A New Strategy in Applying the Learning Machine to Study Phase Transitions

Rongxing Xu, Weicheng Fu, and Hong Zhao∗
Department of Physics and Jiujiang Research Institute,
Xiamen University, Xiamen 361005, Fujian, People’s Republic of China
(Dated: January 4, 2019)

In this Letter, we present a new strategy for applying the learning machine to study phase transitions. We train the learning machine with samples only obtained at a non-critical parameter point, aiming to establish intrinsic correlations between the learning machine and the target system. Then, we find that the accuracy of the learning machine, which is the most important performance index in conventional learning machines, is no longer a key goal of the training in our approach. Instead, relatively low accuracy of identifying unlabeled data category can help to determine the critical point with more precision, manifesting the singularity around the critical point. It thus provides a robust tool to study the phase transition. The classical ferromagnetic and percolation phase transitions are employed as illustrative examples.

Introduction — Machine learning has become a new tool for solving various physical problems such as crystal structure prediction, quantum problems and the identification of phase transitions and, in particular, the identification of phase transitions and the application of unsupervised machine learning represents a landmark in the study of phase transitions.

However, these researches are based on Ising-like models of which configurations are the results of dynamic evolutions. Here, dynamic evolution drives the system to be different clustered configurations that are distinguished according to the statistical characteristics of the configuration vectors. For other models, phase features are not always rigidly correlated to statistical properties. For example, configurations of the two-dimensional (2D) square-lattice site percolation model are represented by a binary vector with open (+1) and closed (−1) components. The open or closed state of each site is randomly determined according to the probability p or (1 − p), where p is the probability of the open state for a site. A percolation configuration is defined as a specific route that is connected from one side of the square to the opposite by sites residing in open states, which is not a consequence of dynamic evolution. This is similar to defining the shape of a smile in the binary image, which is a feature with no simple correlation to the intrinsic properties of the configuration vectors. In the case of percolation models then, the information pertaining to a desired phase feature must be included by supervisors.

On the other hand, supervised learning approaches have already been applied to phase transition problems. Wang first applied principle component analysis (PCA) to classify the two phases in the Ising model. Later, van Nieuwenburg et al. proposed the so-called confusion scheme to obtain critical points successfully for several Ising-like models. The application of unsupervised machine learning represents a landmark in the study of phase transitions.

Instead, relatively low accuracy of identifying unlabeled data category can help to determine the critical point with more precision, manifesting the singularity around the critical point. It thus provides a robust tool to study the phase transition. The classical ferromagnetic and percolation phase transitions are employed as illustrative examples.

∗zhaoh@xmu.edu.cn
ity around the critical point. The singularity may be manifested as the maximum or minimum uncertainty in identifying data categories, or sudden changes in some observable quantities. Therefore, the unorthodox application of the learning machine provides a robust tool to recover phase diagram, and even to find sub-phase structures that conventional physical approaches may ignored.

FIG. 1. Architecture of the convolutional neural network.

**Learning Machine and Models** — The machine learning proposed in the present work employs a convolutional neural network (CNN). The architecture is illustrated in Fig. 1. The convolutional layer is divided into two parts, which include a layer with eight $3 \times 3$ filters and a small three-layer network with eight $3 \times 3$ filters per layer. A rectified linear unit is adopted as the activation function. The output layer adopts a classical softmax classifier following a fully connected layer with 256 rectified linear units [26, 27].

Two models, the 2D square-lattice site percolation model and the 2D Ising model, are applied in our study. The former one has been already introduced above. The latter is defined on the square lattice with the nearest-four spin-spin interactions by the Hamiltonian, $H = -\sum_{i,j} \sigma_i \sigma_j - h \sum_i \sigma_i$, where the spin $\sigma = \pm 1$. Periodic boundary conditions are adopted here. In the thermodynamic limit with $h = 0$, a second-order phase transition occurs at the critical temperature $T_c (\approx 2.269)$, beyond which configurations will simultaneously converge to the non-ferromagnetic phase for which $\langle m \rangle = 0$, where $\langle m \rangle$ is the magnetization defined as $\langle m \rangle = \frac{1}{N} \sum_{i=1}^{N} \sigma_i$ [28]. Mathematically, configurations for both models can all be regarded as square binary images. For machine learning each image is converted to a 1D vector according to orders of its rows. Besides, the exact categories of configurations are determined by the two-pass algorithm [29] for the percolation model and the traditional Metropolis Monte-Carlo methods [28] for the Ising model.

**Unsupervised Learning** — We first illustrate that the unsupervised learning methods, including K-means algorithm, the confusion scheme and the PCA method, fail in finding the percolation phase transition. Here, the training set is obtained by sampling $10^3$ configurations every 0.01 at the entire parameter interval of $p$. K-means algorithm aims to figure out the training set into $k$ clusters in which each data point belongs to the cluster with the nearest mean [30]. To cluster configurations into two classes, we initialize 2 clustering centroids randomly, calculate the Euclidian distances from each configuration to every centroid, find the nearest one to the configuration and tag it with the label of the centroid. Positions of clustering centroids can be obtained by updating the averaging positions of all the points of configurations until they converge. With these centroids, labels can be tagged and the proportion $P$ of percolating configurations at every $p$ in the total set can be calculated. Fig. 2 (a) shows the predicted phase transition curve for two choice of the data set. One is from the interval $p \in [0,1]$ and the other from $p \in [0,1]$, respectively. It can be found that the predicting critical point is not in accordance with the real one (0.593). Instead, the predicted curves depend on the intervals of data sets.

To apply the confusion scheme, we need to guess a sequence of critical point $c'$ in the parameter interval $[c_1, c_2]$ and assign configurations at the two sides divided by $c'$ with different labels respectively. Thus, a sequence of training and testing data sets are constructed. Then, a learning machine trained by these training sets with different boundary points $c'$ should provide a sequence of accuracies on the corresponding testing set and show a “W-shape” in the range $[c_1, c_2]$ of $p$, of which the middle peak is expected to be corresponding to the critical point [21]. However, in Fig. 2 (b) we can see a “V-shape” performance curve with a minimum value at approximately $p = 0.5$. Therefore, the confusion scheme also fails in the percolation model.

Last, we check PCA. For the purpose of comparison, we investigate both models here. Let $N$ be the dimension of the vector of a configuration, such that $n$ configurations construct a $n \times N$ sample matrix $X$. In PCA, the equation $X^T X \omega = \lambda_i \omega_i$ is solved to obtain $N$ eigenvalues and eigenvectors. The corresponding eigenvectors of the first several eigenvalues are considered as principle components, according to the fact that larger eigenval-
ues carry more information. By projecting the original sample matrix to these principle components, the feature space is significantly reduced. As such, PCA represents a linear transformation that can extract mutually orthogonal directions along which samples are distributed according to their intrinsic correlations, and can therefore provide basic information for clustering samples into different classes [23]. In particular models studied here, only the first eigenvector is the dominant component [27]. For a better visual effect, we show the PCA results by 2D plots by the first two components in Fig. 3. For the Ising model, points of configurations are manifested into three clusters, corresponding to two low-temperature clusters and one high-temperature cluster in Fig. 3 (a), which are respectively coincident with the ferromagnetic and non-ferromagnetic phases in Fig. 3 (c). The clustering structure then can be easily extracted by the K-means algorithm. Also, the confusion scheme works in this situation since at $T_c$ there do exist a boundary between the two phases. However, configurations in percolation model are obtained randomly without any dynamical constrains, which leads to a non-clustering figure in the PCA’s diagram (Fig. 3 (c)). Hence, the K-means algorithm gives a trivial classification, i.e., it equally divides the data into two parts. Meanwhile, the confusion scheme introduces a high degree of confusion at c because the non-percolating and percolating configurations are mixed around this point (Fig. 3 (d)). Therefore, the unsupervised learning can be applied only for models which configuration vectors can be clustered by PCA.

**Supervised Learning** — Firstly, for the percolation model, we construct two sets of $10^4$ labeled configurations at two points $p = 0.4$ and $p = 0.5$ far from the critical point $p_c \approx 0.593$, and use them to train two CNNs. In Fig. 4 (a), we plot the predicted phase transition curves of the two trained CNNs from the $n = 10^5$ unlabeled configurations in the interval $p \in [0, 1]$. For comparison, the target curve is provided by the two-pass algorithm [28]. It can be seen that the predicted curves are very close to the target one. Especially, that of $p = 0.5$ is visually indistinguishable from the target. An analysis indicates that the $10^4$ training samples obtained at $p = 0.4$ include only 403 percolating configurations, which is about 0.4% of the total number of samples. This is quite a low proportion. Even the training samples obtained at $p = 0.5$ include only approximately 20% percolating configurations. Note that the geometries of the percolation configurations at different value of $p$, particularly those at low and high $p$, have remarkable difference [27]. Hence, the generalization ability of the learning machine is quite surprising, which can extend limited knowledge it learned to more general situations. This is the basis that the supervised learning can be applied to study the phase transition.

The learning machine of $p = 0.4$ shows a slight deviation from the target. Indeed, one can find out that the critical point obtained by this learning machine is about $p_c \approx 0.61$. As such, the accuracy of the learning machine may affect the accuracy for identifying the critical point when applying the traditional method of learning machine to study the phase transition. Thus we seek for an accuracy-independent way to identify the critical point. We divide the testing set into 100 subsets at each point of $p$, and calculated the standard deviations $\sigma$ of the predicted proportion for each $p$ when using the training data sets obtained at $p = 0.4$ and $p = 0.5$, respectively, as shown in Fig. 4 (b). The significant feature here is that the critical point appears at the maximum values of $\sigma$ for both CNNs. Moreover, these two curves has almost no obvious difference on the critical point identification. To reveal the underlying mechanism, we show accuracies of these CNNs on the testing sets along $p$, see Fig. 4 (c). It can be seen that there is a minimum on the accuracy curve, where the position of it exactly corresponds to the critical point, and is independent with the accuracy. This fact indicates that the information
The underling mechanism is explained as follows. In the non-ferromagnetic phase, \( S \) takes the Gaussian distribution. The up-dominant and down-dominant classes are divided when \( S = 0 \). If the boundary of a learning machine to classify them perfectly consist to the standard boundary, all of the configurations can be correctly classified. However, the boundary of classification often has a bias to the standard one when trained by a finite training set, and it is independent of the training algorithm. The bias, even very slight, may result a remarkable error rate when the Gaussian distribution lies around the region of \( S = 0 \). Note that the error is proportional to the area between the boundary of classification and the standard one. This situation occurs in the high-temperature region. With the decrease of the temperature, the peak of the distribution shifts, so that the standard boundary lies across the tail part of the Gaussian distribution. Since the tail of the Gaussian distribution decays in the manner faster than the exponential, the error of the learning machine decreases very quickly. Below the critical point, the Gaussian distribution function shift above or below completely to the line of zero, resulting a gap there. Only when the boundary of classification of a learning machine locate in the gap, the error rate vanishes [27].

Here a relatively inaccurate learning machine may be helpful. As a particular example, in the case of \( h = 0 \), the direct counting should give a constant curve of \( P = 0.5 \) independent of the temperature because of the symmetry of the system, which provides no information of the critical point. However, the learning machine in Fig. 5 (a) still reveals the critical point following the criterion that the gradient turns to vanish.

The plateaus of \( P = 1 \) for \( h > 0 \) and \( P = 0 \) for \( h < 0 \) represent the symmetry breaking that each configuration takes a definite positive or negative magnetization slightly under the critical point. Another type of plateaus left to plateaus of \( P = 1 \) and \( P = 0 \) indeed reveal a sub-phase. In this phase, certain initial configurations may evolve to a stationary state with positive magnetization for \( h > 0 \). Similar phenomenon appears when \( h < 0 \). Particularly, it is interesting that the proportion of these initial configurations keep to be constant in this phase, independent of the temperature. Further studies show that plateaus in this phase are finite-size effects, since they may eventually approach the plateaus of \( P = 1 \) or \( P = 0 \) with the increase of the system size. Nevertheless, it can also be an inevitable effect in nanoscale systems. According to our knowledge, this phenomenon has not been documented in Ising-like models. If concatenate the turning points on the two edges of the plateaus of \( P = 1 \) and \( P = 0 \), two lines divide the parameter plane \((h,T)\) into three regions: a low-temperature ferromagnetic phase region (I), a high-temperature ferromagnetic phase region (II), and a non-ferromagnetic phase region (III) shown in Fig. 5 (b).

**Summary** — One can apply the learning machine in a non-traditional way to design automated algorithm to study phase transitions. With the new strategy, the pur-
pose of training is to establish the intrinsic correlation between the learning machine and the target system, and only a training set made at a non-critical parameter is fascinatingly sufficient to fulfil this purpose. The critical point, as well as other possible inter-phase transition points, can be identified according to the global behavior of the learning machine. This method is insensitive to the accuracy of the learning machine. We hope this strategy open a new road to study physical problems using learning machine.

We acknowledge the support by NSFC (Grant No. 11335006).

[1] R. Girshick, J. Donahue, T. Darrell, and J. Malik, in Proceedings of the IEEE conference on computer vision and pattern recognition (2014) pp. 580–587.
[2] A. Toshev and C. Szegedy, in Computer Vision and Pattern Recognition (2014) pp. 1653–1660.
[3] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, in 2014 IEEE Conference on Computer Vision and Pattern Recognition, Vol. 00 (2014) pp. 1725–1732.
[4] M. J. Jordan and T. M. Mitchell, Science 349, 255 (2015).
[5] D.-L. Deng, X. Li, and S. Das Sarma, Phys. Rev. B 96, 195145 (2017).
[6] P. Rowe, G. Csányi, D. Alfé, and A. Michaelides, Phys. Rev. B 97, 054303 (2018).
[7] Y.-Z. You, Z. Yang, and X.-L. Qi, Phys. Rev. B 97, 045153 (2018).
[8] J. Venderley, V. Khemani, and E.-A. Kim, Phys. Rev. Lett. 120, 257204 (2018).
[9] N. Liu and P. Rebentrost, Phys. Rev. A 97, 042315 (2018).
[10] J. Gray, L. Banchi, A. Bayat, and S. Bose, Phys. Rev. Lett. 121, 150503 (2018).
[11] N. Walker, K.-M. Tam, B. Novak, and M. Jarrell, Phys. Rev. E 98, 053305 (2018).
[12] P. Ponte and R. G. Melko, Phys. Rev. B 96, 205146 (2017).
[13] P. Zhang, H. Shen, and H. Zhai, Phys. Rev. Lett. 120, 066401 (2018).
[14] S. Curtarolo, D. Morgan, K. Persson, J. Rodgers, and G. Ceder, Phys. Rev. Lett. 91, 135503 (2003).
[15] G. Carleo and M. Troyer, Science 355, 602 (2017).
[16] L.-F. m. c. Arsenault, A. Lopez-Bezanilla, O. A. von Lilienfeld, and A. J. Millis, Phys. Rev. B 90, 155136 (2014).
[17] J. Gao, L.-F. Qiao, Z.-Q. Jiao, Y.-C. Ma, C.-Q. Hu, R.-J. Ren, A.-L. Yang, H. Tang, M.-H. Yung, and X.-M. Jin, Phys. Rev. Lett. 120, 240501 (2018).
[18] Y. Zhang and E.-A. Kim, Phys. Rev. Lett. 118, 216401 (2017).
[19] M. E. J. Newman and G. T. Barkema, Topics in Current Physics 46, 252 (1999).
[20] G. Stockman and L. G. Shapiro, Computer Vision (Prentice Hall, 2001) pp. 69–73.
[21] J. MacQueen et al., in Proceedings of the fifth Berkeley symposium on mathematical statistics and probability, Vol. 1 (Oakland, CA, USA, 1967) pp. 281–297.