations at every μ (effectively ignoring the sign) to yield any physically meaningful quantity. This is specially evident in our case at $\mu < -1.0$, where the sign problem is severe and the scatter plots for $\mathcal{S} = -1$ and $\mathcal{S} = +1$ look essentially alike.

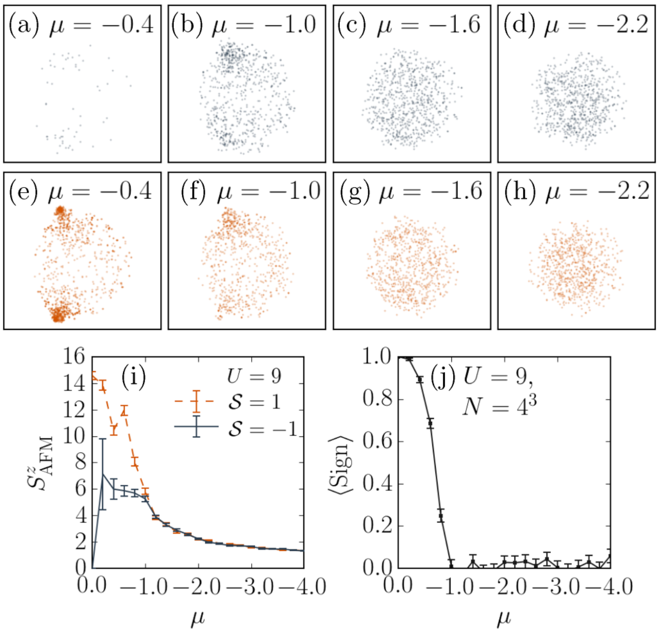

FIG. 7. (a)–(h) The t-SNE output for the 3D Hubbard model with $U = 9$ away from half-filing. We have used 1500 auxiliary field configurations per chemical potential μ, generated deep in the AFM phase ($T = 0.16$) for μ ranging from $\mu = -0.2$ to $\mu = -4.0$ in steps of 0.2. We show snapshots of the output at select μ, separated into configurations with negative sign ($\mathcal{S} = -1$) in the top row and those with a positive sign ($\mathcal{S} = +1$) in the bottom row. We use the same t-SNE parameters as in the training for the 3D Hubbard model at half filling. (i) The expectation value of the AFM structure factor obtained using configurations with positive or negative signs separately as a function of the chemical potential. (j) The average sign as a function of μ.

Finally, we examine the application of t-SNE separately on configurations with positive or negative signs to make sure that the results in Fig. 7 are not biased due to the dominance of the $\mathcal{S} = 1$ configurations near half filling. Figure 8 shows the same trend in the similarity of the scatter plots between the configurations with different signs when the average sign drops to zero.

In summary, we have applied various unsupervised machine learning techniques, such as autoencoders, random trees

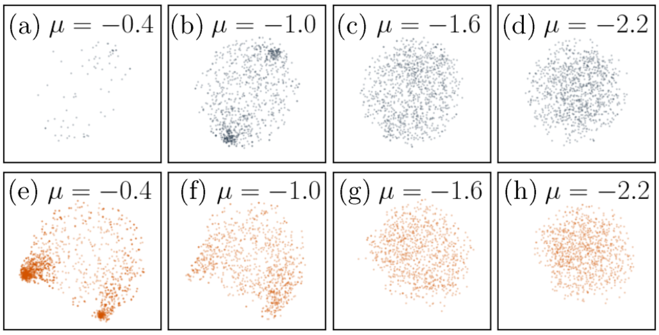

FIG. 8. Same as Fig. 7, except that here, t-SNE calculations have been performed separately on configurations with the same sign. We also use 2000 auxiliary field configurations per μ.

embedding, k-means, and t-SNE, to obtain low-dimensional representations for the auxiliary spin configurations of the Fermi-Hubbard models in different interaction regimes. We show that one can extract features from the data in reduced dimensions that resemble physical observables related to the magnetic correlations in the physical system. The configurations are sampled during DQMC simulations at finite temperatures at and away from the half-filing.

As a benchmark, we first train a convolutional autoencoder using spin configurations of a 3D classical Ising model and obtain indicators that closely resemble magnetization and susceptibility. The low-dimensional representations of autoencoders, or a combination of them with random trees embedding techniques, however, are largely affected by quantum fluctuations in the Hubbard model, preventing us from mapping the results to physical observables despite the existence of distinct features that can point to a phase transition at low temperatures at least in the weak-coupling regime.

On the other hand, we find that the t-SNE algorithm combined with k-means provides results that perfectly correlate with the AFM structure factor of the model in two or three spatial dimensions as a function of temperature. We also explore the effect of a symmetry breaking magnetic field on the outcome of the unsupervised techniques and show that t-SNE is also capable of capturing the transition to a different type of phase at low temperatures. We employ t-SNE to demonstrate that the measures we extract from low-dimensional representations of the auxiliary fields in order to describe the physics can no longer serve that purpose away from half filling in the presence of the sign problem in the DQMC simulations.

ACKNOWLEDGMENTS

We thank Demetrius Almada for providing the Monte Carlo data for the 3D Ising model. We acknowledge support from the National Science Foundation (NSF) under the Grant No. DMR-1609560. The computations were performed in part on the Teal computer cluster of the Department of Physics and Astronomy of San Jose State University and in part on the Spartan high-performance computing facility at San Jose State University supported by the NSF under the Grant No. OAC-1626645.

[1] L. Arsenault, A. Lopez-Bezanilla, O. A. von Lilienfeld, and A. J. Millis, Machine learning for many-body physics: The case of the anderson impurity model, Phys. Rev. B **90**, 155136 (2014).

[2] J. Carrasquilla and R. G. Melko, Machine learning phases of matter, Nat. Phys. **13**, 431 (2017).

[3] P. Broecker, J. Carrasquilla, R. G. Melko, and S. Trebst, Machine learning quantum phases of matter beyond the fermion sign problem, Sci. Rep. **7**, 8823 (2017).

[4] K. Ch'ng, J. Carrasquilla, R. G. Melko, and E. Khatami, Machine Learning Quantum Phases of Strongly-Correlated Fermions, Phys. Rev. X **7**, 031038 (2017).

[5] E. P. L. van Nieuwenburg, Y.-H. Liu, and S. D. Huber, Learning phase transitions by confusion, Nat. Phys. **13**, 435 (2017).

[6] Y. Zhang and E.-A. Kim, Quantum Loop Topography for Machine Learning, Phys. Rev. Lett. **118**, 216401 (2017).

[7] Y. Zhang, R. G. Melko, and E.-A. Kim, Machine learning z_2 quantum spin liquids with quasi-particle statistics, Phys. Rev. B **96**, 245119 (2017).

[8] G. Torlai and R. G. Melko, Learning thermodynamics with boltzmann machines, Phys. Rev. B **94**, 165134 (2016).

[9] G. Carleo and M. Troyer, Solving the quantum many-body problem with artificial neural networks, Science **355**, 602 (2017).

[10] G. Torlai, G. Mazzola, J. Carrasquilla, M. Troyer, R. Melko, and G. Carleo, Many-body quantum state tomography with neural networks, arXiv:1703.05334 (2017).

[11] D.-L. Deng, X. Li, and S. D. Sarma, Exact machine learning topological states, Phys. Rev. B **96**, 195145 (2017).

[12] P. Mehta and D. J. Schwab, An exact mapping between the variational renormalization group and deep learning, arXiv:1410.3831 (2014).

[13] E. Stoudenmire and D. J. Schwab, Supervised learning with tensor networks, in *Advances in Neural Information Processing Systems 29*, edited by D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett (Curran Associates, Inc., Red Hook, NY, 2016), pp. 4799–4807.

[14] L. van der Maaten and G. Hinton, Visualizing data using t-sne, J. Mach. Learn. Res. **9**, 2579 (2008).

[15] A tutorial on how to use t-SNE effectively can be found at https://distill.pub/2016/misread-tsne.

[16] Codes available at https://lvdmaaten.github.io/tsne.

[17] L. Wang, Discovering phase transitions with unsupervised learning, Phys. Rev. B **94**, 195105 (2016).

[18] I. Jolliffe, *Principal Component Analysis* (John Wiley and Sons, Ltd, New York, 2002).

[19] S. J. Wetzel, Unsupervised learning of phase transitions: From principal component analysis to variational autoencoders, Phys. Rev. E **96**, 022140 (2017).

[20] W. Hu, R. R. P. Singh, and R. T. Scalettar, Discovering phases, phase transitions, and crossovers through unsupervised machine learning: A critical examination, Phys. Rev. E **95**, 062122 (2017).

[21] C. Wang and H. Zhai, Machine learning studies of frustrated classical spin models, Phys. Rev. B **96**, 144432 (2017).

[22] P. Broecker, F. F. Assaad, and S. Trebst, Quantum phase recognition via unsupervised machine learning, arXiv:1707.00663 (2017).

[23] H. Bourlard and Y. Kamp, Auto-association by multilayer perceptrons and singular value decomposition, Biol. Cybern. **59**, 291 (1988).

[24] G. E. Hinton and R. R. Salakhutdinov, Reducing the dimensionality of data with neural networks, Science **313**, 504 (2006).

[25] F. Chollet, Building autoencoders in keras, https://blog.keras.io/building-autoencoders-in-keras.html (2016).

[26] P. Geurts, D. Ernst, and L. Wehenkel, Extremely randomized trees, Mach. Learn. **63**, 3 (2006).

[27] F. Moosmann, B. Triggs, and F. Jurie, Fast discriminative visual codebooks using randomized clustering forests, in *Proceedings of the 19th International Conference on Neural Information Processing Systems*, NIPS'06 (MIT Press, Cambridge, MA, 2006), pp. 985–992.

[28] Code availble through scikit-learn at http://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomTreesEmbedding.html.

[29] J. Hubbard, Electron correlations in narrow energy bands, Proc. Roy. Soc. London A: Math. Phys. Eng. Sci. **276**, 238–257 (1963).

[30] E. Y. Loh, J. E. Gubernatis, R. T. Scalettar, S. R. White, D. J. Scalapino, and R. L. Sugar, Sign problem in the numerical simulation of many-electron systems, Phys. Rev. B **41**, 9301 (1990).

[31] V. I. Iglovikov, E. Khatami, and R. T. Scalettar, Geometry dependence of the sign problem in quantum Monte Carlo simulations, Phys. Rev. B **92**, 045110 (2015).

[32] M. N. Barber, R. B. Pearson, D. Toussaint, and J. L. Richardson, Finite-size scaling in the three-dimensional ising model, Phys. Rev. B **32**, 1720 (1985).

[33] W. K. Hastings, Monte Carlo sampling methods using Markov chains and their applications, Biometrika **57**, 97 (1970).

[34] R. T. Scalettar, D. J. Scalapino, R. L. Sugar, and D. Toussaint, Phase diagram of the half-filled 3D Hubbard model, Phys. Rev. B **39**, 4711 (1989).

[35] R. Staudt, M. Dzierzawa, and A. Muramatsu, Phase diagram of the three-dimensional Hubbard model at half filling, Eur. Phys. J. B—Cond. Matt. Complex Syst. **17**, 411 (2000).

[36] P. R. C. Kent, M. Jarrell, T. A. Maier, and Th. Pruschke, Efficient calculation of the antiferromagnetic phase diagram of the three-dimensional Hubbard model, Phys. Rev. B **72**, 060411 (2005).

[37] T. Paiva, Y. L. Loh, M. Randeria, R. T. Scalettar, and N. Trivedi, Fermions in 3D Optical Lattices: Cooling Protocol to Obtain Antiferromagnetism, Phys. Rev. Lett. **107**, 086401 (2011).

[38] E. Kozik, E. Burovski, V. W. Scarola, and M. Troyer, Néel temperature and thermodynamics of the half-filled three-dimensional Hubbard model by diagrammatic determinant Monte Carlo, Phys. Rev. B **87**, 205102 (2013).

[39] D. Hirschmeier, H. Hafermann, E. Gull, A. I. Lichtenstein, and A. E. Antipov, Mechanisms of finite-temperature magnetism in the three-dimensional Hubbard model, Phys. Rev. B **92**, 144409 (2015).

[40] E. Khatami, Three-dimensional Hubbard model in the thermodynamic limit, Phys. Rev. B **94**, 125114 (2016).

[41] A. W. Sandvik, Critical Temperature and the Transition from Quantum to Classical Order Parameter Fluctuations in the Three-Dimensional Heisenberg Antiferromagnet, Phys. Rev. Lett. **80**, 5196 (1998).

[42] R. A. Hart, P. M. Duarte, T. L. Yang, X. Liu, T. Paiva, E. Khatami, R. T. Scalettar, N. Trivedi, D. A. Huse, and R. G. Hulet, Observation of antiferromagnetic correlations in the Hubbard model with ultracold atoms, Nature **519**, 211 (2015).

[43] R. Blankenbecler, D. J. Scalapino, and R. L. Sugar, Monte Carlo calculations of coupled boson-fermion systems, Phys. Rev. D **24**, 2278 (1981).

[44] Codes available at http://quest.ucdavis.edu/ and https://code.google.com/p/quest-qmc.

[45] T. Paiva, R. Scalettar, M. Randeria, and N. Trivedi, Fermions in 2D Optical Lattices: Temperature and Entropy Scales for Observing Antiferromagnetism and Superfluidity, Phys. Rev. Lett. **104**, 066406 (2010).

[46] E. Khatami and M. Rigol, Thermodynamics of strongly interacting fermions in two-dimensional optical lattices, Phys. Rev. A **84**, 053611 (2011).

[47] Wikipedia, Artificial neural network—Wikipedia, the free encyclopedia (2017), https://en.wikipedia.org/wiki/Artificial_neural_network.

[48] P. Baldi and K. Hornik, Neural networks and principal component analysis: Learning from examples without local minima, Neural Networks **2**, 53 (1989).

[49] R. Scalettar, Talk at the Machine Learning and Many-body Physics conference at KTIS, Beijing, China. http://kits.ucas.ac.cn/images/articles/2017/Machine_Learning/conference/10RichardScalettar.pdf.

[50] G. E. Hinton and S. T. Roweis, Stochastic neighbor embedding, in *Advances in Neural Information Processing Systems 15* (MIT Press, 2003), pp. 857–864.

[51] Z. Yang, J. Peltonen, and S. Kaski, Scalable optimization of neighbor embedding for visualization, in *Proceedings of the 30th International Conference on Machine Learning*, Proceedings of Machine Learning Research, Vol. 28, edited by Sanjoy Dasgupta and David McAllester (PMLR, Atlanta, GA, 2013), pp. 127–135, http://proceedings.mlr.press/v28/yang13b.html.

[52] We have also tried larger system sizes. We find that the autoencoder training is about 6 times slower for $L = 8$ than for $L = 4$, which is mostly due to having to add more convolution and upsampling layers. The additional complexity of the network and the added cost makes the search for architecture parameters that result in learning much more difficult.

[53] The structure factor is defined as $S^z_{\mathbf{Q}} = 1/N \sum_{ij} e^{i\mathbf{Q}\cdot\mathbf{R}_{ij}} S^z_i S^z_j$, where N is the system size, $S^z_i = (n_{i\uparrow} - n_{i\downarrow})/2$ is the z component of spin at site i, $\mathbf{R}_{ij}$ is the vector connecting sites i and j, and $\mathbf{Q}$ is the wave vector. The latter is 0 for the ferromagnetic structure factor and (π,π,π) or (π,π) for the AFM structure factor in 3D or 2D, respectively. In the absence of any SU(2) symmetry breaking terms in the Hamiltonian, this structure factor is equal to $S^{xy}_{\mathbf{Q}} = 1/2N \sum_{ij} e^{i\mathbf{Q}\cdot\mathbf{R}_{ij}} S^+_i S^-_j$, where $S^\pm_i$ are the spin raising/lowering operators.

[54] For high-dimensional data, the initial PCA analysis would be the most expensive component of the t-SNE. It scales as $(N\mathcal{L})^3$ due to a matrix diagonalization that also requires a large computer memory ($\sim$168 GB for $L = 8$ in our case). So, we have restricted our analysis to $L = 4$.