Supplementary material for "A learning algorithm reflecting universal scaling behavior near phase transitions"

N. Maskara,[1,2,*] M. Buchhold,[1,3] M. Endres,[1] and E. van Nieuwenburg[1,4]

[1]*Division of Physics, Mathematics and Astronomy, California Institute of Technology, Pasadena, CA 91125, USA*
[2]*Department of Physics, Harvard University, Cambridge, MA 02138, USA*
[3]*Institute for Theoretical Physics, University of Cologne, Zülpicher Strasse 77, 50937, Cologne, Germany*
[4]*Niels Bohr International Academy, Niels Bohr Institute, University of Copenhagen, Blegdamsvej 17, 2100 Copenhagen, Denmark*

(Dated: December 14, 2021)

I. NETWORK ARCHITECTURE

In this section, we provide a few additional details on our neural network implementation. As explained in the main text, each module m^k is a two layer convolutional neural network. The first convolution takes an ℓ_k sized input to a vector of $4\ell_k$ features, and applies a ReLU non-linearity defined by $\mathrm{ReLU}(x) = \max(0, x)$. Then, the second convolution collapses the $4\ell_k$ features into a single value, and applies a tanh activation function $\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$. The module outputs are then combined using a linear classifier, described in the main text. The network architecture for the third iteration $k = 3$ is shown in Fig. S4. Note that in our implementation, for the first module m^1 we only apply a single convolution layer with linear activation, which can approximate all mappings from a binary input to a real valued output.

II. CORRELATION FUNCTIONS IN DIFFERENT BASIS

Here, we discuss correlations functions of the TFIM in different measurement basis. Conventionally two helpful correlation functions in the Ising model are the two-point correlation function $C_2(l) = \langle \sigma^z_{m+l} \sigma^z_m \rangle$ and the string (parity) order correlation function $C_s(l) = \langle \prod_{m=1}^{l} \sigma^x_m \rangle$, which both are sharp indicators for the ordered (disordered) phase [1].

Generalized to an arbitrary measurement basis, which we parametrize with the angle θ, the two-point correlation function $C_2^\theta(l) = \langle \sigma^\theta_{m+l} \sigma^\theta_m \rangle$ and the string order correlation function $C_s^\theta(l) = \langle \prod_{m=1}^{l} \sigma^\theta_m \rangle$ are expressed as functions of the spin $\sigma^\theta_l = \cos(\theta)\sigma^z_l + \sin(\theta)\sigma^x_l$. For $\theta = 0$ this corresponds to correlation functions measured in the z basis while $\theta = \frac{\pi}{2}$ correspond to measurements in the x basis. We also consider intermediate bases with $\theta = 0.18\pi, \pi/4$, and 0.32π.

Correlation functions like C_2^θ, C_s^θ can be reconstructed from individual snapshots (projective measurements) by averaging over many different positions (due to translational invariance), and are thus accessible to the machine learning algorithm as well. Furthermore, in the z and x bases, these are sharp identifiers of the corresponding phase, and also witness the scaling with distance l. It is therefore likely that the algorithm learns (a variant of) these correlation functions. However, these are two of the simplest correlations functions one can consider, and most generally the network can capture any linear combination of correlation functions constructed from σ^θ.

In Fig. S2, we display the correlation functions C_2^θ, C_s^θ for a set of angles θ, obtained from the ground state for different values of g. We observe that in the intermediate bases, the contrast in these correlation functions on the two sides of the transition is reduced. Most strikingly at $\theta = \frac{\pi}{4}$, both correlation functions appear identical away from the critical point, and only change qualitatively close to $g = g_c$. Thus, the two phases are not readily distinguishable by studying these two "conventional" correlation functions in this basis, yet the algorithm learns to distinguish them successfully.

III. COMPARISION OF THE ALGORITHM WITH OTHER TECHNIQUES

In this section, we further support the claim that classifying phases of the transverse field Ising model in different bases is signficantly more difficult. This justifies it as a benchmark for our algorithm.

Specifically, we compare the CNN based classifier to one which can only capture one and two-site correlations. Specifically, for each snapshot and basis, we compute both $C_1^\theta(l) = \langle \sigma^\theta \rangle$ and $C_2^\theta(l)$ for $l = 1, ..., 18$, and use these as the input for a linear classifier trained using scikit-learn's stochastic gradient descent (SGD) package. Results at intermediate rotation angles near $\theta = \pi/4$ clearly show that two-point correlations are insufficient to reliably distinguish between the two phases, with the classifier performing no better than random guessing in the ordered phase. In contrast, the machine learning model performs similarly to the z-basis ($\theta = 0$ case). We see that including both one and two-point correlators is sufficient to reproduce the performance of the machine learning model in this basis, suggesting these are the only correlations the neural network needs to capture.

However strikingly, near $\theta = \pi/2$ these few-site observables are unable to match the performance of the machine learning algorithm. This is because the ideal order parameter in this regime is the parity string order correlation function described above. This shows the network is learning correlation functions beyond two-point correlators, as we claim in the main

* nmaskara@g.harvard.edu

text.

This test also illustrates how as we rotate the bases, the optimal correlation functions for phase classification should change. As such, the problem of classifying phases in different bases is an interesting benchmark for neural network scaling to pass.

IV. DIFFERENT LABELLINGS

In the main text, we always partition our data into ordered (1) and disordered (0) based on knowledge of the critical point g_c. However, in many experimental settings, the critical point may not be known a priori. Hence, in this section, we present some results where a different threshold g_{th} was used to partition the data. Specifically, we chose $g_{th} = 1.05$, label snapshots from $g < g_{th}$ ordered, snapshots with $g > g_{th}$ disordered, and repeat the same training procedures. However, when we look at the classification $\langle M^\ell \rangle$ on a validation set, we see the the accuracy stops improving beyond $\ell \approx 6$, see Fig. S1. This likely reflects the finite correlation length at $g_{th} = 1.05$, versus the divergent correlation at $g = 1.00$, which is what enabled improvement at larger ℓ in main text Fig. 2. This also suggests that scaling of M^ℓ could be combined with confusion methods [2] to aide in identification of critical points.

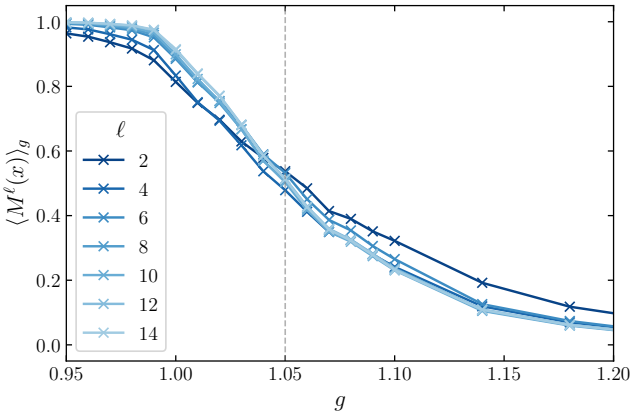

Figure S1. Classifications curves for the incorrectly chosen partition $g_{th} = 1.05$. Notice that the classification improves as we increase ℓ for $\ell < 5$, but then saturates and stops improving. This is likely a reflection of the finite correlation length at $g_{th} = 1.05$.

[1] P. Pfeuty, Annals of Physics **57**, 79 (1970).
[2] E. van Nieuwenburg, Y.-H. Liu, and S. Huber, Nature Physics **13**, 435 (2017).

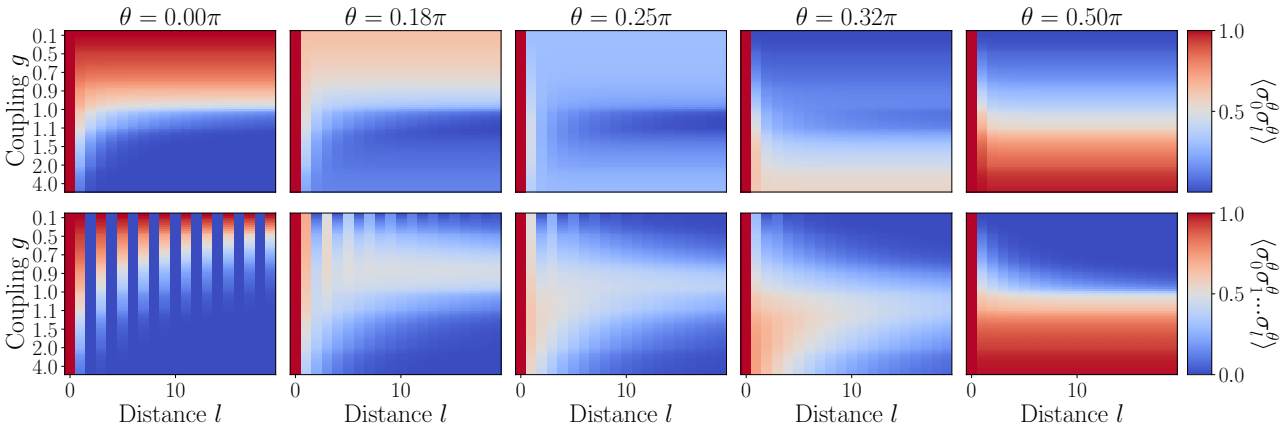

Figure S2. Two-point correlation function $C_2^\theta(l)$ (top row) and string order correlation function $C_s^\theta(l)$ (bottom row) obtained from the ground state of the TFIM in different measurement basis (parametrized by θ). We see that $C_2(l)$ exhibits scaling with l in the z basis, but not in the x basis, while the opposite is true for $C_s(l)$. Away from the critical point, depending on the angle θ, at least one of the two correlation functions unambiguously distinguishes the two phases. An exception is the peculiar behavior at $\theta = \frac{\pi}{4}$, which we discuss in the text.

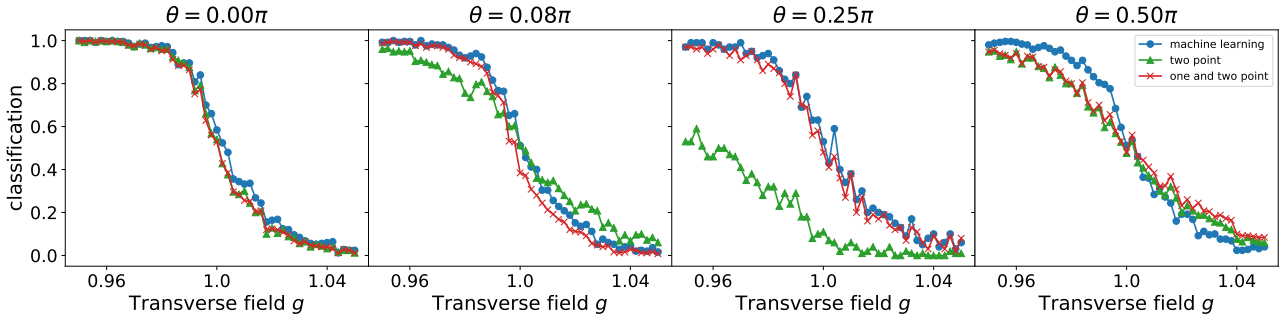

Figure S3. Comparison of the performance of neural network classifier M^{18} versus simpler models. Machine learning is significantly more powerful because it can capture multi-site correlations, and use this to improve the classification accuracy. We see for the xz basis ($\theta = \pi/4$), the classification accuracy is significantly improved by using both one and two-point correlators instead of just two-point correlators. The ML model captures this. Further, for the x basis ($\theta = \pi/2$), the ideal order parameter is the parity, which cannot be captured with only one and two-point correlators. There, we see the network considerably outperforms these local classifiers, showing it is harnessing multi-site correlations.

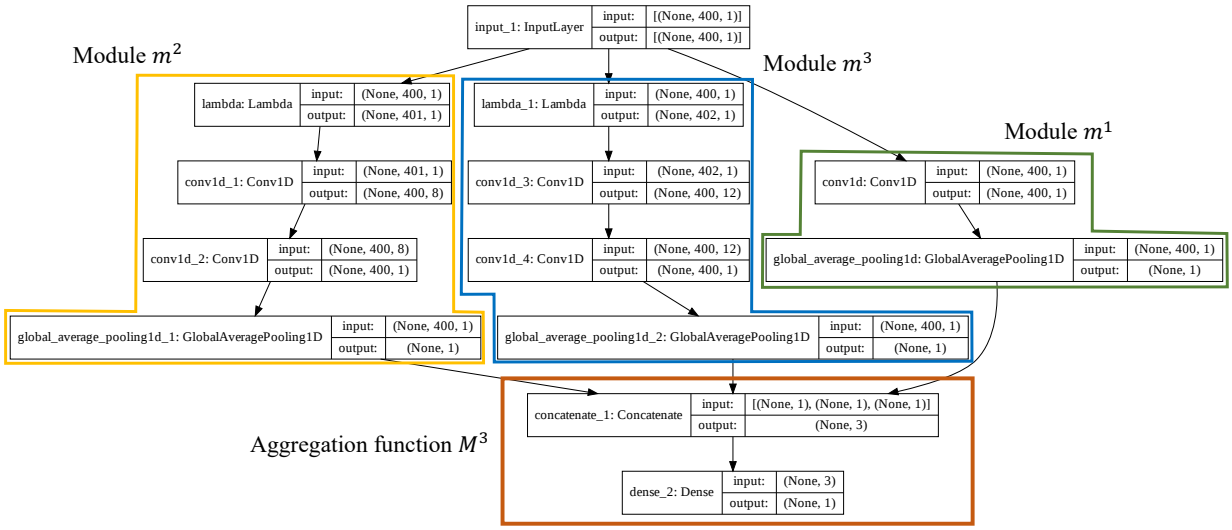

Figure S4. Network architecture at the third iteration, calculating the function denoted M^3. The three modules m^1, m^2, m^3 are convolutional layers, added in parallel, and are combined by a standard linear classifier. The Lambda layer is a custom layer, added to simulate convolutions which wrap around the periodic boundaries. The activation function for the first convolution (conv1d_1,conv1d_3) is a ReLU, while the activation for the second convolution (conv1d_2,conv1d_4) is tanh. The final dense layer has a sigmoid activation. Higher iterations have more modules added in parallel. For the Potts model, an equivalent architecture but with 2D convolutions was used.