Estimating the Critical Temperature of the Two-Dimensional Five-State Potts Model through Monte Carlo Simulation and Machine Learning

Yazan Almajnouni
MIT Physics Department, 77 Massachusetts Ave, Cambridge, MA 02139-4307
(Dated: May 21, 2023)

We investigate the dimensionless critical temperature of the 2D five state Potts model using Monte Carlo simulations and using a neural network. We obtained $T_c = 0.848 \pm 0.007$ from the Monte Carlo approach and $T_c = 0.82 \pm 0.04$ from the neural network. Both are consistent with the theoretical result.

I. INTRODUCTION

In statistical mechanics, phase transitions - abrupt changes in the physical state of matter induced by variations in external parameters - play a pivotal role in understanding a diverse range of physical phenomena, from boiling water to magnetization in ferromagnetic materials. At the heart of understanding phase transitions lies the concept of critical temperature, the distinct temperature at which a substance transitions between different phases. There are only few systems where the exact critical temperature can be derived analytically, which is why numerical methods are often used to estimate it.

Monte Carlo simulations serve as one of the most prevalent numerical methods for studying phase transitions and estimating critical temperatures in such systems. They offer a probabilistic technique for generating samples in complex multi-dimensional spaces, thereby enabling a statistical examination of the model's behavior under varied conditions. Monte Carlo simulations have been extensively used to study systems such as the Ising model and Potts model, yielding valuable insights into the nature of phase transitions. Despite their efficacy, these simulations can be computationally expensive, especially for higher-dimensional or more complex models.

Another approach that has been under investigation recently is the utilization in machine learning and neural networks to find the critical temperature. As demonstrated in this paper, a simple neural network can be trained to accurately classify spin configurations of the Ising model yielding an accurate estimate for the critical temperature and the critical exponent ν. One issue with this approach is that training the network requires prior knowledge of the critical temperature in order to assign labels to the training data.

In this paper, we use a Monte Carlo simulation to find the critical temperature of the two-dimensional five-state Potts model. We also follow the approach outlined in this paper[1] to train a neural network to classify configurations of the Potts model and obtain the critical temperature. The advantage of this approach is that training is only done using the theoretical ground states of the model being studied, which means no prior knowledge of the theoretical critical temperature is needed. This also means that the training data can be obtained very cheaply, especially for systems with simple ground states. We compare the results obtained from these two approached with the analytical critical temperature of the Potts model.

II. THE POTTS MODEL

The model we study in this paper is the Potts model, which is a generalization of the famous Ising model. The Potts model has the following Hamiltonian:

$$\beta H = -J \sum_{<i,j>} \delta_{s_i, s_j} \tag{1}$$

Where $\beta = 1/k_b T$ is the inverse temperature, $< i, j >$ indicates that the sum is over nearest neighbors, δ is the Kronecker-delta function, and s_i are the Potts spin variables which take integer values in $\{0, 1, 2, ..., q-1\}$.

It's well known that the dimensionless critical temperature for the two-dimensional q-state Potts model is given by: $T_c = \frac{1}{\ln(1+\sqrt{q})}$ in the thermodynamic limit [2]. For the five-state model we study in this paper, $T_c \approx 0.852$. We choose to study the Potts model because the theoretical critical temperature is known exactly, which makes it possible to check our results.

III. MONTE CARLO APPROACH

In this paper, we study lattices with dimensions $L \times L$ where $L \in \{8, 12, 16, 20, 24, 28, 32\}$. We simulate the lattices at 100 equally spaced temperatures in the range $[0.5, 1.1]$. We generated 30000 configurations using the Swendsen-Wang algorithm as follows [5]:

1. Start with a configuration of the spins in the Potts model. In the q-state Potts model, each spin can be in one of q different states.

2. Form clusters of neighboring spins that are in the same state. The rule for whether or not to connect two neighboring spins into the same cluster is as follows: If the spins are in the same state, connect them with a probability $p = 1 - \exp(-1/T)$. If the spins are not in the same state, do not connect them.

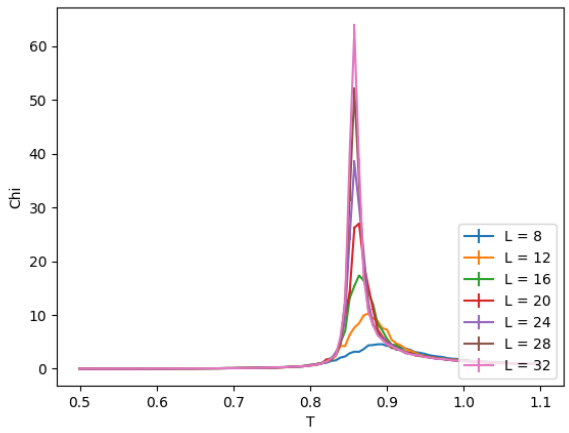

FIG. 1: Susceptibility as a function of temperature for different lattice sizes. Error bars are obtained by statistical bootstrap resampling

3. Once the clusters are formed, choose a new state at random for all the spins in each cluster. This change is done all at once, not spin by spin.

4. Repeat steps 2 and 3 for a large number of times. This allows the system to sample configurations from the correct distribution.

We allow the system to reach an equilibrium state by running the algorithm for 5000 steps before we start recording the configurations. Since the configurations generated this way are highly correlated, we only use one in every 100 generated configurations. The total number of generated configurations is 30000.

We use the fact that the magnetic susceptibility diverges at the critical temperature to estimate the critical temperature by recording the temperature at which the magnetic susceptibility is maximized for each lattice dimension. The magnetic susceptibility is given by

$$\chi = \frac{N(\langle M^2\rangle - \langle M\rangle^2)}{T} \tag{2}$$

Where N is the number of spins and M is the magnetization. The magnetization for the Potts model can be defined in multiple ways. We chose the following definition:

$$M = |\frac{1}{L^2}\sum_j \exp{(i\frac{2\pi s_i}{q})}| \tag{3}$$

Plots for the susceptibility as a function of temperature for various lattice sizes are shown in figure 1. We confirm that the susceptibility spikes up at the transition temperature.

Since our simulations are on a finite lattice, our results for transition temperatures are larger than the true critical temperature. To find the critical temperature, we make use of finite scaling and make a least squares fit of our results to the equation:

$$T(L) = T_c(1 + (x_0/L)^{1/\nu}) \tag{4}$$

Doing so, we get $T_c = 0.848 \pm 0.007$, which is consistent with the theoretical result. Note that in principle, the critical exponent ν can also be found this way, but our data is insufficient to make an accurate determination. This is possibly because of the narrow range of lattice sizes we consider.

IV. NEURAL NETWORK APPROACH

We follow the approach outlined in these two papers[1] [4]. We train the network to classify the q different theoretical ground states of the Potts model. The classification is given as q-long vector where each component represents the probability that the given configuration is in the corresponding state. The training is done using number of copies (200 in our case) of each of the q ground states.

The ides behind this approach is that configurations below t he critical temperature will be similar to one of the q ground states, and hence the magnitude of the output vector will be close to 1. However, configurations above the critical temperature consist of spins in random states, which leads the network to predict a more or less equal likelihood to being in all q classes. So the magnitude of the output vector will drop down to $\frac{1}{\sqrt{q}}$ as illustrated in figure 2. In the figure, The first lattice is in the first ground state so only the first component of the output is large. The second lattice is close to the second ground state so the second component in the output vector is large. The third lattice is above the critical temperature so the components of the output vector are almost equal.

We use TensorFlow and Keras to build and train our neural network[6][7]. Our neural network consists of an input layer, followed by a hidden layer with 512 nodes, followed by an output layer as illustrated in figure 3. The activation function of the hidden layer is ReLU and the for the output layer is softmax. We use the minibatch algorithm and the Adam [8] optimizer to train the network. Our loss function is categorical cross entropy. We also use L_2 regularization to avoid overfitting to training data. Since there is no intrinsic ordering to the states, we use a one-hot encoding to map each spin variable to a q-dimensional vector.

After training, we run the model on configurations generated using the same Swendsen-Wang algorithm described in the previous section. Due to memory and time limitations, we had to use a smaller data set for this ap-

$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \Rightarrow \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \quad \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \Rightarrow \begin{bmatrix} 0.02 \\ 0.98 \\ 0 \end{bmatrix} \quad \begin{bmatrix} 0 & 1 & 2 \\ 0 & 2 & 1 \\ 1 & 2 & 0 \end{bmatrix} \Rightarrow \begin{bmatrix} 0.31 \\ 0.36 \\ 0.33 \end{bmatrix}$$

FIG. 2: Matrices on the left represent the lattice configurations and the vectors on the right represent the output of the neural network.

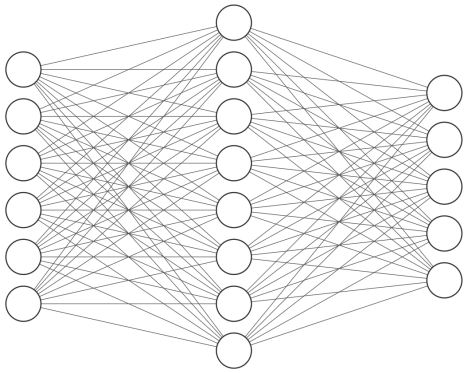

FIG. 3: Schematic of the neural network we use. The first layer has L^2 nodes, the hidden layer has 512 nodes, and the output layer has $q = 5$ nodes.

proach. We use the same values for lattice size and temperatures as in the Monte Carlo simulation, but We only generated 3000 configurations for each (T, L) pair and we used one in every 10 configurations. We calculate the norm of the output vector for each configuration and we plot the average of these norms (denoted as R) as a function of temperature for each lattice size L in figure 4.

Compared to the plots in the cited papers, our data is considerably more noisy which is likely due to our smaller data set. We smoothed out the plots above by applying a savgol filter on the data.

We define the transition temperature in this approach as the temperature that maximize the absolute value of the derivative $\frac{dR}{dT}$ calculated numerically using the formula:

$$|\frac{dR}{dT}| = |\frac{R(T + \delta T) - R(T - \delta T)}{2\delta T}| \tag{5}$$

Similarly to the Monte Carlo approach, we fit our results to equation (4) and we obtain $T_c = 0.82 \pm 0.04$. This also consistent with the theoretical result, but the uncertainty is significantly larger. This is once again likely due to the smaller data set. This approach also allows us in principle to obtain ν, but more data is needed to get an accurate estimate.

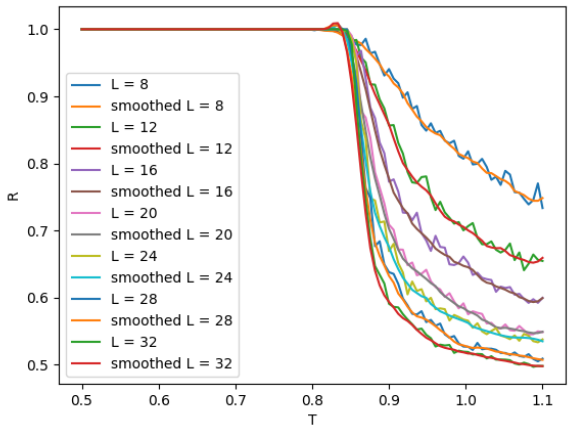

FIG. 4: Plot of R as a function of T for various lattice sizes. Both the raw data and the smoothed data are included.

V. CONCLUSIONS

We estimated the critical temperature of the 2D five-state Potts model using Monte Carlo simulations and using neural networks. We found that both approaches can be used to obtain reasonably accurate results even for a relatively small data set. We also demonstrated the effectiveness of training the network using ground states only, which provides a numerically efficient way of finding the critical temperature. More research is needed to determine the efficacy of the neural network approach to determine critical exponents.

[1] Li, C., Tan, D. R., Jiang, F. (2018). Applications of neural networks to the studies of phase transitions of two-dimensional Potts models. Annals of Physics, 391, 312-331. https://doi.org/10.1016/j.aop.2018.02.018

[2] Beffara, V. (2010, June 25). The self-dual point of the two-dimensional random-cluster model is critical for $q \geq 1$. arXiv.org. https://arxiv.org/abs/1006.5073

[3] Sandvik, A. W. (2010). Computational Studies of Quantum Spin Systems. In Nucleation and Atmospheric Aerosols. American Institute of Physics. https://doi.org/10.1063/1.3518900

[4] Tan, D. R., De Li, C., Zhu, W., Jiang, F. (2020). A comprehensive neural networks study of the phase transitions of Potts model. New Journal of Physics, 22(6), 063016. https://doi.org/10.1088/1367-2630/ab8ab4

[5] Swendsen, R. H., Wang, J. (1987). Nonuniversal critical dynamics in Monte Carlo simulations. Physical Review Letters, 58(2), 86–88.

https://doi.org/10.1103/physrevlett.58.86
[6] https://www.tensorflow.org
[7] https://keras.io
[8] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017.