Phase transition detection without order parameters

Ratul Das and Mihir Chandra
School of Physical Sciences
National Institute of Science Education and Research
Jatni, Odisha 752050
`ratul.das@niser.ac.in and mihir.chandra@niser.ac.in`

Abstract

Phase transitions are critical points in the thermodynamic behavior of many physical systems. Thus, detecting these transitions accurately is important for understanding and predicting systemic behaviour. However, it is not always possible or practical to obtain the order parameter (an indicator of a phase transition) for a given system, especially near the phase boundary. In this report, we propose a machine learning approach for phase transition detection, which can be applied to study the second order phase transition for the 2d Ising model without the being provided an order parameter, by employing a convolutional neural network.

1 Introduction

Traditionally, phase transitions are detected by analyzing the thermodynamic properties of a system, such as its specific heat, susceptibility, or magnetization. They can be classified into two kinds - first order and second order transitions. In general, first order transitions encompass transitions where latent heat exchange is involved (Ice to water (or vice-versa) or sublimation of solid naphthalene into gaseous naphthalene). In second order transitions, the change in the state of the system is comparatively less sensitive to the thermodynamic conditions. For example, the transition between the ferromagnetic and paramagnetic phases for a material, or between superconducting and non-superconducting phases, both of which occur across a certain critical temperature of the system.

Measuring thermodynamic variables can turn out to be challenging for complex systems with high-dimensional thermodynamic properties. In recent years, machine learning has shown great promise in analyzing complex systems and detecting phase transitions. Machine learning approaches have been used to analyze the spin configurations of the Ising model [4] and the XY model [3].

In this paper, we propose a machine learning approach for phase transition detection that can be applied to a wide range of physical systems. Specifically, we employ a Convolutional Neural Network (CNN) to study the correlation between the spin configuration and temperature for the 2D Ising model, which is known to harbour a second order phase transition. The 2D Ising model was simulated using the Metropolis Hastings algorithm, and a CNN algorithm was used to find the characteristic feature of the phase transition.

2 Theory

2.1 Ising Model

The model consists of a lattice of spins, where each spin can be either up or down. The energy of the system is given by the Hamiltonian

$$H = -J \sum_{\langle i,j \rangle} s_i s_j - h \sum_i s_i, \tag{1}$$

Preprint. Under review.

where s_i is the spin at site i, J is the interaction strength between neighboring spins, $\langle i, j \rangle$ denotes the sum over neighboring sites, and h is an external magnetic field. The Ising model exhibits a phase transition between a ferromagnetic phase, where all spins are aligned, and a paramagnetic phase, where the spins are randomly oriented. The magnetisation is given as:

$$M = \frac{1}{N}\sum_{i=1}^{N} s_i \tag{2}$$

from where we can get the critical temperature T_c as:

$$\frac{kT_c}{J} = \frac{2}{\ln(1+\sqrt{2})} \tag{3}$$

2.2 Metropolis Algorithm

We initialize a two-dimensional lattice of spins, with each spin randomly oriented up or down. We then choose a spin at random and propose a new orientation for that spin. We compute the change in energy ΔE resulting from the proposed change in orientation, and use the Metropolis acceptance criterion to decide whether to accept or reject the proposed change.

The acceptance probability for a proposed spin flip is given by:

$P_{\text{accept}} = \min\left(1, e^{-\Delta E/kT}\right)$

where ΔE is the change in energy resulting from the proposed spin flip, and T is the temperature of the system. If the change in energy ΔE is negative, indicating that the proposed change lowers the energy of the system, then the spin flip is accepted. If the change in energy ΔE is positive, indicating that the proposed change raises the energy of the system, then the spin flip is accepted with a probability proportional to $e^{-\Delta E/T}$ and continue the process till equilibrium is reached.

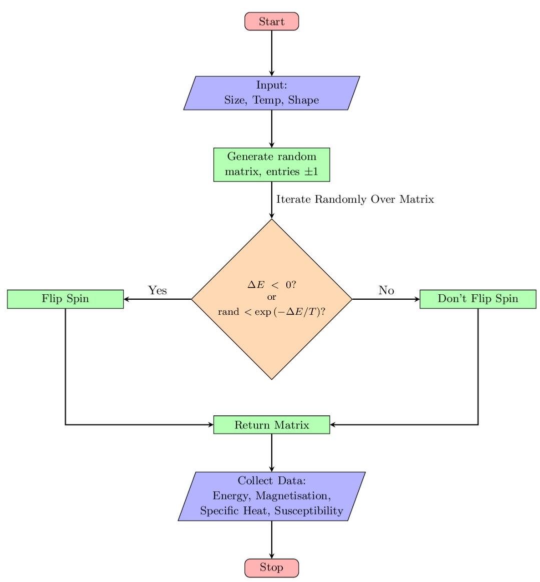


Figure 1: Flowchart depicting the metropolis algorithm [2]

2.3 Cross-entropy loss function

The loss function that we make use of in our classifier algorithms is given as:

$$L_{CE} = \sum_{j=1}^{n} y_i^* . \log y_i \tag{4}$$

where y_i^* refers to the true probability distribution and y_i refers to the predicted class distribution

We know that entropy in the 2d Ising model is given as,

$$S = -\frac{\partial F}{\partial T} \tag{5}$$

Here, F is the free energy, and T is the temperature. Free energy is defined as follows:

$$F = -kT\ln(Z) \tag{6}$$

Here, Z is the partition function, also known as the sum over states (in this case, spins across the lattice) for the system. It can be proved that:

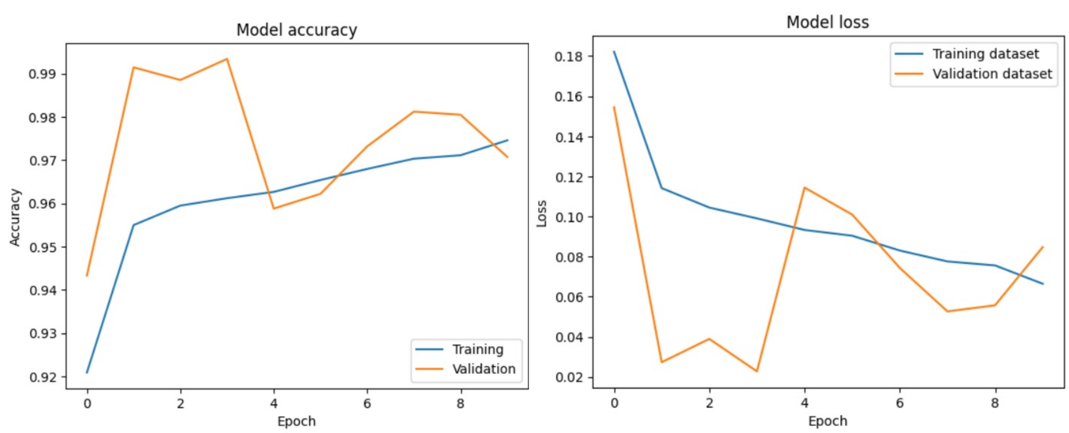
$$S = k\sum_i \rho_i \ln \rho_i \tag{7}$$

Later in the experiment section, we try to make use of this quantity to achieve one of the main objectives in the context of this project.

3 Baseline

We ran a sequential CNN (Tensorflow 2.11 [1]) to check the presence of a phase transition on snapshots for the 2d Ising model, when the CNN is trained on snapshots away from the phase boundary. The CNN model consists of two convolutional layers with 32 and 64 filters, respectively, followed by max pooling layers, a flatten layer, and two fully connected layers with 64 neurons and 1 neuron respectively. The output layer uses sigmoid activation to output a probability of the system being in the ordered phase. We compile the model with binary crossentropy loss and Adam optimizer, and trained it for 10 epochs with 20% of the data used for validation.

Training accuracy and loss were seen to follow a clear trend (increasing and decreasing respectively). Final training accuracy was 97.46%. Final training loss was 0.0665 However, validation accuracy and loss were not as consistent in terms of showing a trend (see 2a and 2b)



(a) (b)

Figure 2: Model accuracy and model loss for the sequential CNN

4 Experiments

4.1 GitHub

The datasets and the code can be found on this GitHub repository.

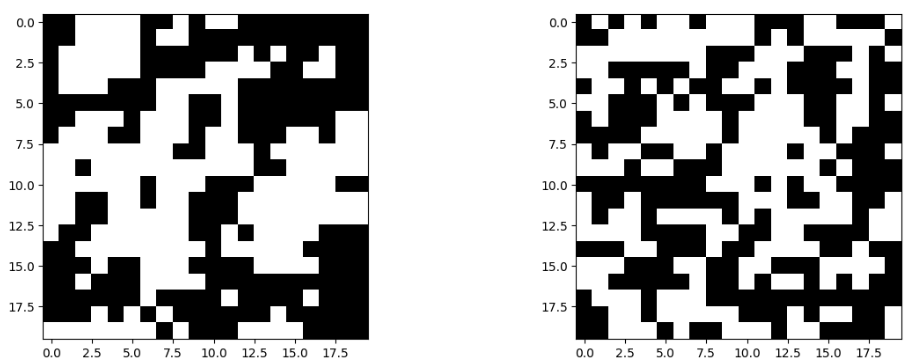
4.2 Description

Earlier, the CNN model had been trained on a dataset where the snapshots were produced from simulations away from the phase transition to check how well it performs for snapshots that were produced for simulations away and near the critical point.

However, we decided to take a different approach for the new dataset, intending to train our algorithms over the critical points as well.

4.3 Dataset

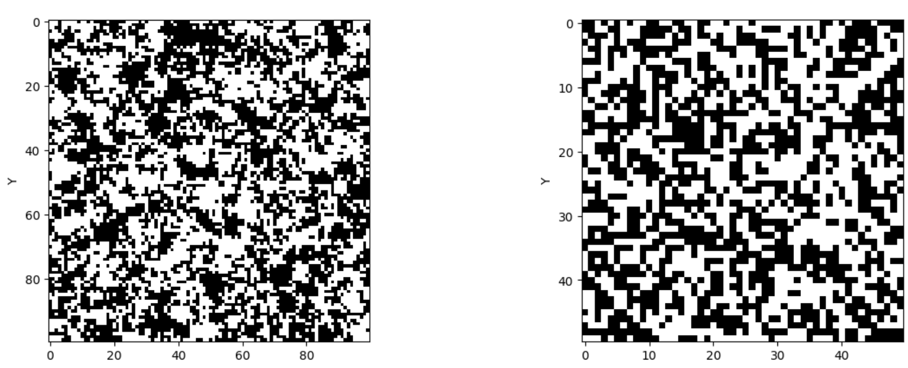
We produced 20,000 snapshots from Ising model simulations for each combination of coupling constant and lattice size. We produced datasets with lattice sizes 20×20 (see 3a and 3b), 50×50 (4a), and 100×100 (4b), and with coupling constants 0.5, 1, 2, 5, and 10. In order to check the simulations, energy vs temperature (see 6b and 6a), magnetisation vs temperature (see 5b and 5a) curves were plotted for all of these datasets. The same general for both magnetisation and energy was seen to hold up for various lattice sizes, but in case of variation in the coupling constant, there was seen to be a breakdown in the form of the curves observed (both in case of energy and magnetisation).



(a) T = 0.05 (b) T = 4

Figure 3: Snapshots of 2d Ising model at two extremes

In the ordered phase, one can see formation of larger islands of homogeneous spins, whereas in the disordered phase, noise appears to dominate.

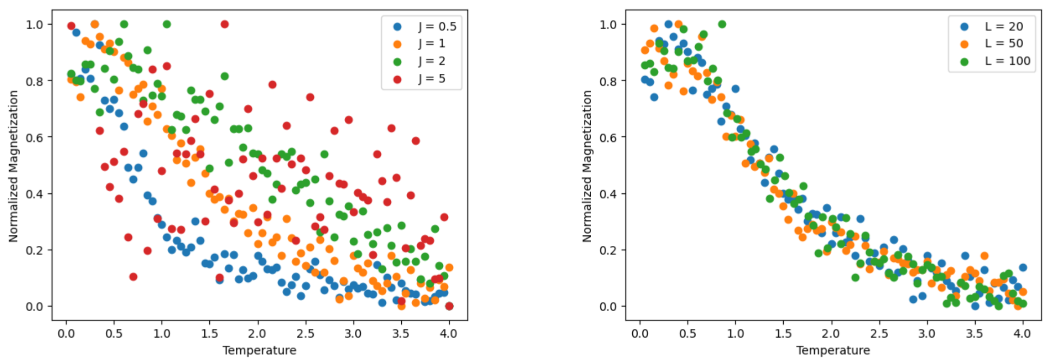


(a) L = 100 at T = 0.05 (b) L = 50 at T = 3.5

Figure 4: Snapshots of 2d Ising model with larger lattice size datasets

4.4 Analysis of the dataset

To validate that our dataset, we studied how some parameters like magnetization and energy varied with the temperature. We studied the variation for different values of J (interaction term) and also different lattice sizes. We normalized the values to help us better compare the trend for different attributes.

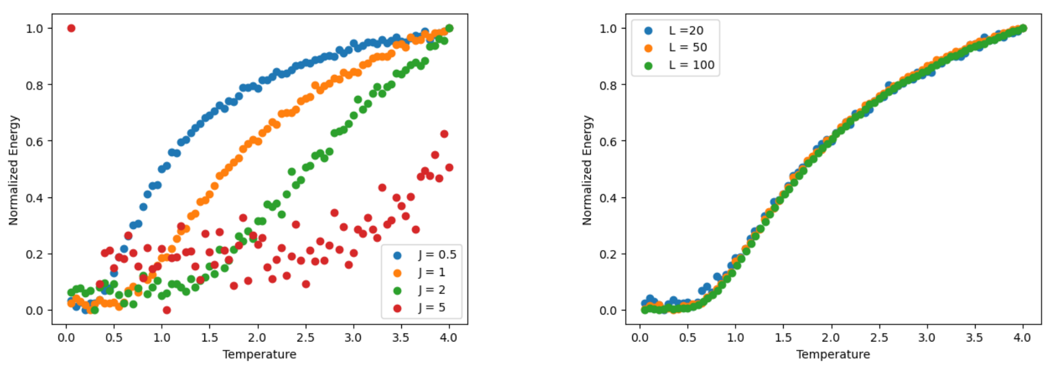


(a) Normalized Magnetization vs Temperature for different values of J

(b) Normalized Magnetization vs Temperature for different lattice sizes

Figure 5: Normalized Magnetization vs Temperature plots

The plots matched the experimentally observed trend for 2D Ising model. Hence our dataset was performing was as expected.
Moreover, we also observed that for a high J value, say 5, the data for magnetisation was scattered, which could be due to the system not getting equilibriated. For higher values of J, due to stronger interactions, the system becomes highly ordered and more likely to be in a low energy state, which can be observed from the graph.



(a) Normalized Energies vs Temperature for different values of J

(b) Normalized Energies vs Temperature for different lattice sizes

Figure 6: Normalized Magnetization vs Temperature plots

4.5 Training and Testing

The baseline model was able to distinguish between ordered and disordered phase snapshots with an accuracy of 100% when tested on snapshots within the temperature ranges the CNN was trained on (50 samples). For temperature ranges much closer to Tc, the model was accurate 76% of the time (50 samples). For a second order transition, there is a bit of smoothness to the change in magnetisation with respect to temperature, hence some confusion is expected near T_c.

4.6 Temperature prediction

The training was done on a temperature range of 0.05 to 4, with 250 snapshots at each temperature. We aimed to predict the temperature for each of our snapshots. We trained a sequential CNN model on it with the following layers (7).

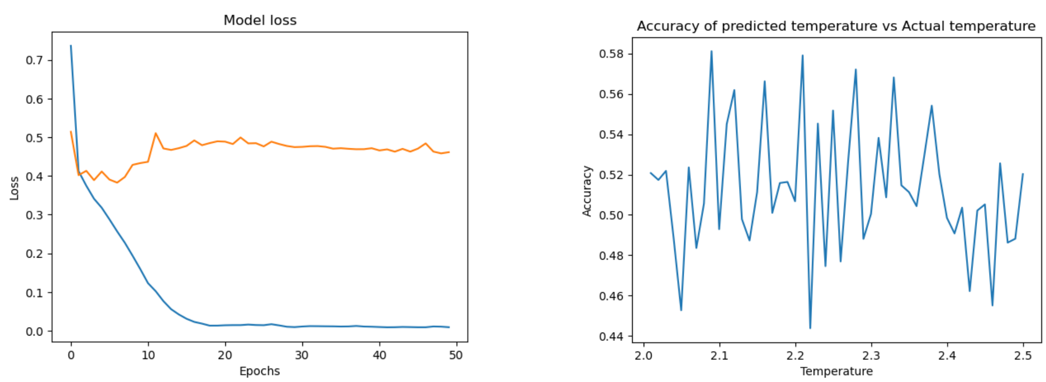
Model: "sequential"

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 18, 18, 64)	640
conv2d_1 (Conv2D)	(None, 16, 16, 128)	73856
max_pooling2d (MaxPooling2D)	(None, 8, 8, 128)	0
conv2d_2 (Conv2D)	(None, 6, 6, 256)	295168
flatten (Flatten)	(None, 9216)	0
dense (Dense)	(None, 256)	2359552
dense_1 (Dense)	(None, 1)	257

Total params: 2,729,473
Trainable params: 2,729,473
Non-trainable params: 0

Figure 7: Model architecture for regression CNN model on $L = 20$

For our testing dataset we had 100 snapshots at each temperature in the range of 2.01 to 2.5. We analysed the accuracy for the temperature predicted by our CNN model and we got accuracy in the range of 0.4231 to 0.604. The poor result in accuracy is due to the model not being able to classify correctly for datasets near the critical temperature where transition from order to disorder takes place.

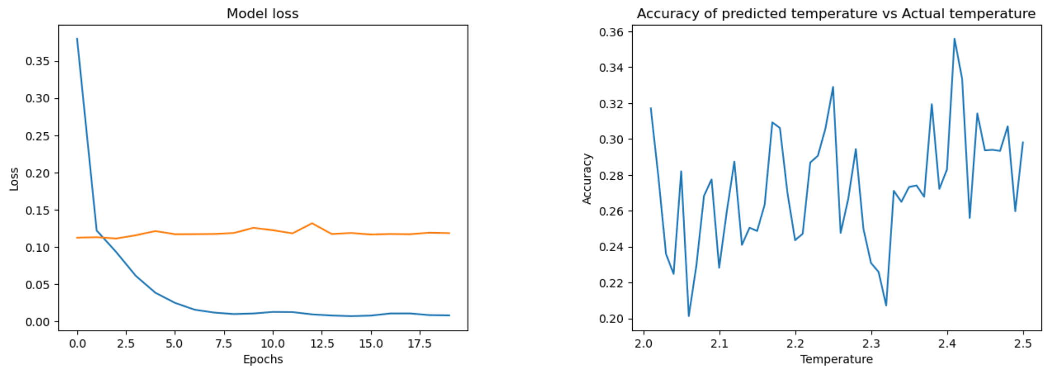


(a) Training loss vs epochs

(b) Accuracy at different temperatures

Figure 8: Metric for temperature prediction near T_c for lattice size of 20

We ran the same algorithm (with changing the filter size for the CNN layers) for a 50∗50 lattice size. There was a considerable drop off in the accuracy compared to 20∗20, and we aim to improve this by changing the model architecture.

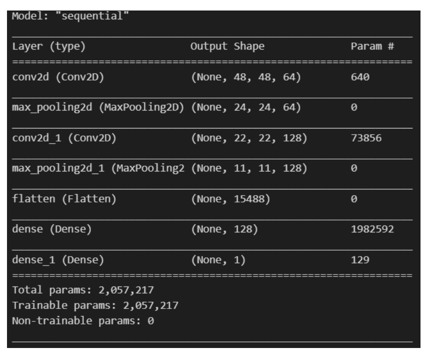


(a) Training loss vs epochs (b) Accuracy at different temperatures

Figure 9: Metric for temperature prediction near T_c for lattice size of 50

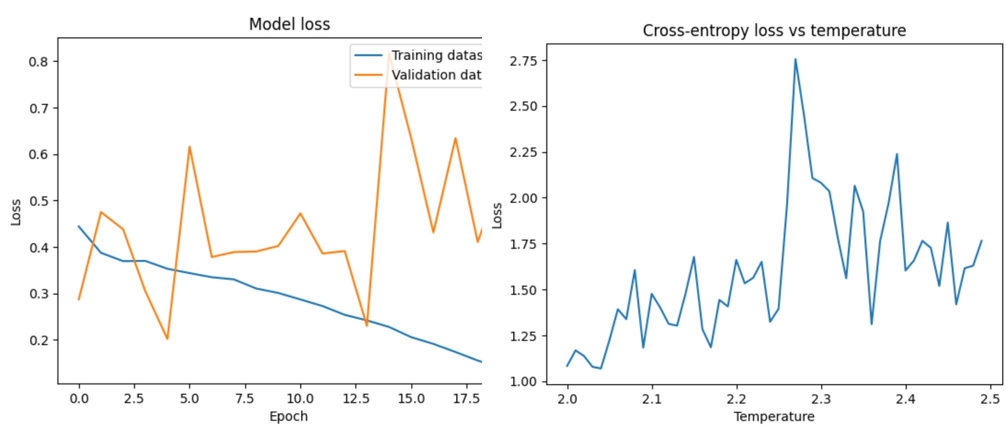
4.7 An Alternate Order Parameter

We decided to make use of testing loss as a function of temperature to check the presence of the phase transition. For our baseline classifier model, we had used the cross-entropy loss function for our model, labelling snapshots at temperatures below critical temperatures as 1 and the ones above as 0. Earlier, we saw the relation between the cross-entropy loss function and entropy for the 2d Ising model. Due to this relation, we expect there to be an increase in loss for snapshots that were produced near the transition point. For this experiment, we employed the following model architecture (10).



```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d (Conv2D)              (None, 48, 48, 64)        640
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 24, 24, 64)        0
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 22, 22, 128)       73856
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 11, 11, 128)       0
_________________________________________________________________
flatten (Flatten)            (None, 15488)             0
_________________________________________________________________
dense (Dense)                (None, 128)               1982592
_________________________________________________________________
dense_1 (Dense)              (None, 1)                 129
=================================================================
Total params: 2,057,217
Trainable params: 2,057,217
Non-trainable params: 0
_________________________________________________________________
```

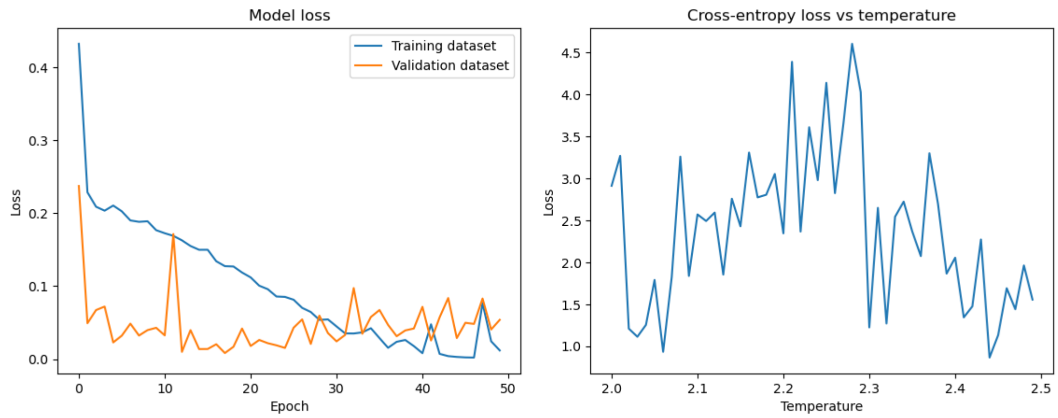
Figure 10: Model architecture for CNN classifier on $L = 50$ lattice



(a) (b)

Figure 11: Model loss and Loss vs Temperature for $L = 20$ with Sequential CNN

As we can see, there is a peak near the critical temperature when testing loss is observed across temperatures in the test dataset.



(a) (b)

Figure 12: Model loss and Loss vs Temperature for $L = 50$ with Sequential CNN

5 Further plans

1. To obtain the weight matrix as a function of temperature as seen in [5] after clearing up bugs faced while running the algorithm
2. To run the model for larger lattice sizes to confirm that the observed trends in our order parameter can be extended for bigger lattice sizes.
3. Use ML to predict when equilibrium is reached while producing snapshots in the 2D Ising Model.

References

[1] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[2] D. Bennett. Numerical solutions to the ising model using the metropolis algorithm. 2016.

[3] P. Broecker, J. Carrasquilla, R. G. Melko, and S. Trebst. Machine learning quantum phases of matter beyond the fermion sign problem. *Scientific Reports*, 7(1):8823, Aug. 2017.

[4] J. Carrasquilla and R. G. Melko. Machine learning phases of matter. *Nature Physics*, 13(5):431–434, feb 2017.

[5] A. Tanaka and A. Tomiya. Detection of Phase Transition via Convolutional Neural Networks. *Journal of the Physical Society of Japan*, 86(6):063001, June 2017. Publisher: The Physical Society of Japan.