May 11, 2024

Machine Learning and Monte Carlo Simulation for phase diagram of interacting Josephson qubits

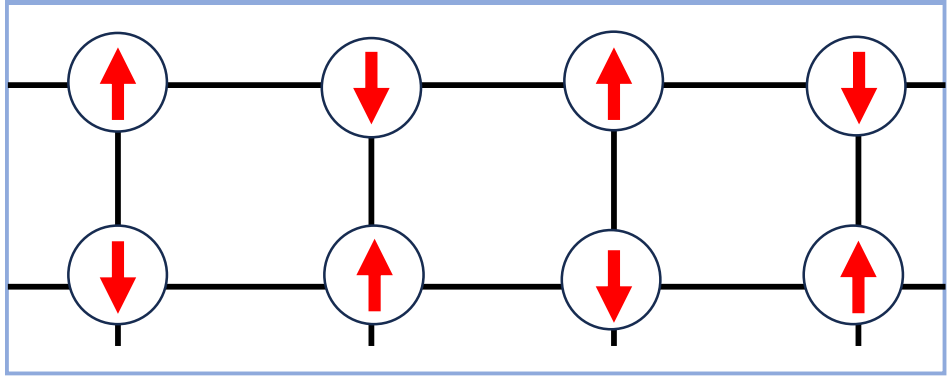
Pratyush Jha (B. Tech Engineering Physics)
Sudatt Kayal (B. Tech Engineering Physics)

Supervisor : Prof. Peayush Choubey

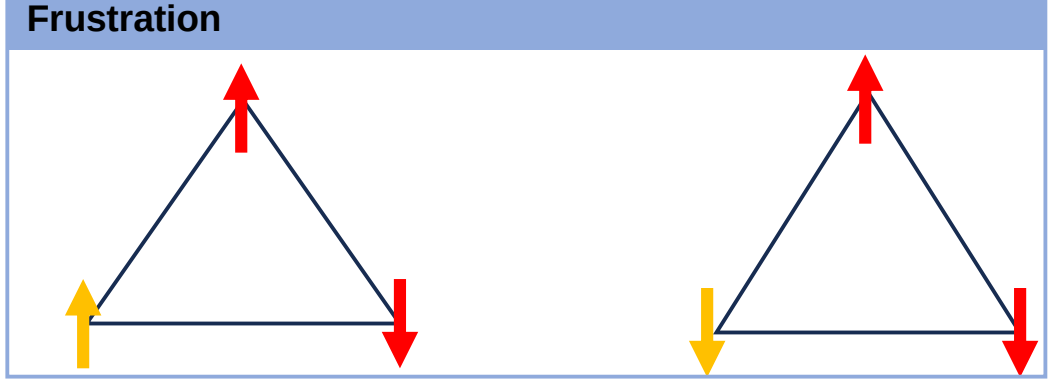
Geometric frustration:

The collective behavior of the low-energy magnetic excitations crucially depends on the geometry of the lattice they inhabit.

Antiferromagnetic state in bipartite (square) lattice



Magnetic ordering in a triangular lattice: Geometric Frustration



- Natural solid-state systems demonstrating interesting behavior due to underlying frustration includes iron-based superconductors, Kagome Magnets, frustrated ferromagnetic chains.
- Currently, artificially prepared systems have drawn special attention.
- Frustrated Josephson Junction Array (f-JJAs) is one of such system.

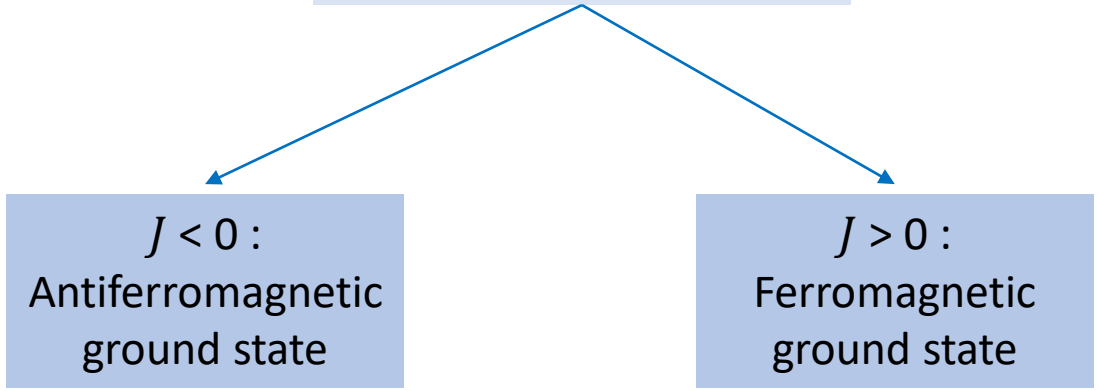
Some features of frustrated Josephson Junction Arrays:

- Current technology allows to form f-JJAs of various shape and size
- Frustration can be tuned due to external magnetic field
- Physics can be mapped to quasi 1D/2D Ising Model or XY Model spin models.
- Provides a feasible experimental platform to establish quantum analog simulations.

Source : Neyenhuys, Oliver, et al. “Long-range Ising Spins Models Emerging From Frustrated Josephson Junctions Arrays With Topological Constraints.” *Physical Review B*, vol. 108, no. 16, American Physical Society (APS), Oct. 2023. *Crossref*, https://doi.org/10.1103/physrevb.108.165413.

Short-range Interactions : Nearest Neighbour Ising Model

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



- The nearest neighbor Ising model has exact solution from which it can be shown that the exact critical temperature for an infinite system is $\frac{T_c}{J} = \frac{2}{\ln(1+\sqrt{2})} \approx 2.269$.
- Computationally, these results can be obtained from Monte Carlo methods.
- Two types of algorithms are used : The Metropolis algorithm and Clustering based algorithms.

Monte Carlo Simulation for Short Range Interaction

Goal of Monte Carlo Simulation

- To generate $\sigma(1), \sigma(2), \ldots, \sigma(K)$, where $\sigma = (\sigma_1 \sigma_2 \ldots \sigma_N)$; representing unbiased sample from Boltzmann distribution.
- $P(\sigma) \propto W_\sigma$ (Boltzmann weight at temperature T): $W_\sigma = e^{\left\{\frac{-E_\sigma}{T}\right\}}$.
- Joint probability distribution is not considered, even though sequence $\sigma(1), \sigma(2), \ldots, \sigma(K)$ can be correlated.

Metropolis Algorithm

Algorithm

do j=1,N
  $i = \mathbf{random}[1, \ldots, N]$
  if $(\mathbf{random}[0-1]) < W_{\sigma^{-i}}/W_\sigma)$ $\sigma_i = -\sigma_i$
enddo

$$P(\sigma \to \sigma^{-i}) = P_{\text{select}}(i) P_{\text{accept}}(\sigma^{-i})$$

Based on detailed balance principle:

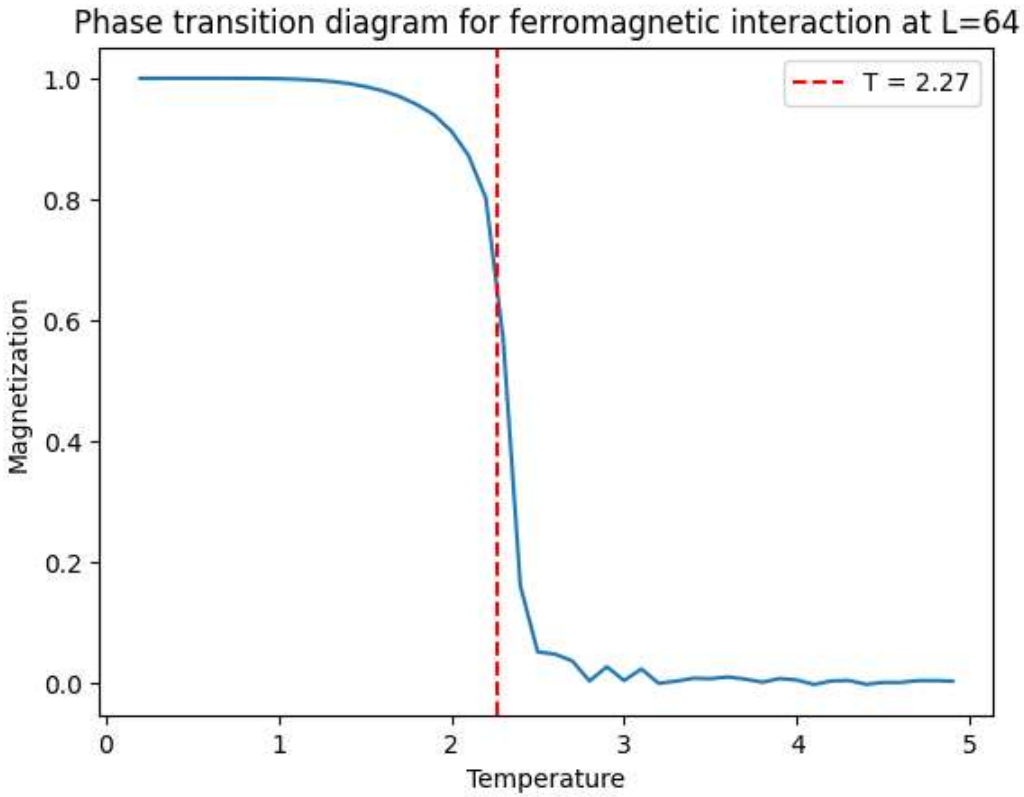
$$\frac{P(A \to B)}{P(B \to A)} = \frac{W(B)}{W(A)}$$

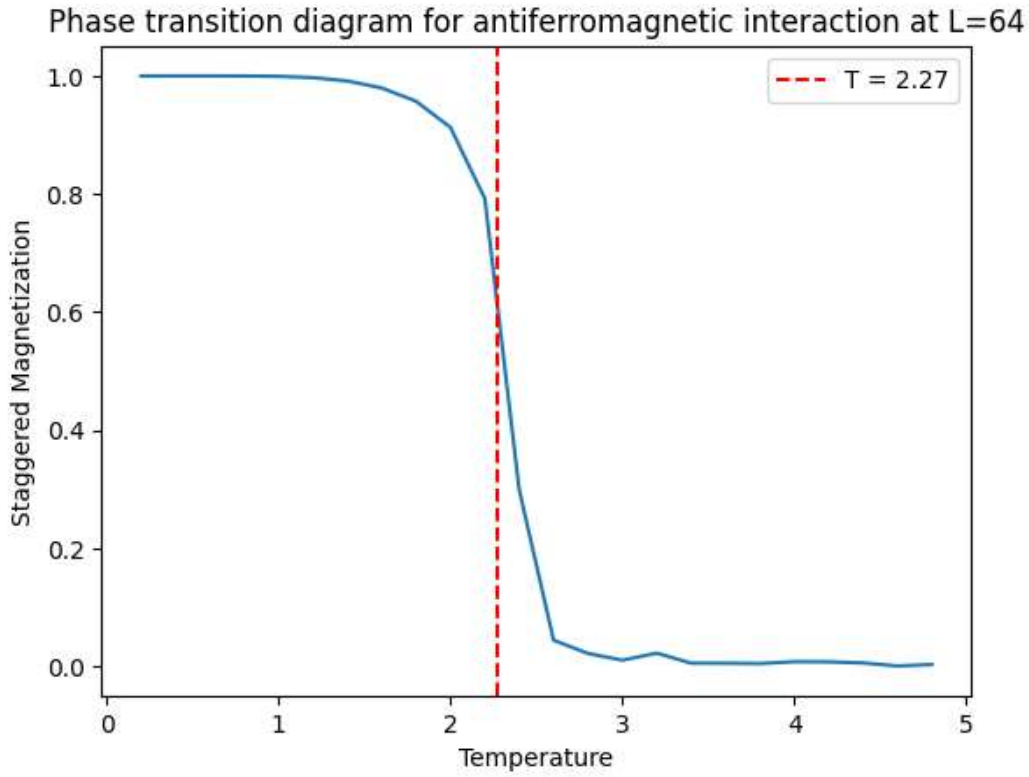
$$P_{\text{accept}}(\sigma^{-i}) = \min\left[\frac{W_{\sigma^{-i}}}{W_\sigma}, 1\right]$$

Source: Sandvik, Anders W., et al. “Computational Studies of Quantum Spin Systems.” *AIP Conference Proceedings*, AIP, 2010. *Crossref*, https://doi.org/10.1063/1.3518900.

Metropolis Algorithm's results on Nearest Neighbor Ising Model

- Temperature(T/J) ranges from 0.2 to 5 at gaps of 0.2.
- At each temperature value, 1000 Monte Carlo spin flip attempts were made for each lattice site.
- Staggered magnetization was used as the order parameter for antiferromagnetic lattices.

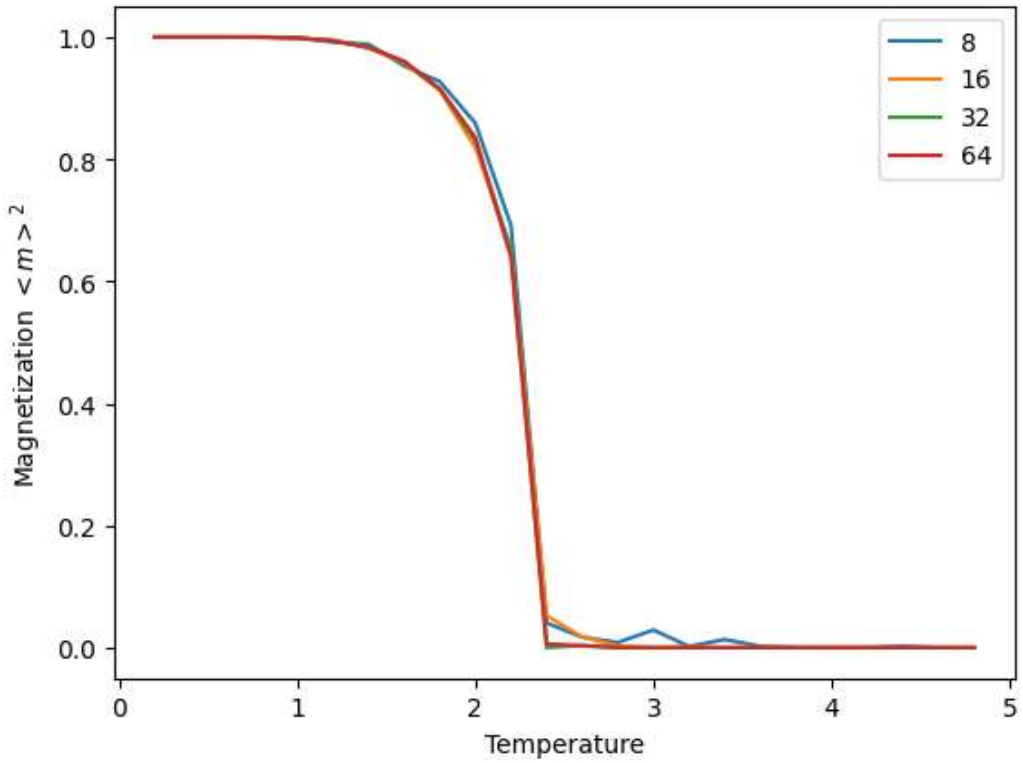




Metropolis Algorithm's results on Nearest Neighbor Ising Model

- Temperature (T/J) ranges from 0.2 to 5 at gaps of 0.2.
- At each temperature value, 1000 Monte Carlo spin flip attempts were made for each lattice site.
- As the lattice size increases, the closer the simulation result gets towards actual results.

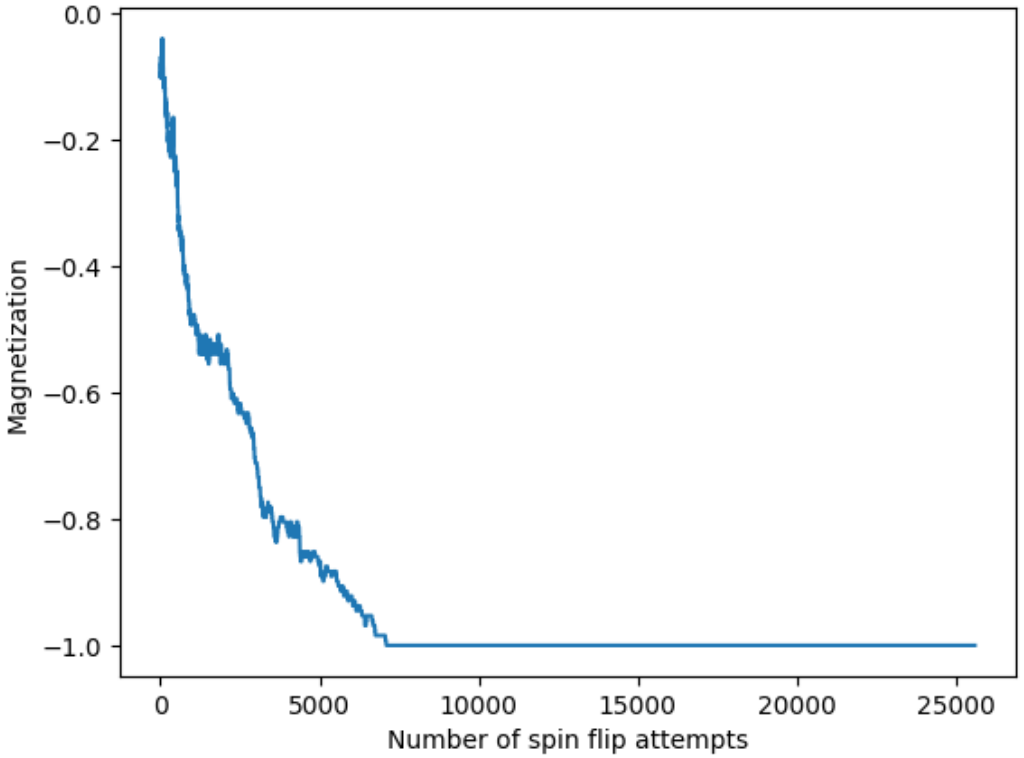
Phase transition diagram for ferromagnetic interactions at different values of L



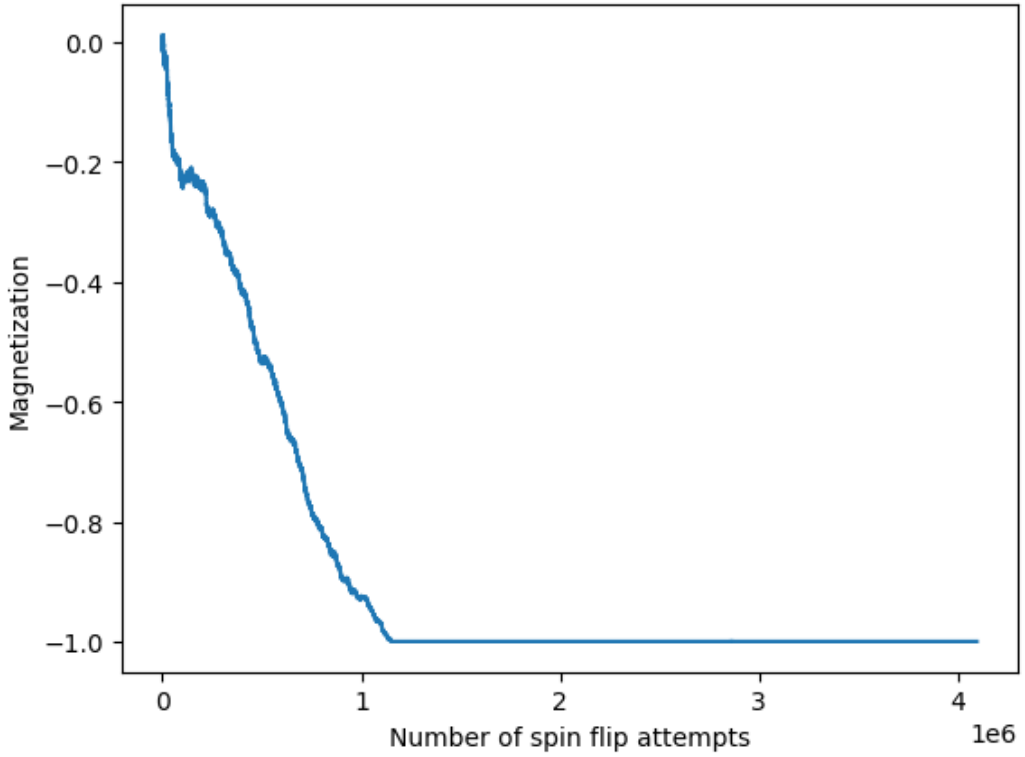
Equilibration Time, Autocorrelation and Critical Slowing Down

- To converge to a stationary distribution, many iterations are required before obtaining samples. The time required to wait for the system to reach equilibrium is referred to as equilibration time.
- Metropolis algorithm require a large equilibration time which increases with the lattice size.
- Using a single lattice and evolving it over the temperature range resolves this issue of slower equilibration.
- Better alternative is to start with a configuration which is closer to the equilibrium configuration at each temperature range.

Equilibration time for antiferromagnets for L = 16 at T < Tc



Equilibration time for antiferromagnets for L = 64 at T < Tc



Equilibration Time, Autocorrelation and Critical Slowing Down

- Metropolis algorithm generates samples which follow the Markov property:

$$P(X_n = x_n \mid X_{n-1} = x_{n-1}, X_{n-2} = x_{n-2}, \dots, X_0 = x_0) = P(X_n = x_n \mid X_{n-1} = x_{n-1})$$

- This leads to generated samples being highly correlated.
- Autocorrelation time is defined as the time required for the samples to get decorrelated from the current sample.
- Metropolis algorithm has high autocorrelation time.
- Autocorrelation time further increases near the critical temperatures. The correlation time at this range scales with the lattice dimension $\tau \sim L^z$, where z is known as dynamic critical exponent which has the value of 2.125 for 2D Ising Model [**3**].
- This leads to critical slowing down of the system dynamics.

3:Simulation of Monte Carlo Methods. aiichironakano.github.io/phys516/01-2Ising.pdf.

Cluster Based Updating: The Swendsen-Wang Algorithm

- Clustering algorithms update the configuration on length scale making it more relevant to the physics of the problem.
- They have lower equilibration time as well as lower autocorrelation time.
- Swendsen and Wang, in 1987, proposed the first cluster algorithm for the Ising model **[4]**.

Swendsen-Wang Algorithm:

Input: Current Ising configuration, $K \equiv \frac{J}{kT}$.

for every pair of interacting spins s_i and s_j **do**

Place an edge connecting them with probability $\delta_{s_i s_j}(1 - e^{-2Kp})$

end for

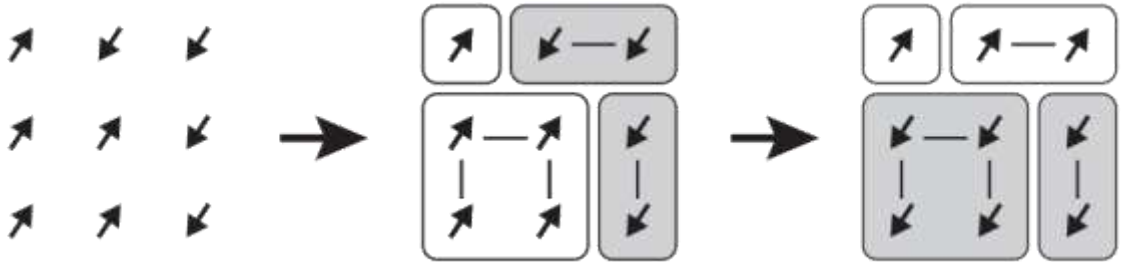
for every cluster of connected spins in the graph **do**

With probability ½, flip all the spins in the cluster;

end for

return the updated configuration.

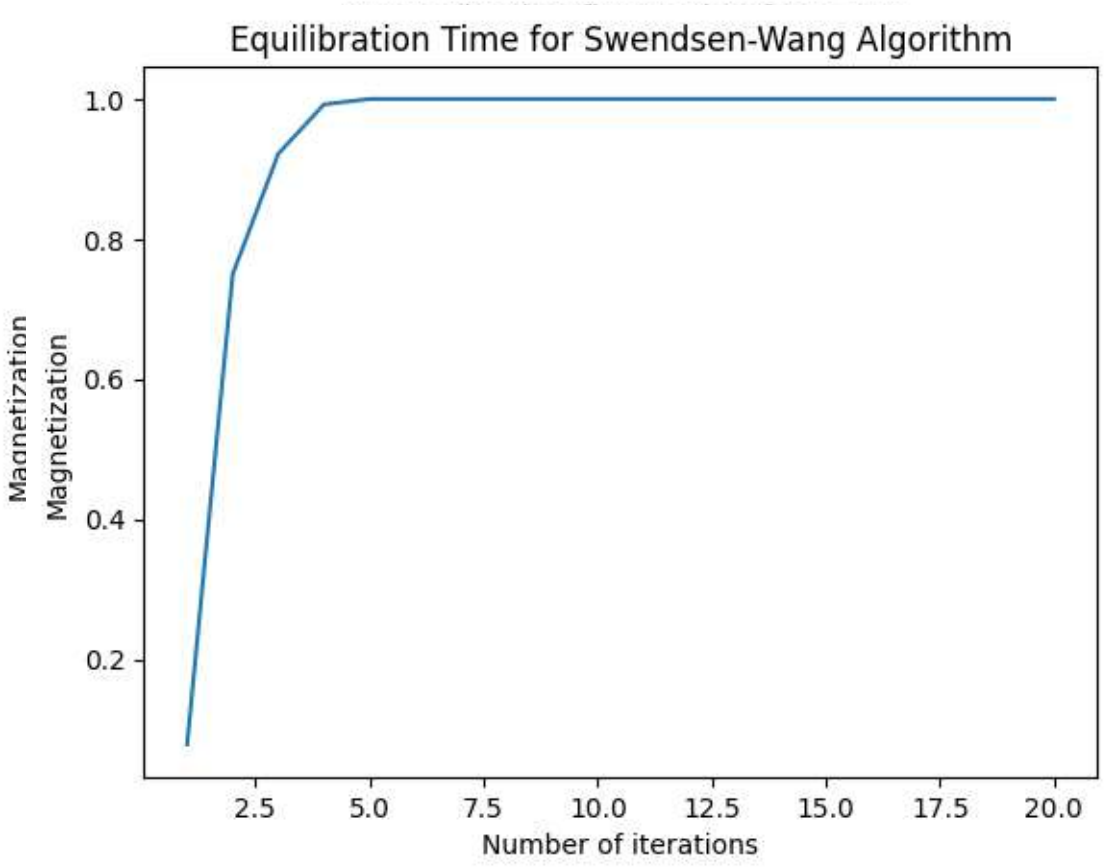
A cycle in Swendsen-Wang cluster update of the Ising model

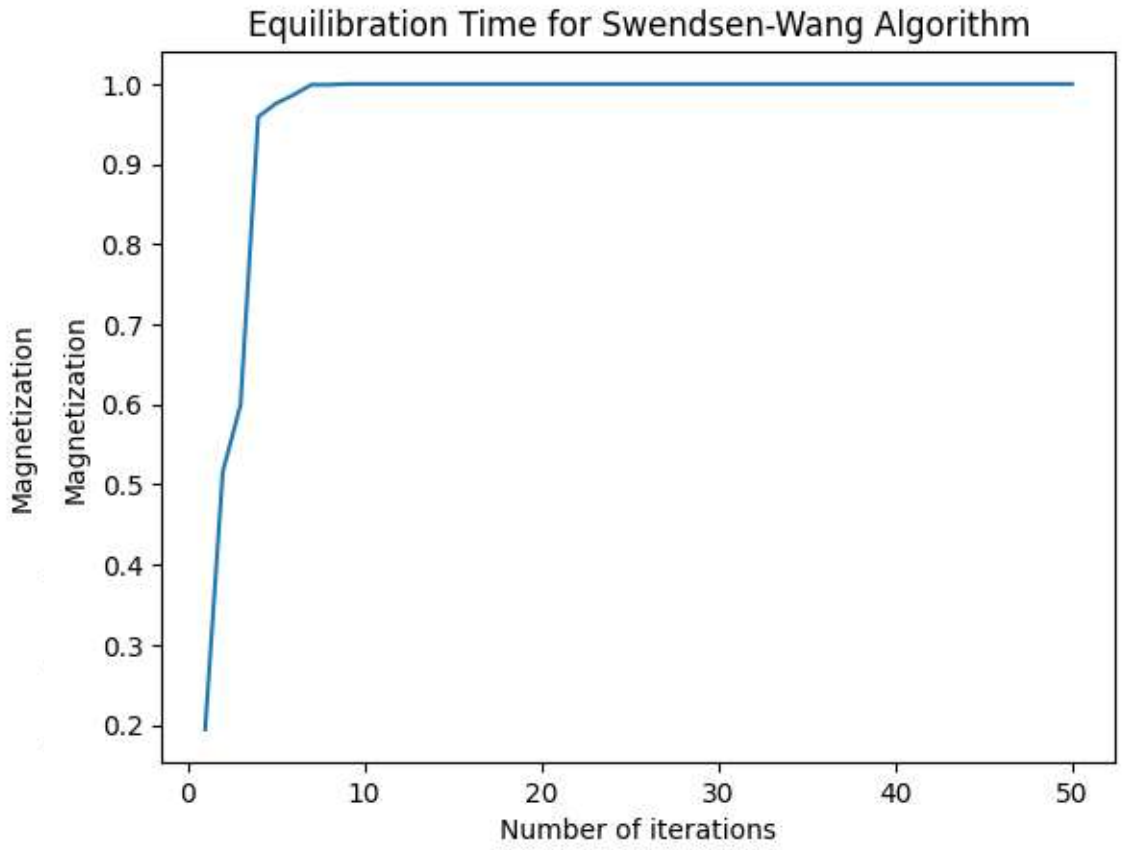


4. R. H. Swendsen and J.-S. Wang, "Nonuniversal critical dynamics in Monte Carlo simulations," Physical Review Letters, vol. 58, no. 2, pp. 86 (1987).
Source of figure: Gubernatis, N. Kawashima, and P. Werner, Quantum Monte Carlo Methods. Cambridge University Press (2016).

Swendsen-Wang Algorithm applied on the Ising Model

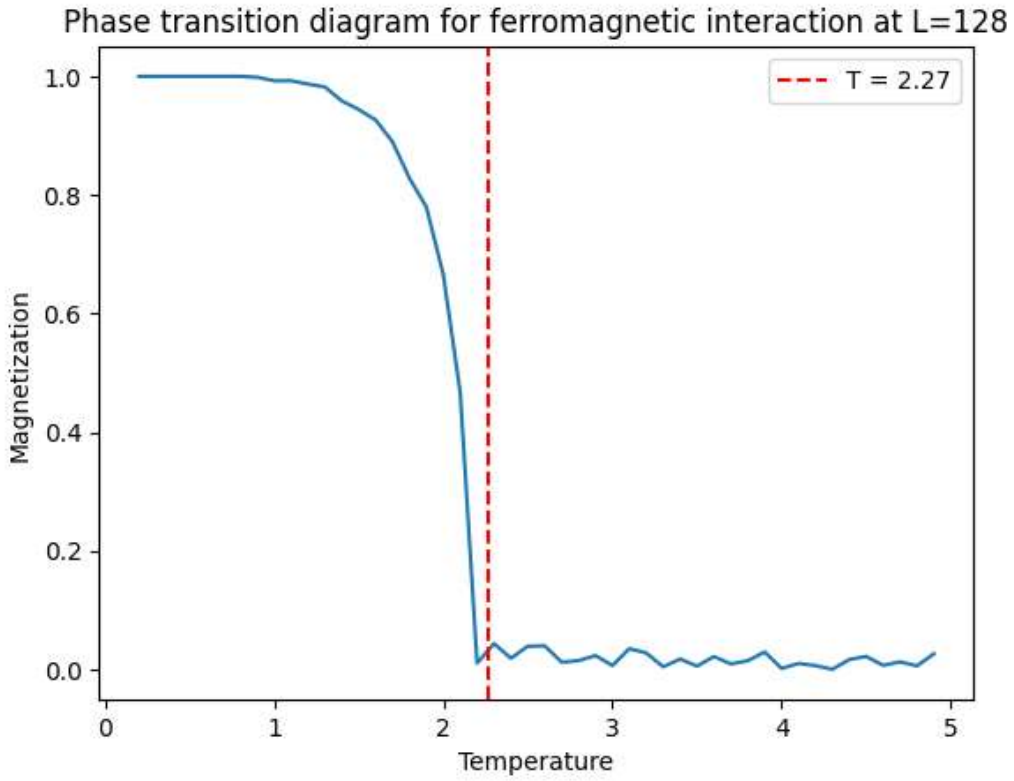
- Equilibration time is very less for Swendsen-Wang algorithm making them useful for faster generation of data compared to Metropolis algorithm.

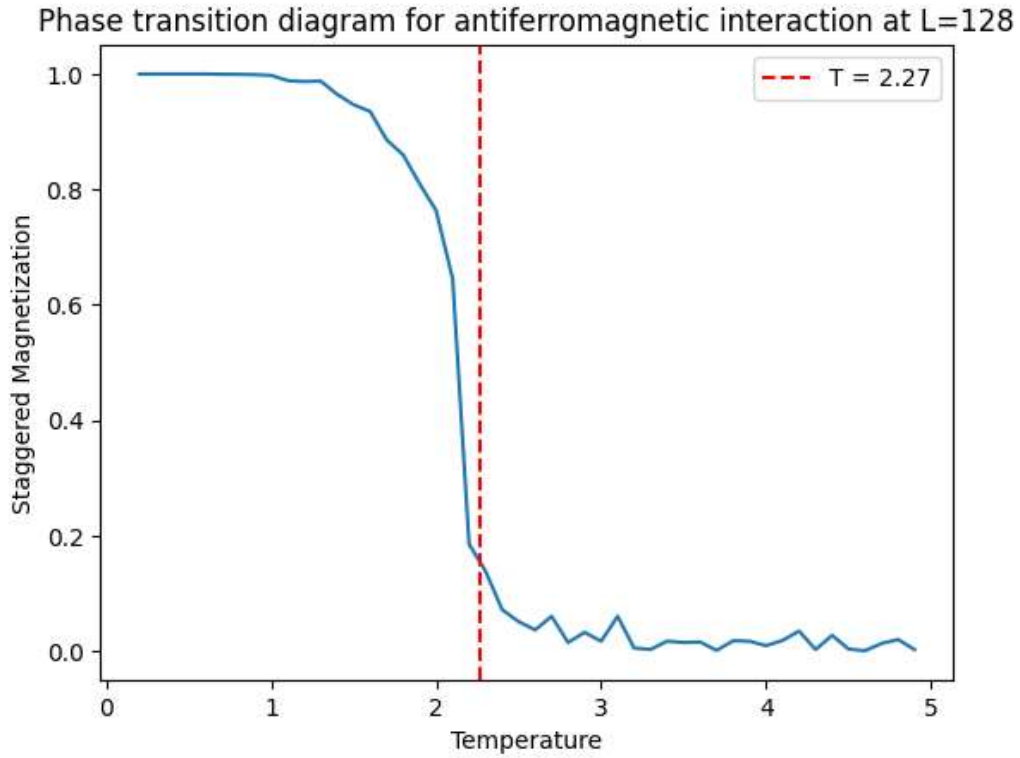




Swendsen-Wang Algorithm applied on the Ising Model

- Following phase transition diagram is obtained for ferromagnetic and antiferromagnetic interactions at L = 128. The results indicate that the algorithm performs quite well on the nearest neighbor Ising model.
- The following plot is obtained for just 4 cluster flip attempts per temperature value with initial lattice randomly initialized (unlike Metropolis).

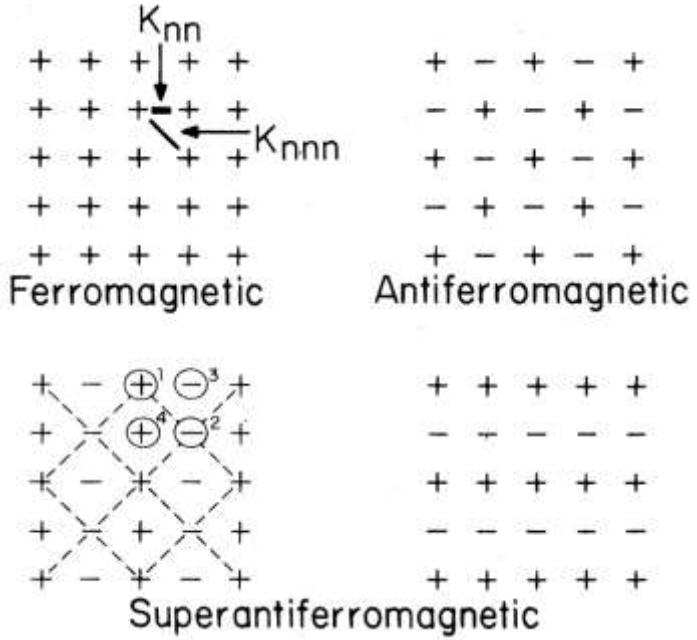




One step closer to long-range interactions: Next Nearest Neighbor Interaction

$$H = J_{nn} \sum_{\text{nn pairs}} s_i s_j + J_{nnn} \sum_{nnn\ pairs} s_i s_k$$

- $R = \frac{J_{\text{nnn}}}{J_{\text{nn}}}$ is defined as the coupling ratio between next nearest neighbors and nearest neighbors' interaction.
- There are three possible ground states for NNN Ising lattice: Ferromagnetic, Antiferromagnetic and Super Antiferromagnetic (SAF) phase.
- For R < 0.5, antiferromagnetic state is the ground state.
- For R > 0.5, the system breaks into two pairs of interpenetrating antiferromagnetic lattice, this is known as super-antiferromagnetic phase.

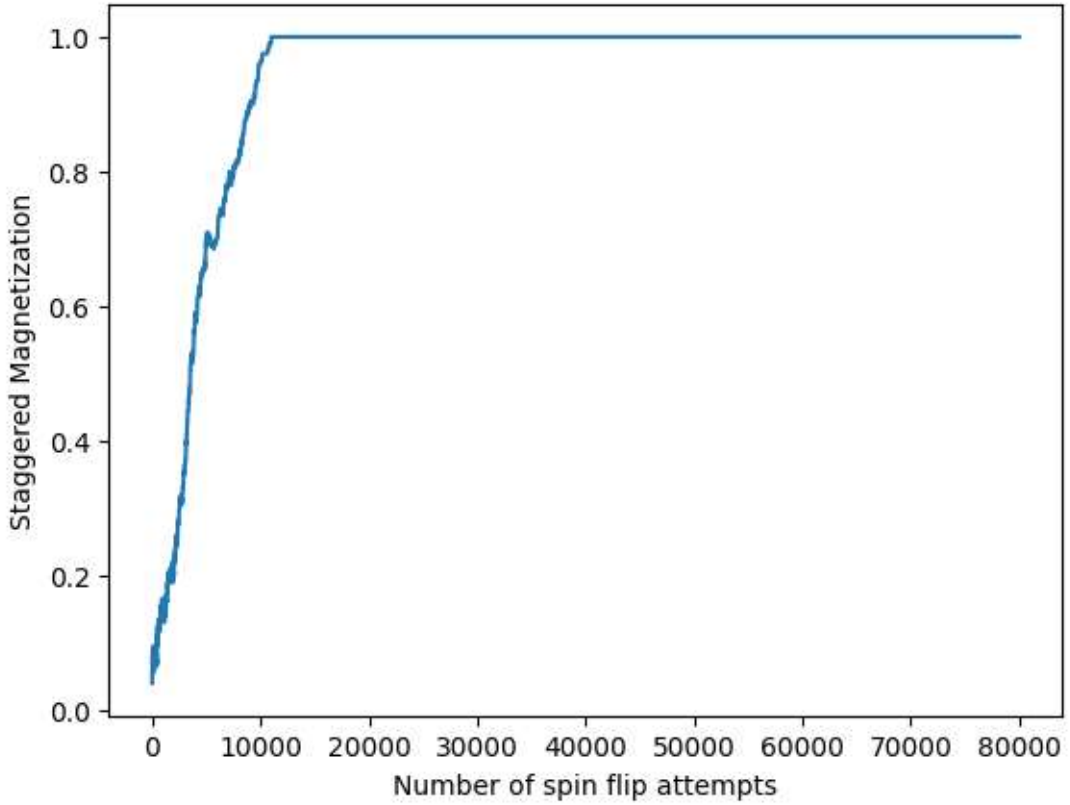


Source of figure: D. Landau, "Phase transitions in the Ising square lattice with next-nearest-neighbor interactions," Physical Review B, vol. 21, no. 3, pp. 1285 (1980).

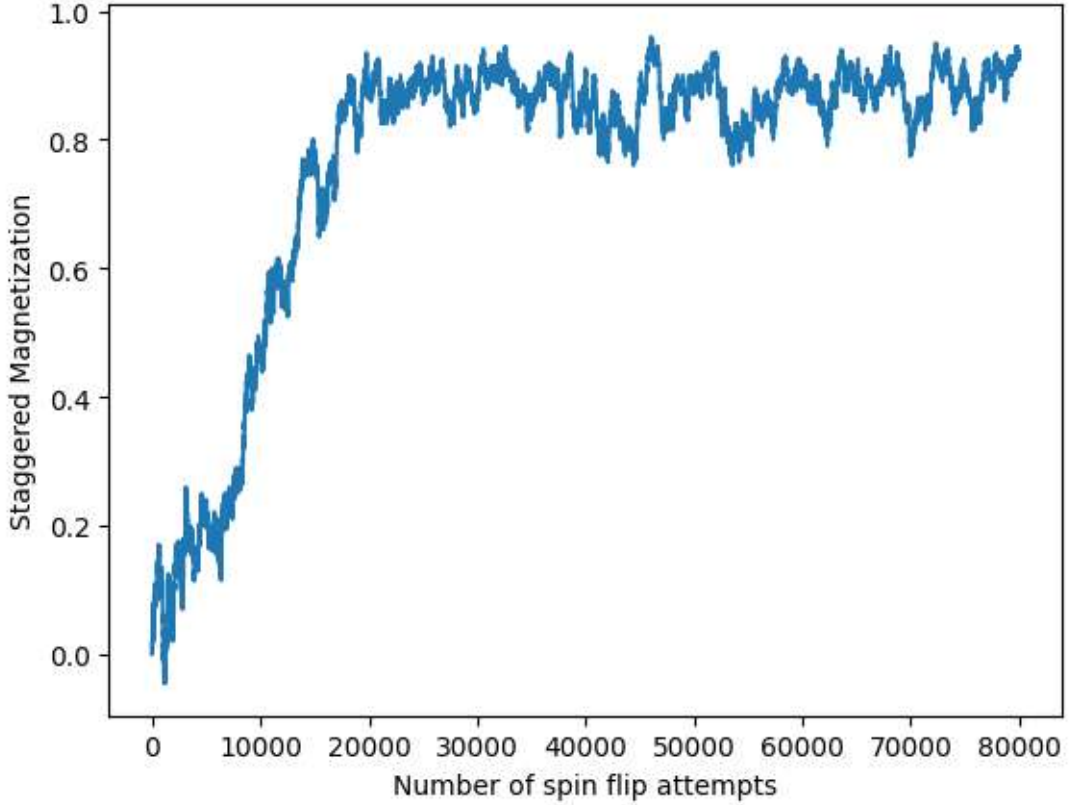
Results of Metropolis on NNN Ising Model for R<0.5

- Due to high equilibration time starting configuration for R<0.5 is taken to be a perfect antiferromagnetic lattice.

Equilibration time for R = -2 for L = 16 at T < Tc

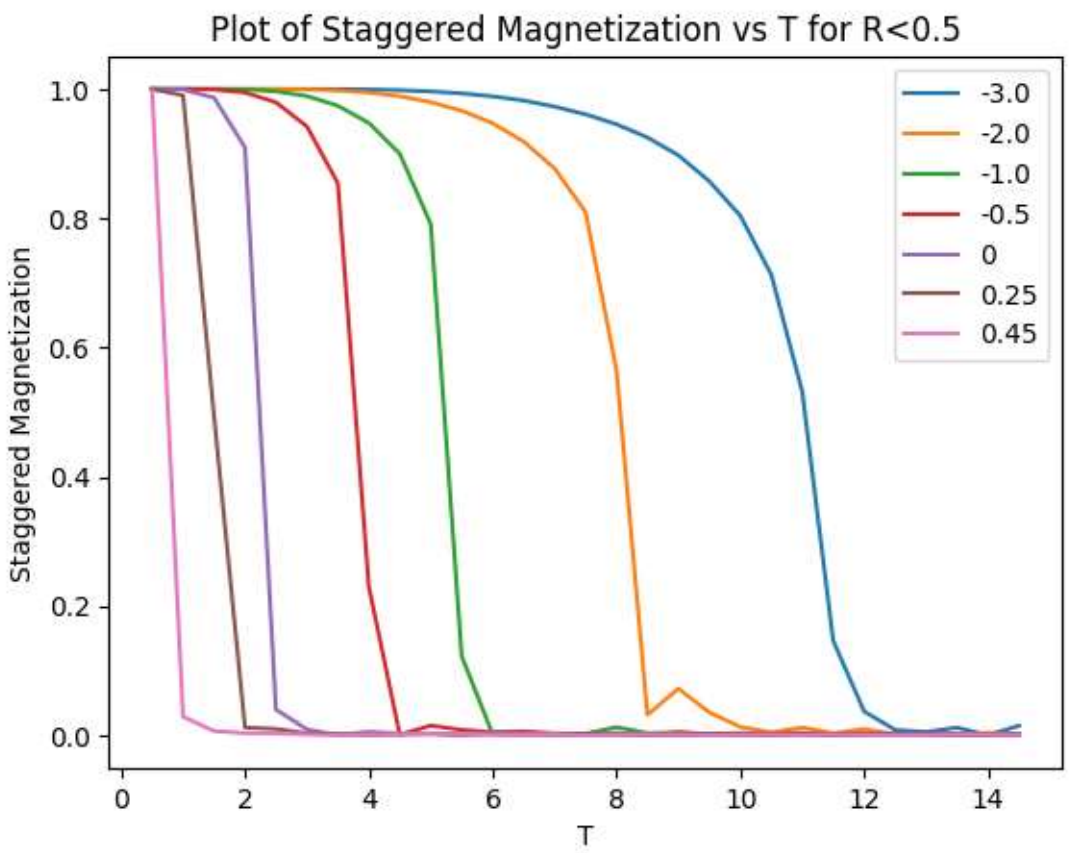


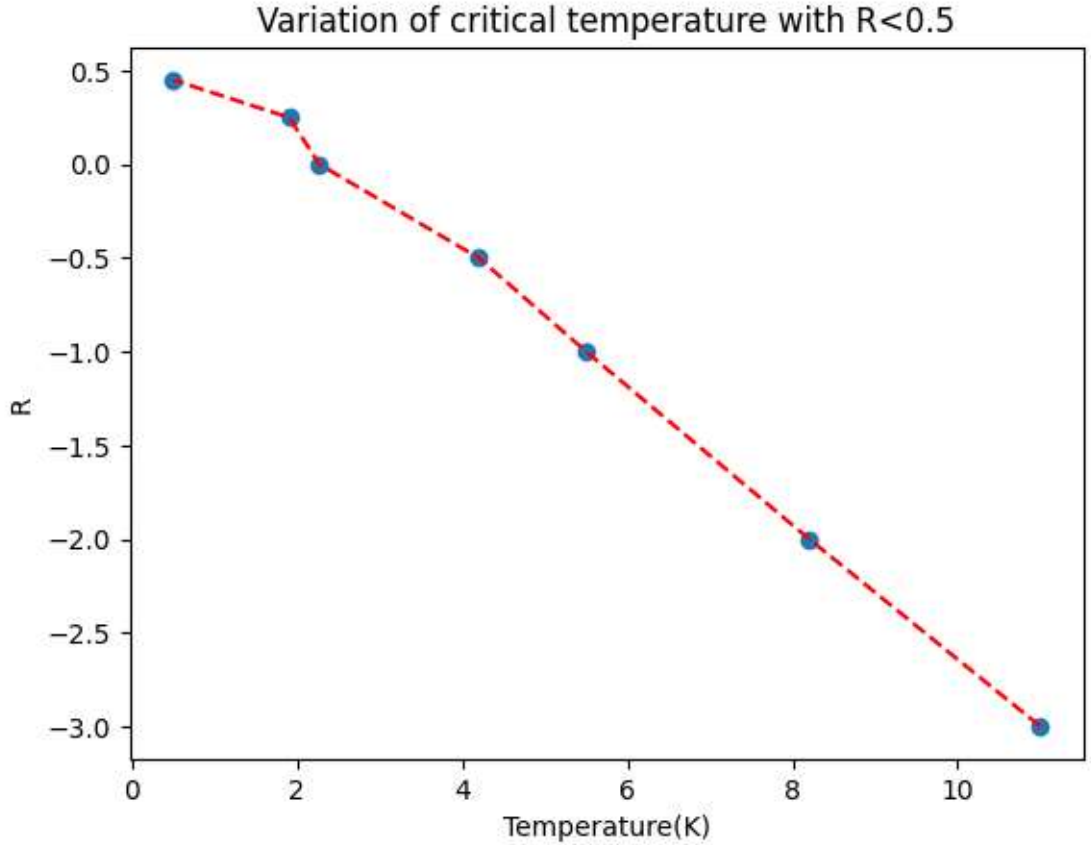
Equilibration time for R = -2 for L = 16 at higher temperature



Results of Metropolis on NNN Ising for R<0.5

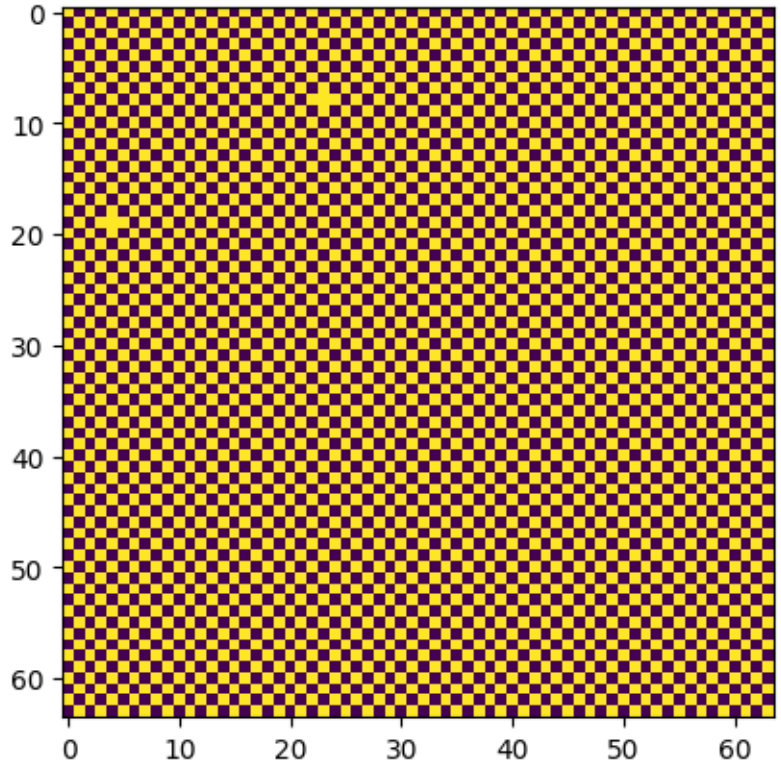
- Critical temperature increases with increasing magnitude of R (for R<0.5).



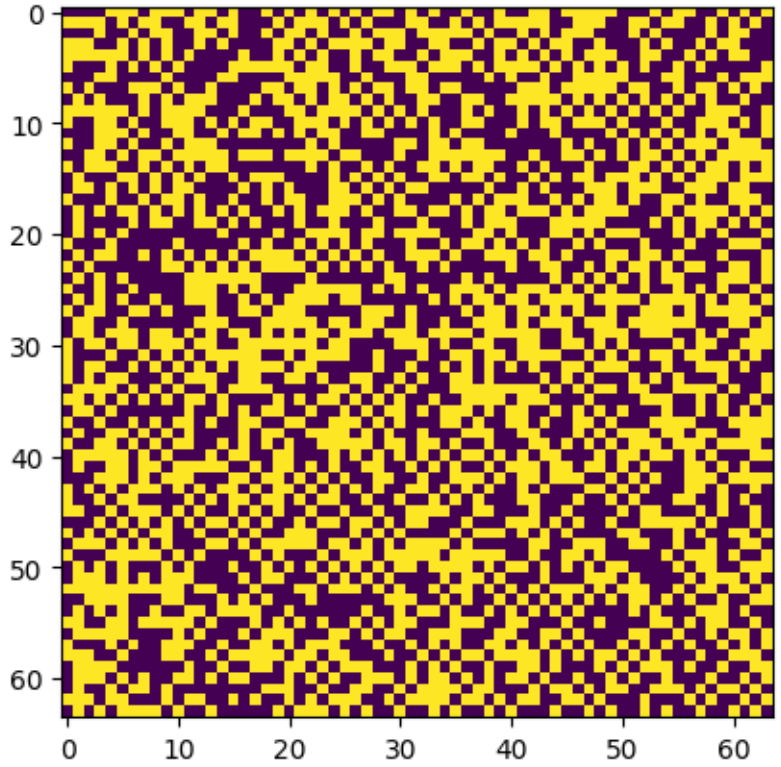


Results of Metropolis on NNN Ising for R<0.5

- Metropolis works perfectly even for NNN interactions. The following figure shows the snapshot of simulation at two different temperatures for R = -2.



Antiferromagnetic Phase obtained from simulation at T = 2.5, for R = -2

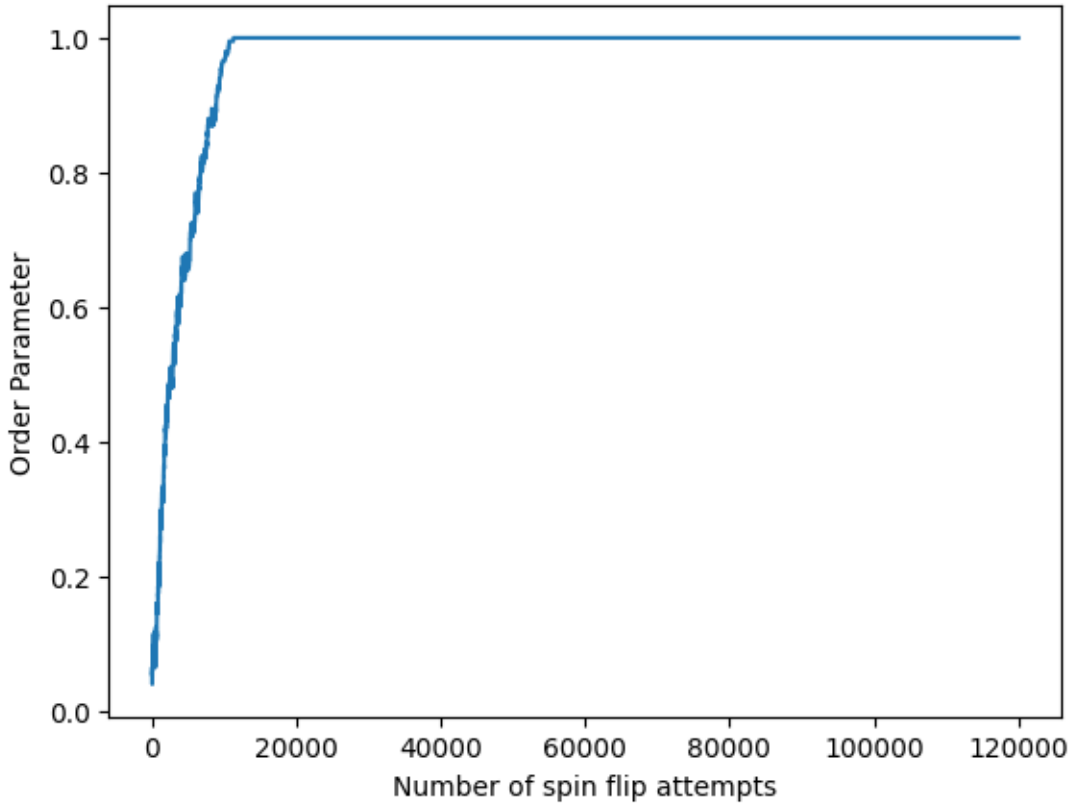


Paramagnetic Phase obtained from simulation at T = 8.5, for R = -2

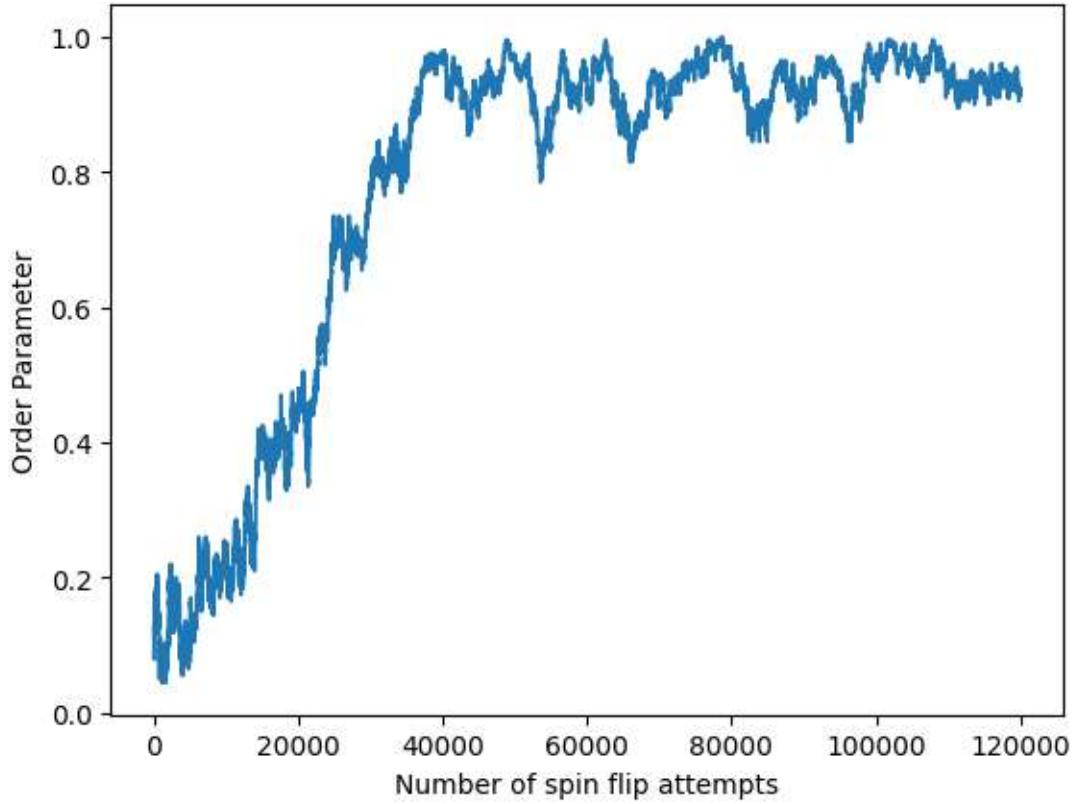
Results of Metropolis on NNN Ising for R<0.5

- For R>0.5, a perfect ferromagnet was allowed to get closer to super-antiferromagnetic state and this lattice was then used as initial lattice.

Equilibration time for R = +1 for L = 16 at lower temperature

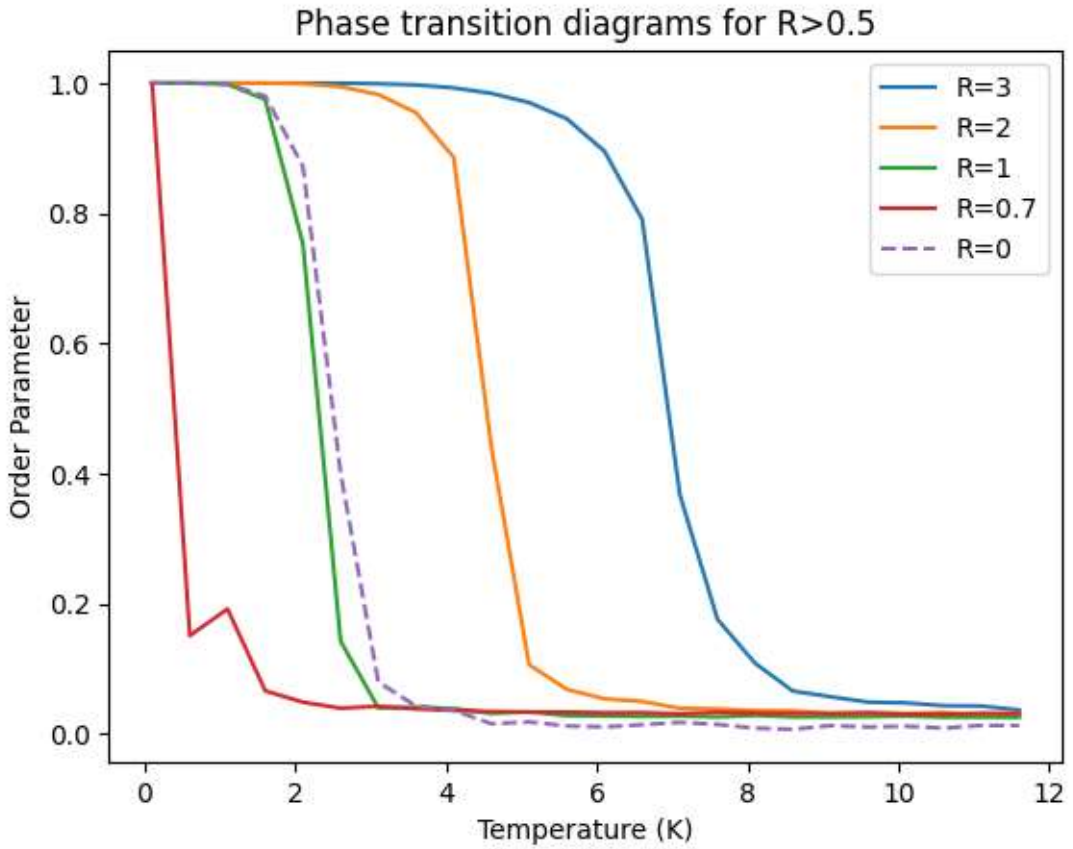


Equilibration time for R = +1 for L = 16 at higher temperature



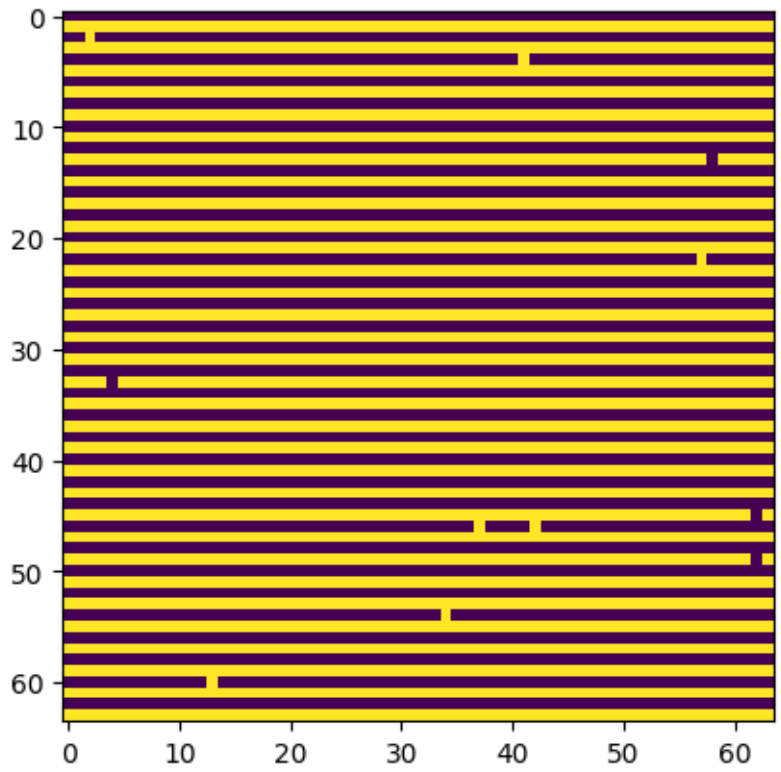
Results of Metropolis on NNN Ising for R<0.5

- Critical temperature increases with increasing magnitude of R (for R>0.5).
- Here the order parameter is taken to be magnetization average of four sublattices.

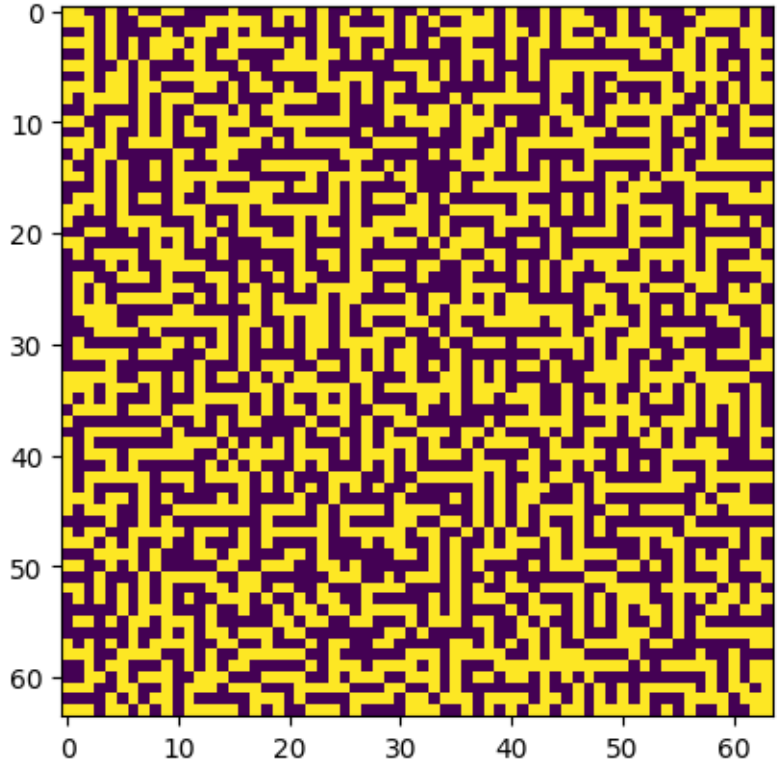


Results of Metropolis on NNN Ising for R>0.5

- Metropolis works perfectly even for NNN interactions. The following figure shows the snapshot of simulation at two different temperatures for R = +2.



Super-antiferromagnetic Phase obtained from simulation at T = 2.5, for R = 2



Paramagnetic Phase obtained from simulation at T = 8.5, for R = 2

Long Range Ising Model

- Frustrated Josephson junction arrays (FJJAs) physics can be mapped to long-range Ising model.
- Introduction of long-range interactions leads to increased time complexity of the traditional Metropolis algorithm to $\mathcal{O}\ (N^2)$.
- Swendsen-Wang algorithm also fails here due to increased time complexity to $\mathcal{O}(N^2)$.
- $H = -\sum_{\langle ij \rangle} J\left(R_{ij}\right) s_i s_j$, where : $J_{ij} = r(i,j)^{-(d+\sigma)}$ is the interaction strength for interaction between s_i and s_j.
- Long range interactions lead to introduction of frustration in the system.
- Better algorithms are hence required to tackle this issue.

Long Range Ising Model Simulation: A “look-up” table approach

- Aim : To restrict calculation for all the interactions by only using some effective number of interactions.
- Luijten and Blöte proposed a “look-up” table based approach. [**5**]
- The probability of placing bonds between p_{th} pair ($p = (1,2, \dots, N(N-1)/2)$) can be written as:

$$P_{\text{bond}}(p) = 1 - e^{-2K_p}$$

$$\text{and, } P_{\text{no bond}}(m;n) = \prod_{p=m+1}^{n-1} e^{-2K_p} = \left[e^{-2\sum_{p=m+1}^{n-1} K_p}\right]$$

- We define an array A (“look-up table”) of length N_{pair} with elements of the form:

$$A_0 = 1, \qquad A_n = \prod_{p=1,\dots,n} e^{-2K_p}$$

$$P_{\text{no bond}}(m;n) = \frac{A_{n-1}}{A_m}$$

5. Luijten, Erik, And Henk W. J. Blöte. “Monte Carlo Method for Spin Models with Long-Range Interactions.” International Journal of Modern Physics C, vol. 06, no. 03, pp. 359 (1995).

Long Range Ising Model Simulation: A “look-up” table approach

Look-up table based approach:

Input: N_{pair}, array A defined in previous slide. l an initially empty list of connected pairs,

p = 0.

loop

Draw random number $\zeta \in [0,1]$.

$r = \zeta A_p$;

if $r < AN_{pair}$ **then**

break;

end if

Search the lowest index p such that $A_p \leq \zeta$

If the spins connected by pair p are parallel, add them to list l ;

end loop

return the list l of connected pairs.

Future Works

- The advantage of Metropolis algorithm is that due to local updating the results are accurate and covers a large variety of cases
- An algorithm which works on the principles of local update while also being fast could prove to be a really effective algorithm to tackle the problem of long-range interactions.
- F. Müller, H. Christiansen, S. Schnabel, and W. Janke, “Fast, Hierarchical, and Adaptive Algorithm for Metropolis Monte Carlo Simulations of Long-Range Interacting Systems,” Physical Review X, vol. 13, no. 3 (2023).
- Reaches time complexity of $\mathcal{O}(Nlog\ N)$ and is also generalizable to a large variety of cases.

$$\rho = e^{-\beta \Delta E}$$

$$\Delta E \leq -\frac{\ln \rho}{\beta} \equiv \Delta E_{\mathrm{th}}$$

- Sufficient to calculate the bounds : $\Delta \mathrm{E}_{\mathrm{min}} \leq \Delta \mathrm{E} \leq \Delta \mathrm{E}_{\mathrm{max}}$

Need for Machine Learning

Time complexity of Metropolis Algorithm is high for simulation of Long-Range Ising Model.

However, even training a machine learning model requires dataset from Monte Carlo.

An alternate machine learning based solution under the constraint of requiring less use of Monte Carlo can be a helpful solution.

Aim: To present a machine learning technique for the simulation of phase transition in Next Nearest Neighbor Ising model by using a model trained on Nearest Neighbor Ising model

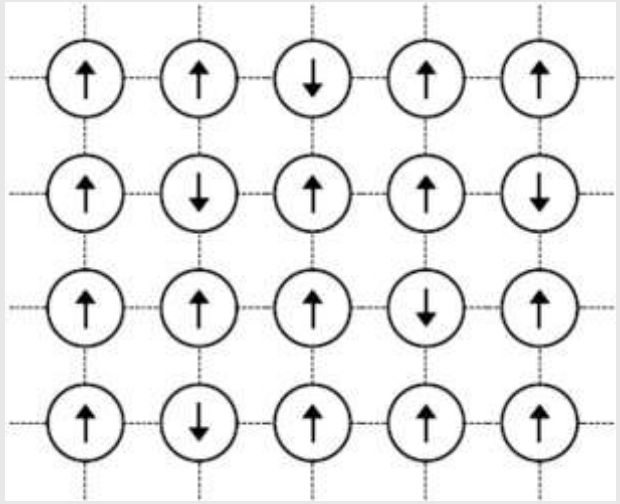
Major tasks:

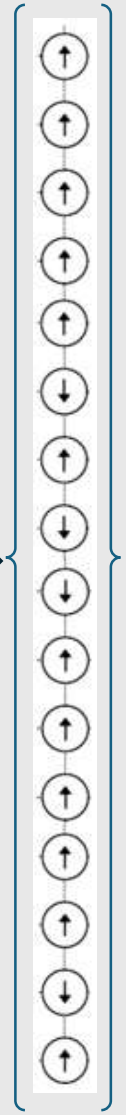
1. Using the pre-trained CNN model trained to perform phase transition detection for NNN Ising model of R<0.5 and testing the same model for R>0.5 as well (for the first time, to the best of my knowledge). $\left(R = \frac{J_{nnn}}{J_{nn}}\right)$
2. Fine-tuning the pre-trained CNN model to perform phase classification on NNN interaction case with R=+2 where the model (for the first time, to the best of my knowledge).

We will perform our analysis using a Fully Connected Neural Network (FCNN) and a Convolutional Neural Network (CNN)*.

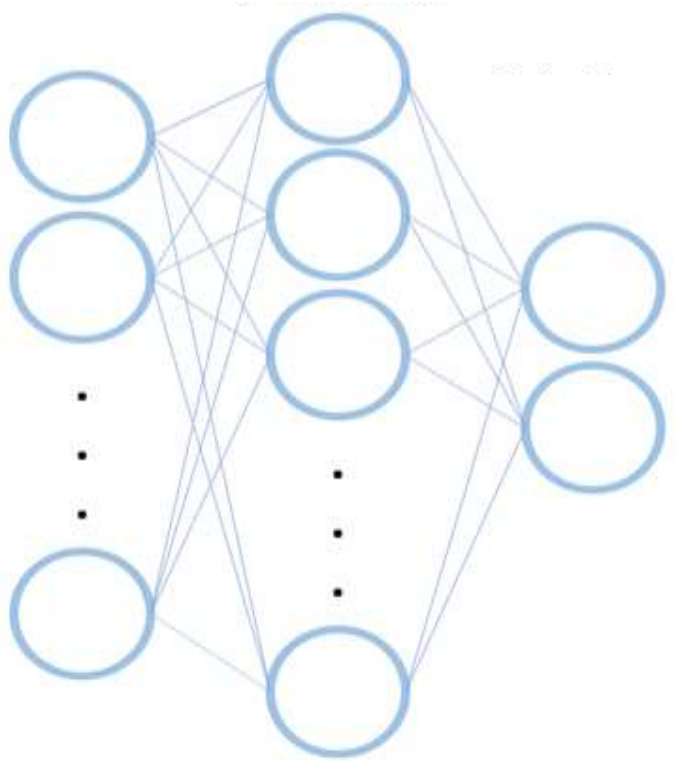
*FCNN structure same as (Carrasquilla and Melko (2017)) whereas CNN structure is slightly modified. Later a completely different CNN model is also proposed.
Source: Carrasquilla J, Melko RG., "*Machine learning phases of matter.*" Nat Phys.,13, 431 (2017).

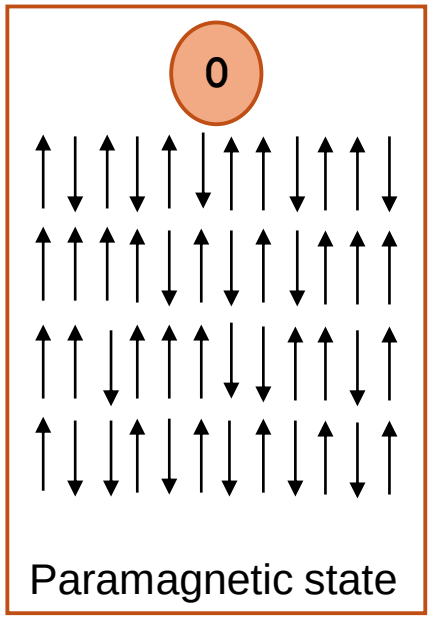
Working of a Fully Connected Neural Network



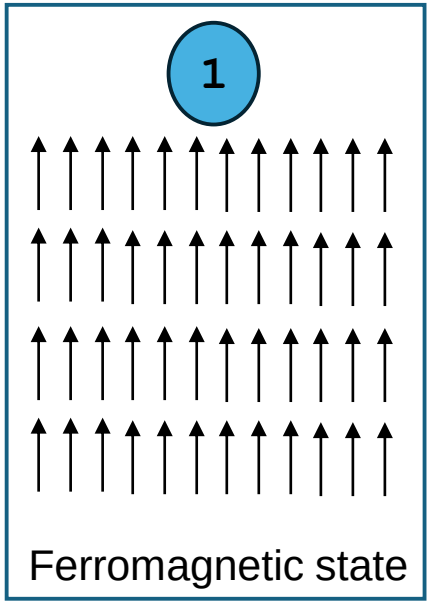


Flattening of input lattice configuration destroys the spatial features



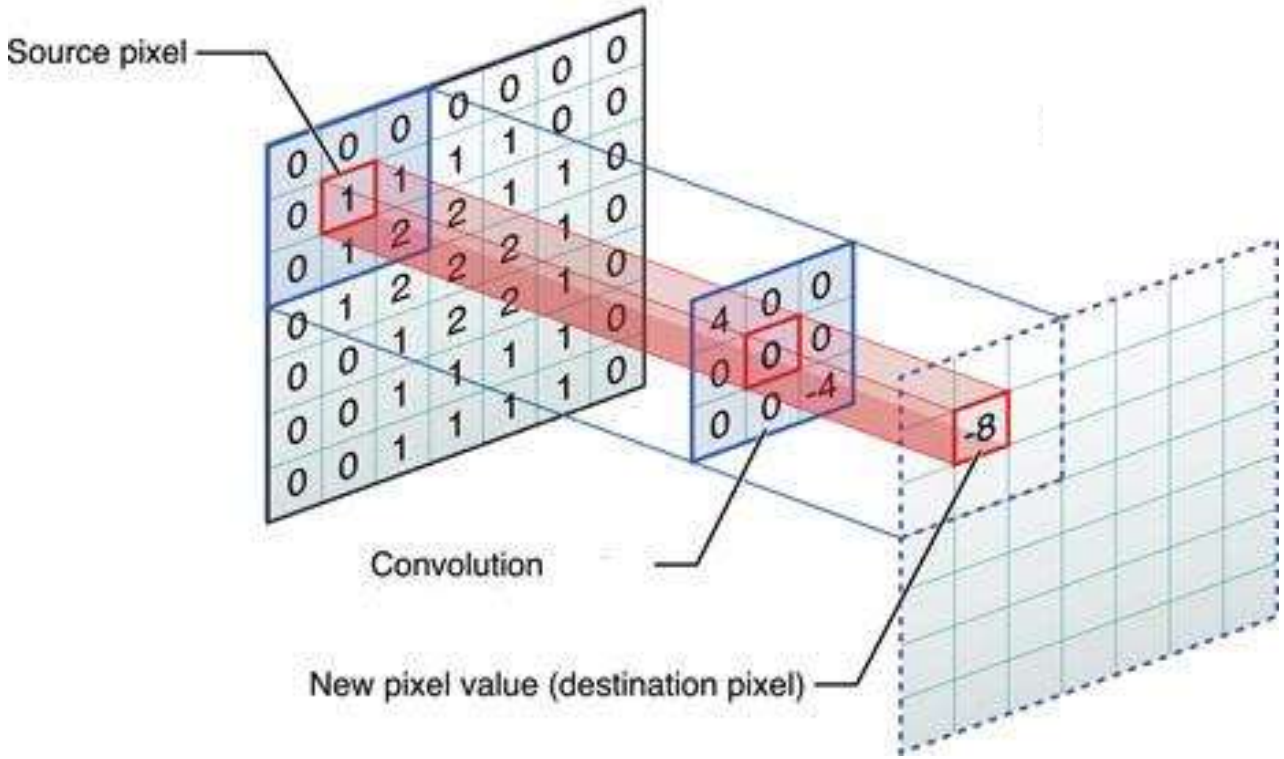


Paramagnetic state



Ferromagnetic state

The Convolution Operation



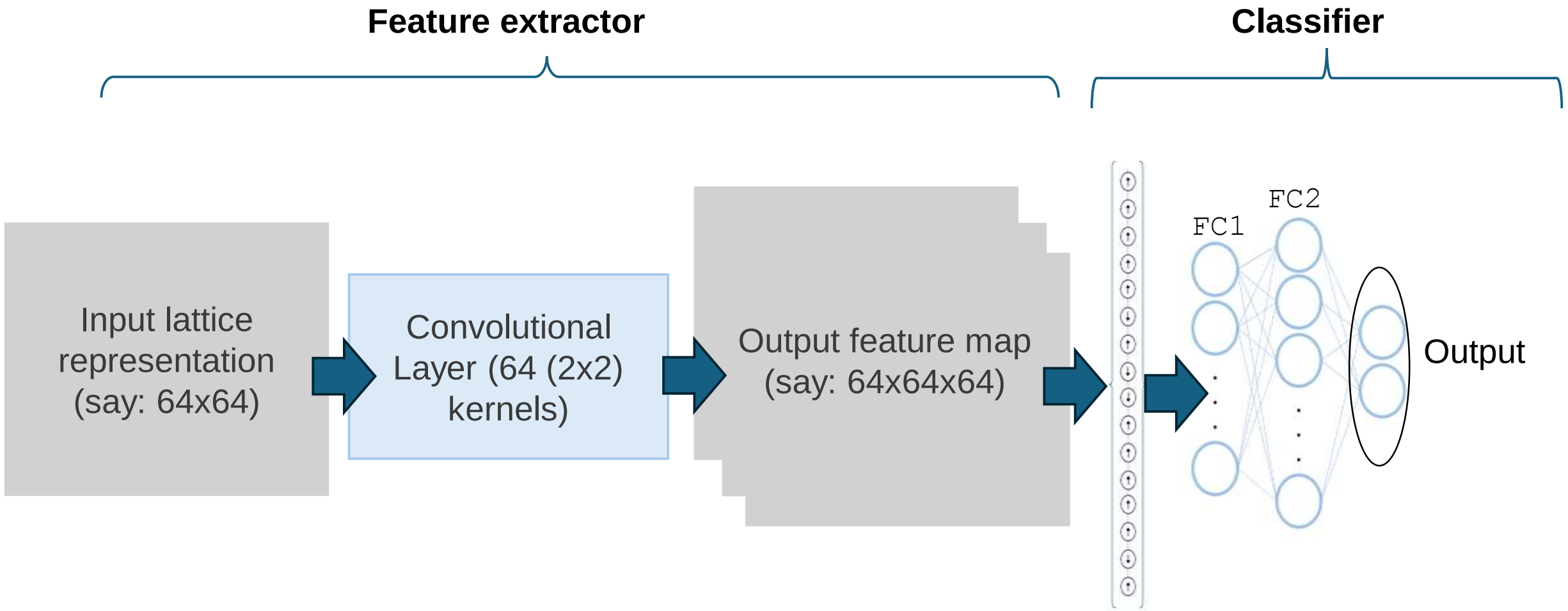
The convolution operation

- The Convolution operation helps the model to learn the **spatial features of the dataset**.

- This is not possible in a Fully Connected Neural Network because flattening the matrix destroys the spatial properties needed to be learned.

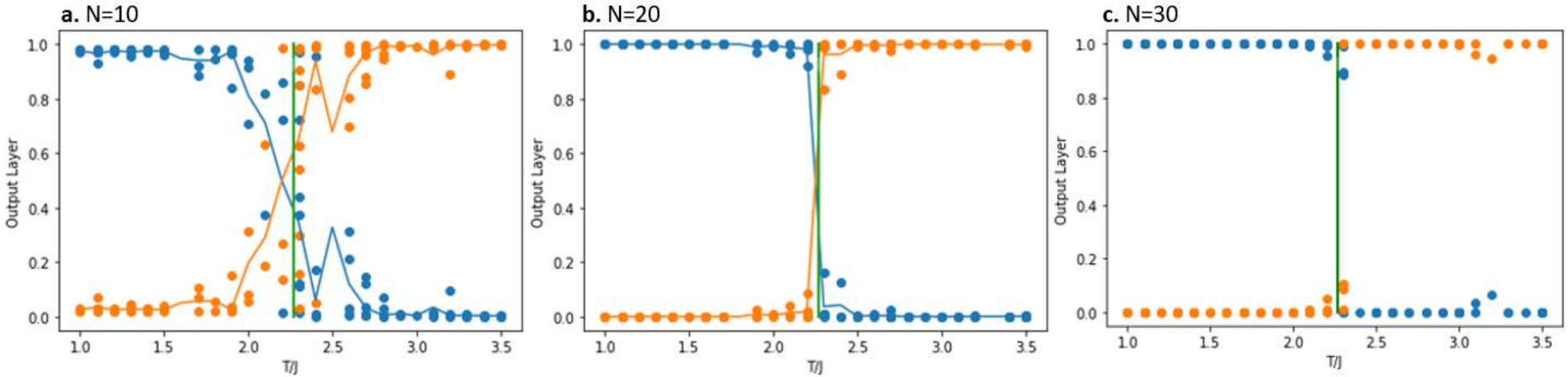
Source of the figure: FutureLearn (https://www.futurelearn.com/info/courses/intelligent-systems/0/steps/245920)

CNN Model


Feature extractor
Classifier
Input lattice representation (say: 64x64)
Convolutional Layer (64 (2x2) kernels)
Output feature map (say: 64x64x64)
FC1
FC2
Output

Task 1: Training FCNN and a CNN model for NN Ising model.

Results obtained from the Fully Connected Neural Network



The blue line depicts probability of input being ferromagnetic whereas the red line depicts probability of input being paramagnetic with temperature on the X-axis for models trained with (a.) N=10, (b.) N=20 and (c.)N=30.

The green line depicts the value corresponding to $\frac{kT_C}{J} = {}^{2}/_{\ln(1+\sqrt{2})}$ whereas the intersection of blue and orange curve is the model's prediction of the Curie Temperature.

These predictions are on the test dataset, thus depicting that the model is able to generalize the results well.

CNN performs better phase transition prediction than FCNN

Accuracy of FCNN and CNN for lattice dimensions of L=10,20 and 30 for anti-ferromagnetic to paramagnetic transition.

	L=10	**L=20**	**L=30**
FCNN	87%	91%	94.7%
CNN	**97.5%**	**98.9%**	**99.2%**

Accuracy of FCNN and CNN for L=10,20 and 30 for ferromagnetic to paramagnetic transition.

	L=10	**L=20**	**L=30**
FCNN	93.7%	96.6%	97.2%
CNN	**97.9%**	**98.8%**	**99.5%**

Conclusion:

Both FCNN and CNN are able to give good accuracies especially for higher lattice dimensions. However, CNN architecture performs slightly better than an FCNN.

Task 2. Using the CNN model trained for the ferromagnetic case to detect phase transition in the anti-ferromagnetic case with data preprocessing and vice versa.

Pre-processing: $s_{i,j} = (-1)^{i+j} s_{i,j}$ for all <i,j>

To use pre-trained FCNN and CNN models (trained on the anti-ferromagnetic to paramagnetic transition) for phase transition detection in the ferromagnetic to paramagnetic transition

Model	Accuracy
FCNN	96%
CNN	96%

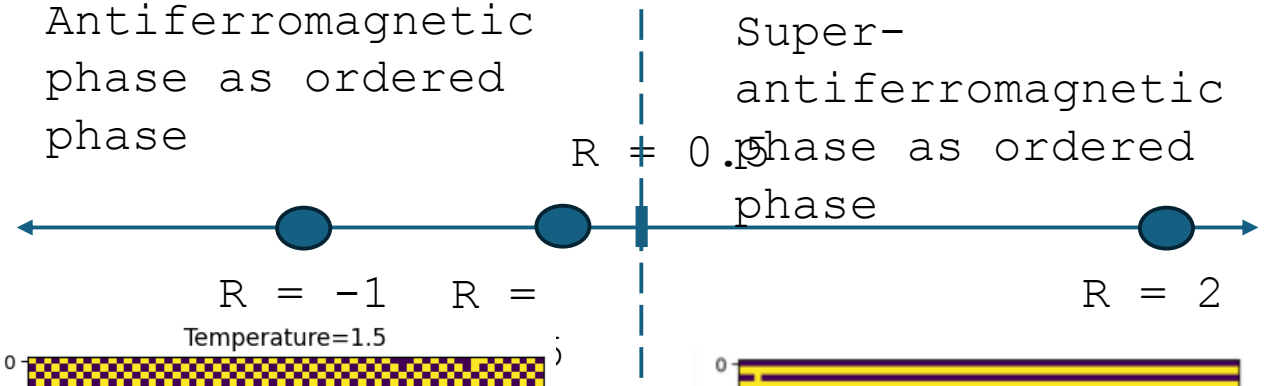
To use pre-trained FCNN and CNN models (trained on the ferromagnetic to paramagnetic transition) for phase transition detection in the anti-ferromagnetic to paramagnetic transition

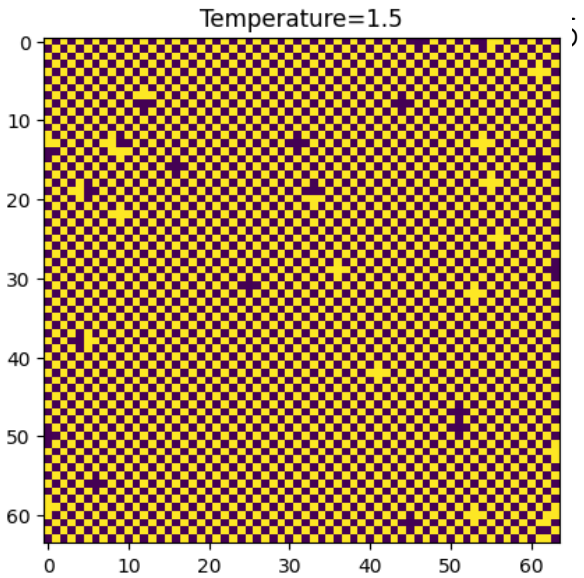
Model	Accuracy
FCNN	95.8%
CNN	95.6%

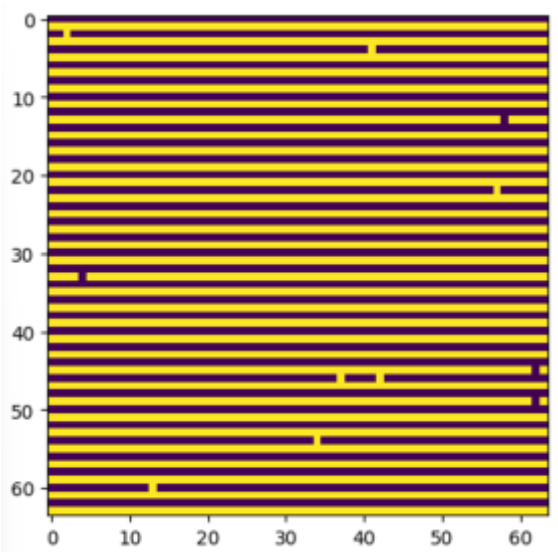
Phase Transition Prediction in Next Nearest Neighbor Ising Model

$$H = J_{nn} \sum_{nn\ pairs} s_i s_j + J_{nnn} \sum_{nnn\ pairs} s_i s_k$$

$$R = \frac{J_{nnn}}{J_{nn}}$$



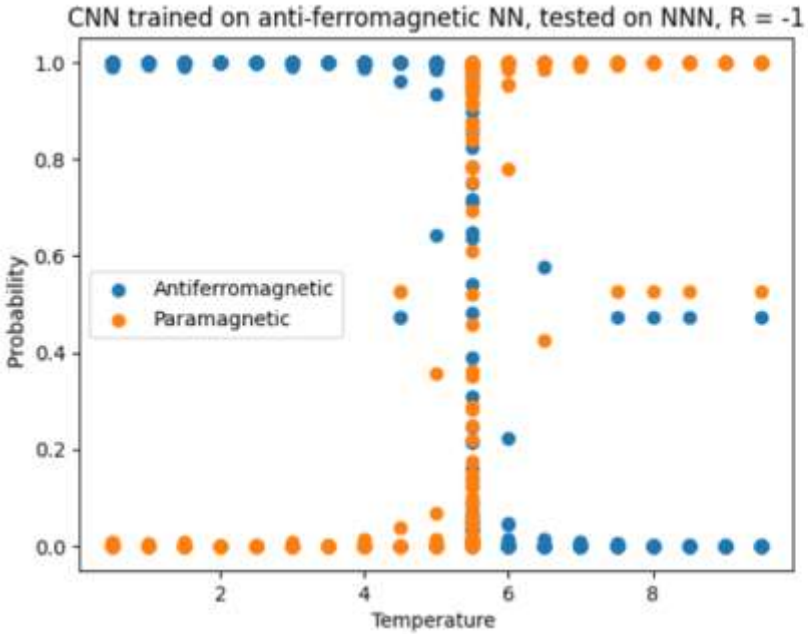


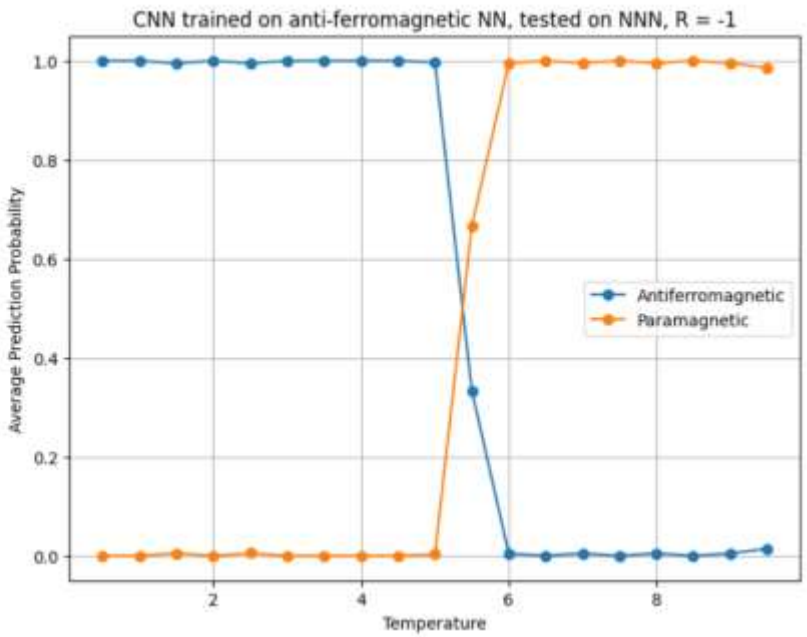


Task 3. Using the CNN model trained to classify between anti-ferromagnetic and paramagnetic configurations to perform phase transition classification for NNN case of R<0.5.

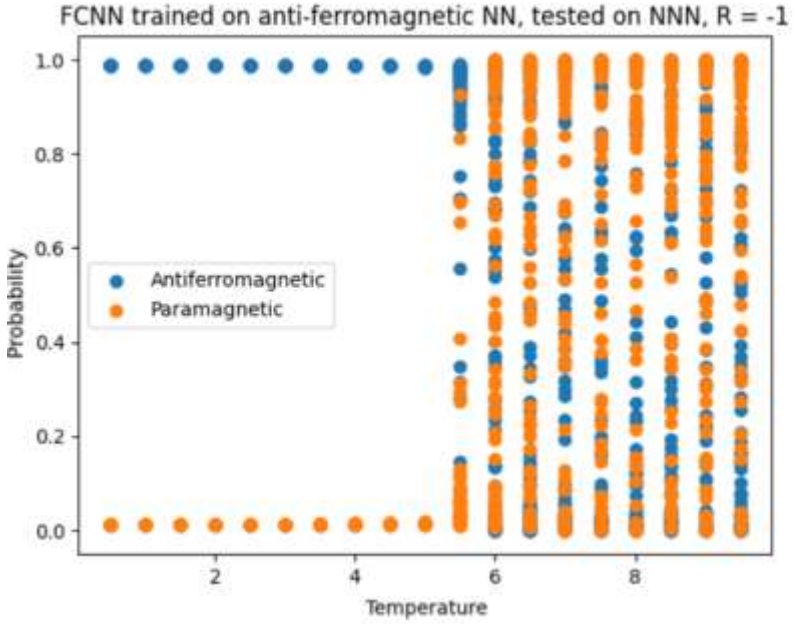
(for the first time in literature, to the best of my knowledge)

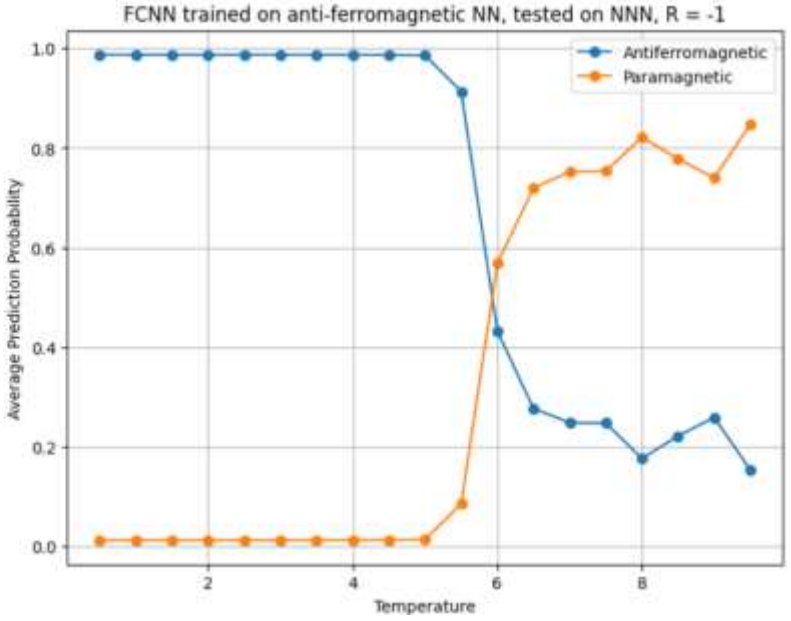
Results for R=-1





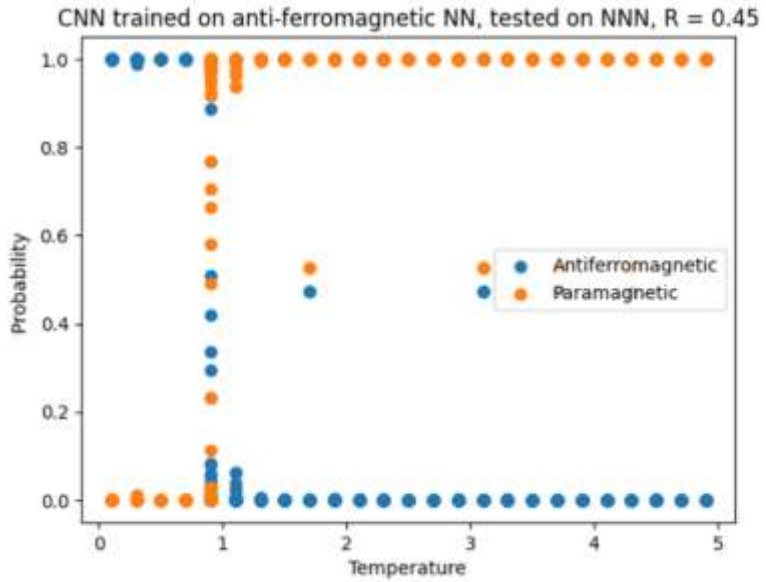
CNN achieves 99.2% accuracy
Achieves perfect classification

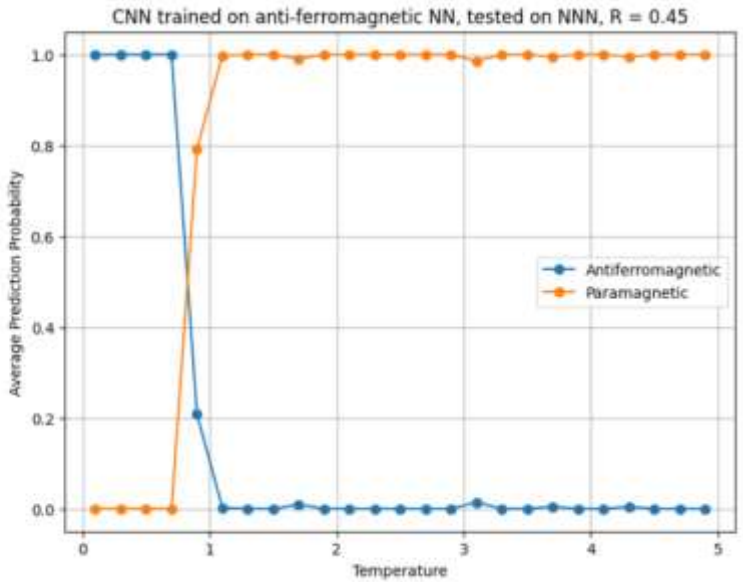




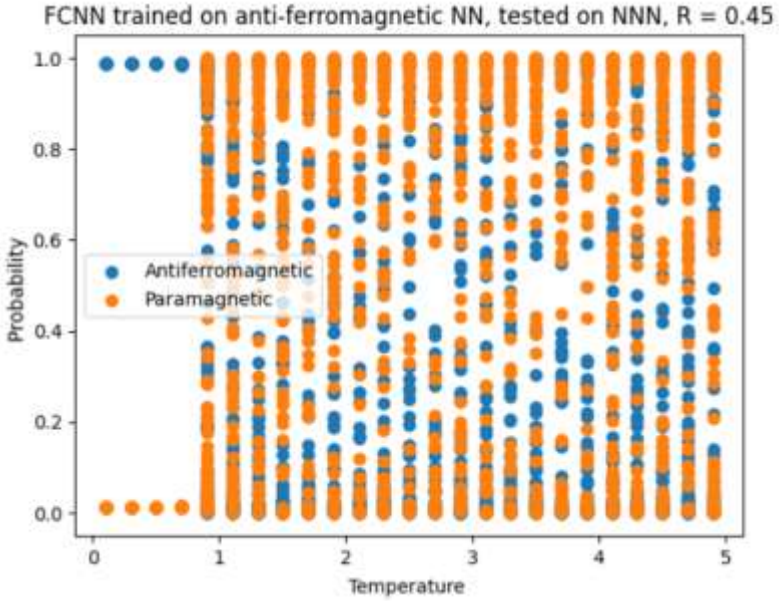
FCNN achieves 80.8% accuracy
After the critical temperature, the model behaves as ‘confused’

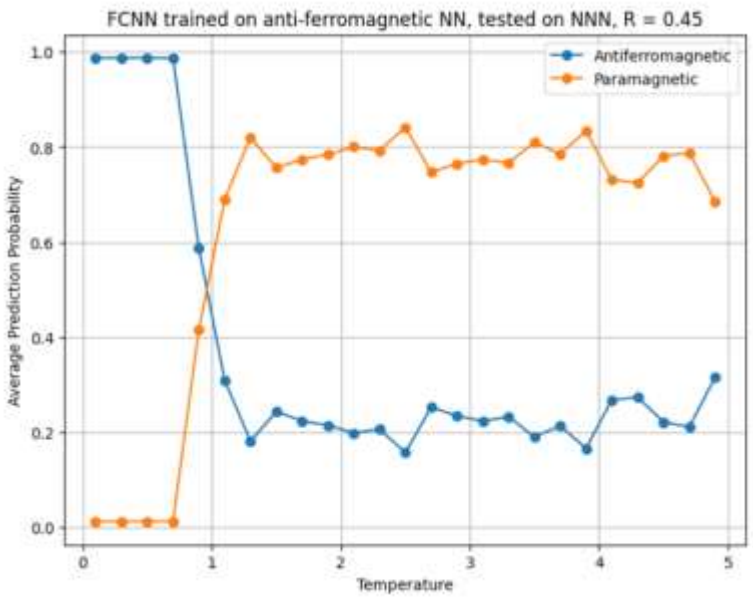
Accuracy of FCNN is 80.8% for R=0.45 due to imbalanced dataset.





Pre-trained CNN1 provides high-accuracy results in the case of R=0.45.





Pre-trained FCNN has an accuracy of 80.8% for R=0.45

CNN vs FCNN for NNN phase transition prediction

Accuracy values for different values of R:

Model	Accuracy		
	R=-1	R=0.45	R=2
CNN	**99.2%**	**96%**	**58%**
FCNN	89%	80.8%	51.2%

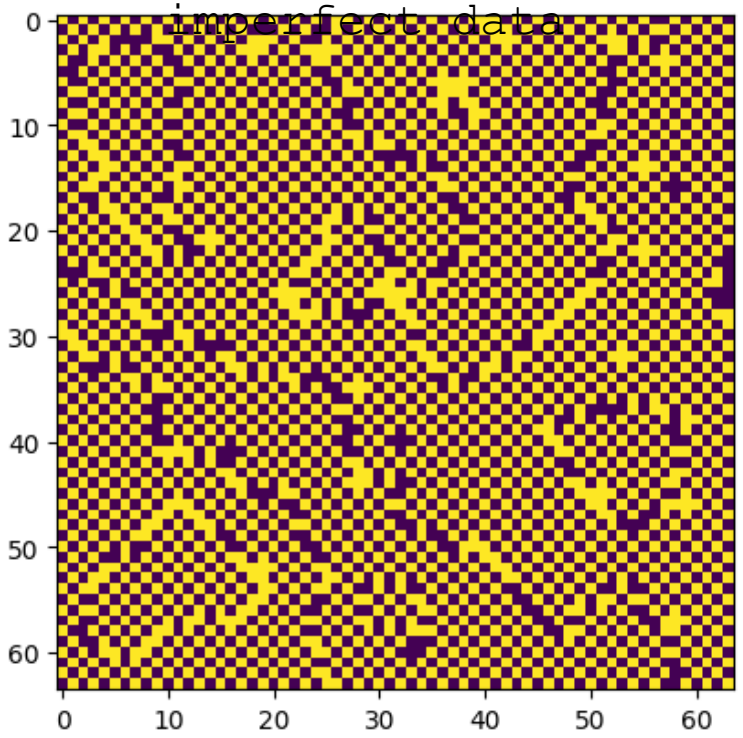
May be due to imbalanced dataset

Both models fail in R=2 (R>0.5)

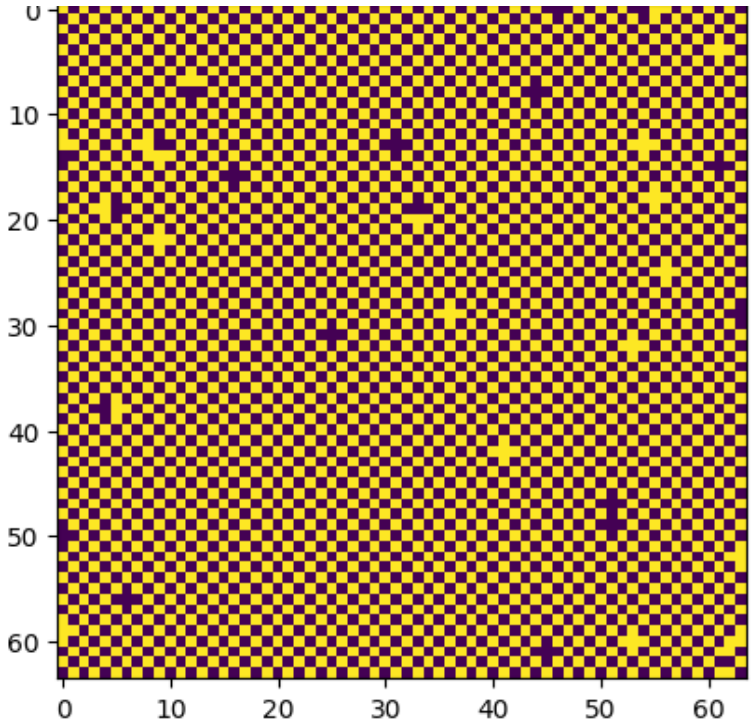
Data generated from Metropolis before reaching equilibrium

Metropolis has a large equilibration time if we start with a ferromagnetic configuration.

Slightly imperfect data

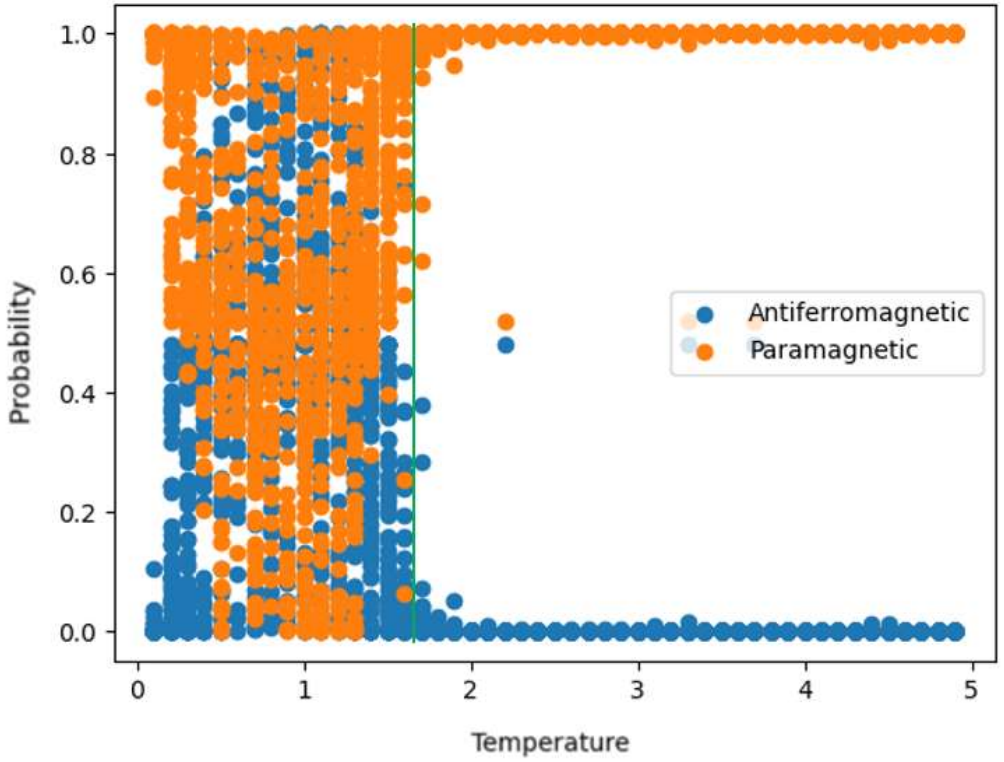


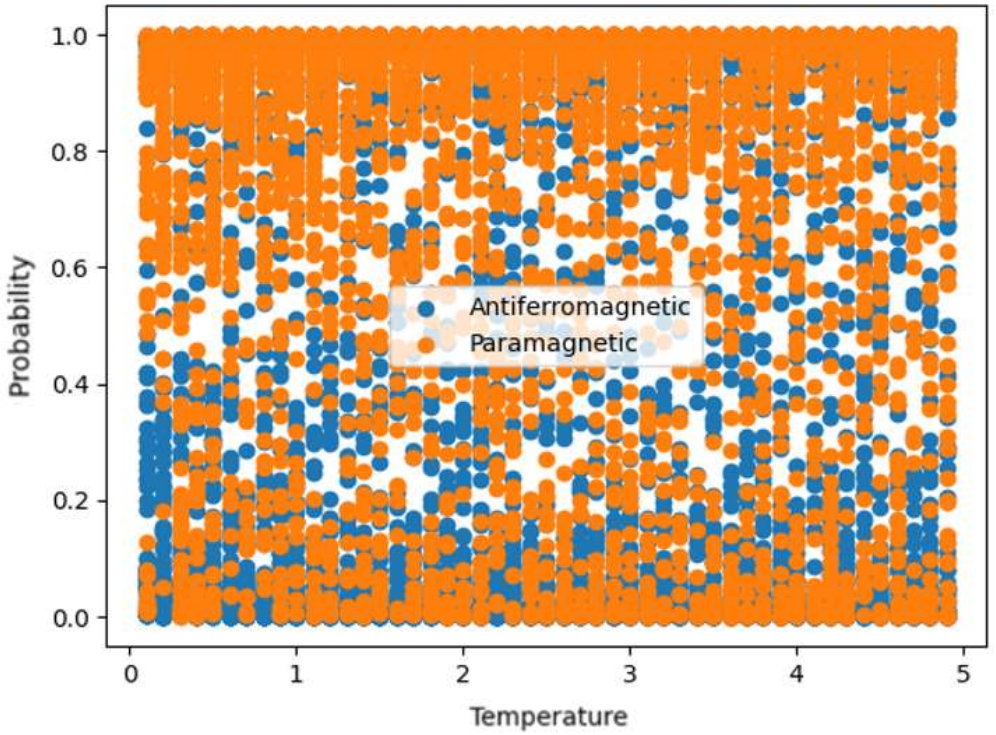
Perfect data



The lattice representation on the left depicts a lattice which has not yet reached equilibrium due to large equilibration time of Monte Carlo, as compared to the equilibrium configuration on the right (yellow represents +1 spin and black represents -1 spin).

Results from data generated from Metropolis before reaching equilibrium





For R = -1, CNN model (left) is able to predict phase transition by giving accurate results for lattice after the transition temperature but giving ‘confused’ results before the same. However, the FCNN model (right) completely fails to predict the phase transition.

For NNN Ising Model R=2, both CNN and FCNN fail to give the correct results

Accuracy values for different values of R:

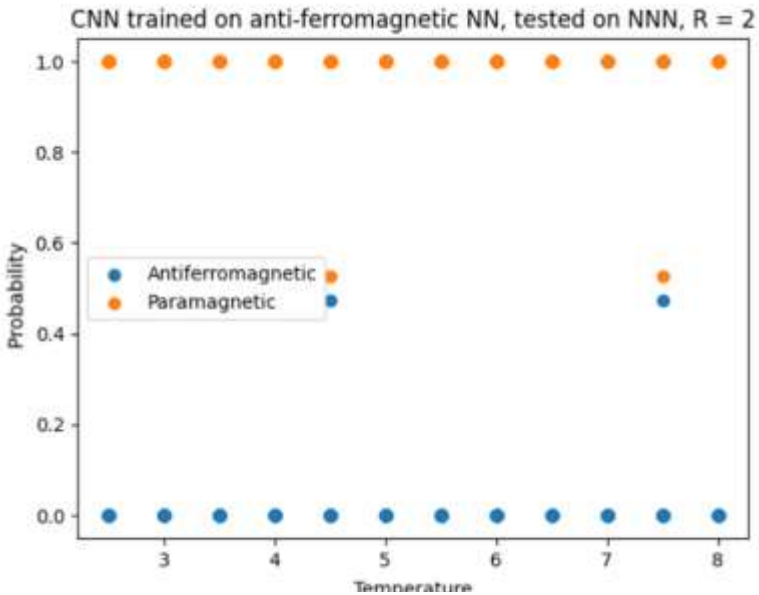
Model	Accuracy		
	R=-1	R=0.45	R=2
CNN	**99.2%**	**96%**	**58%**
FCNN	89%	80.8%	51.2%

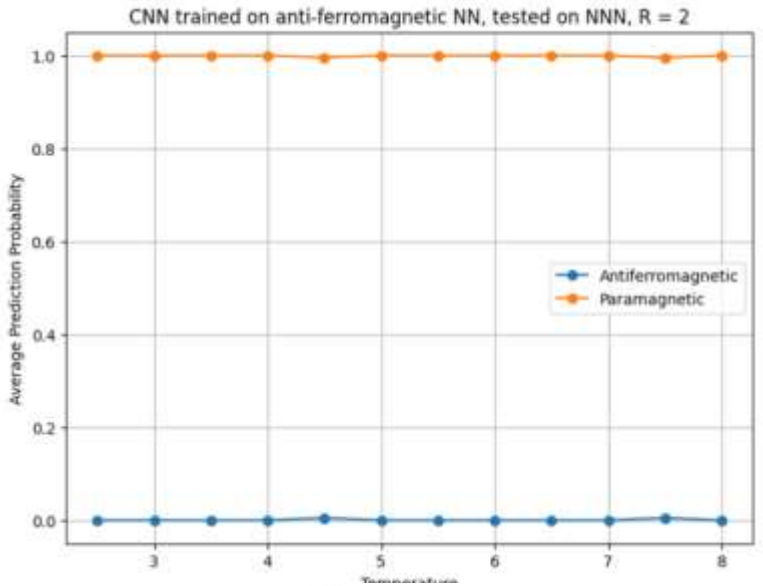
May be due to imbalanced dataset

Both models fail in R=2 (R>0.5)

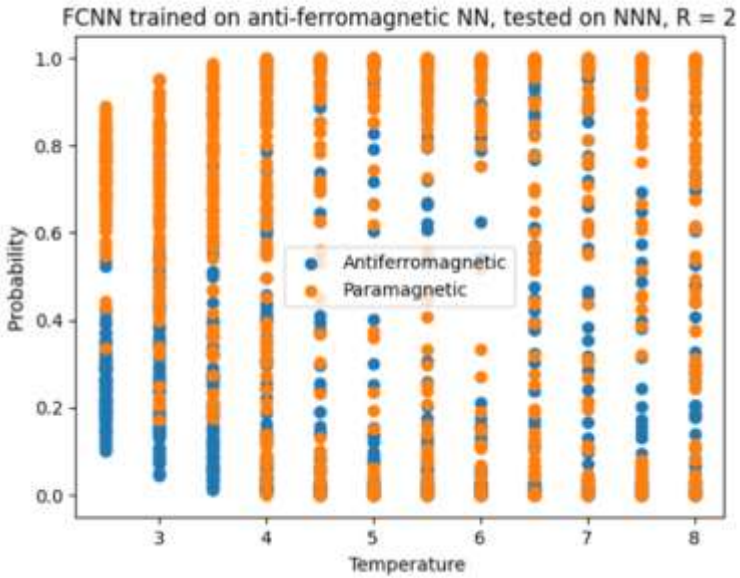
Results for super-antiferromagnetic to paramagnetic transition (R=2)

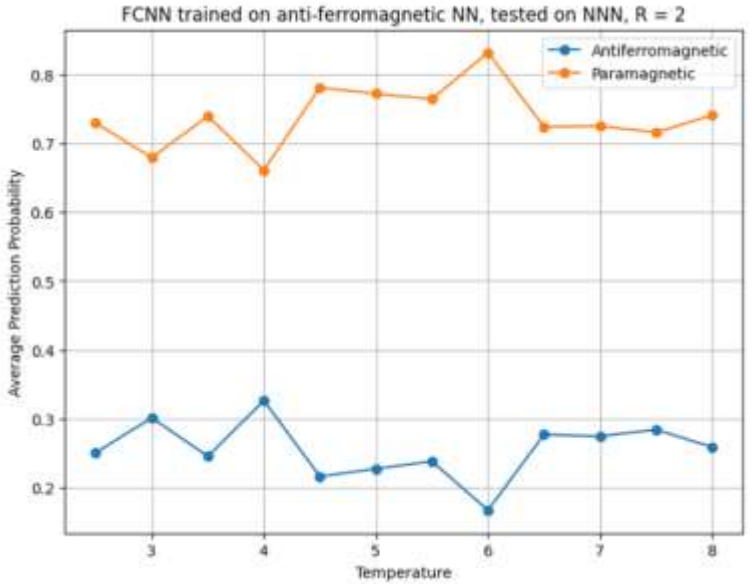
CNN confidently predicts all lattice configurations as **paramagnetic**.





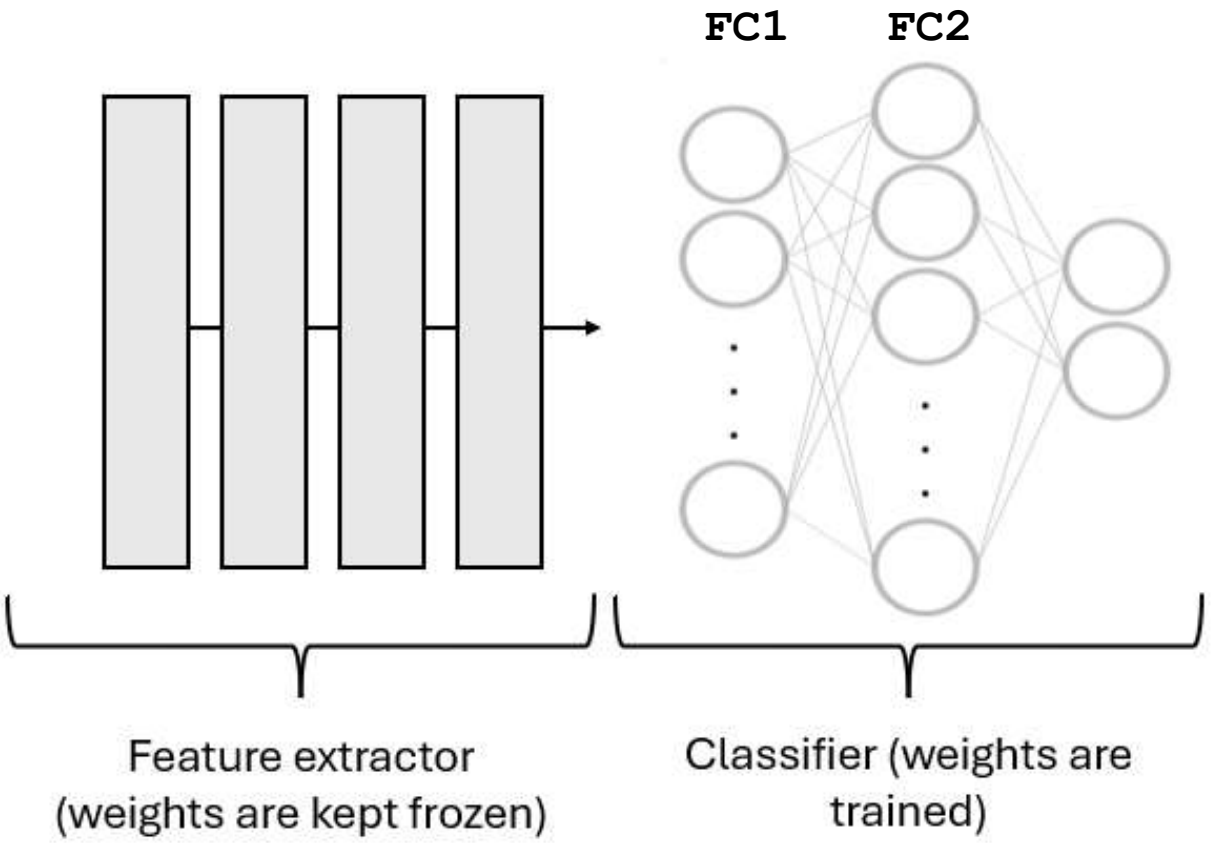
FCNN model gives a "confused" output.





We will use the fine-tuning method of transfer learning to leverage the pre-trained models

TRANSFER LEARNING

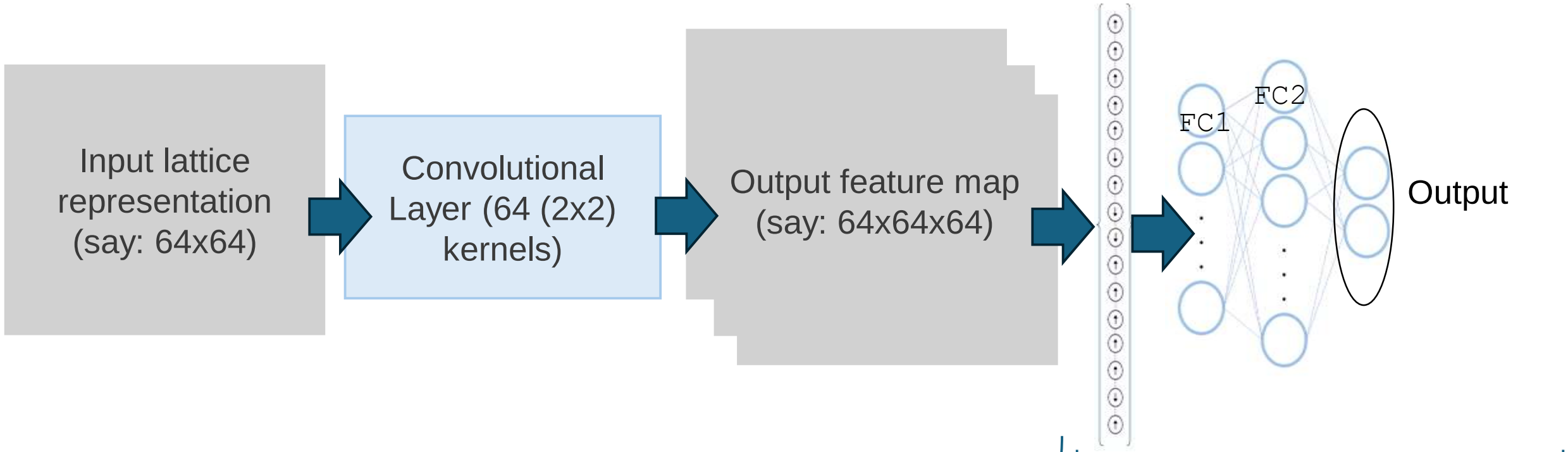


Fine-tuning an ML Model

Task 4. Fine-tuning the CNN model pre-trained on anti-ferromagnetic configurations for performing phase classification on ferromagnetic NN dataset and vice versa without data preprocessing.

We will do the task for 3 iterations of training on a small dataset of 256 lattice configurations and compare the results.

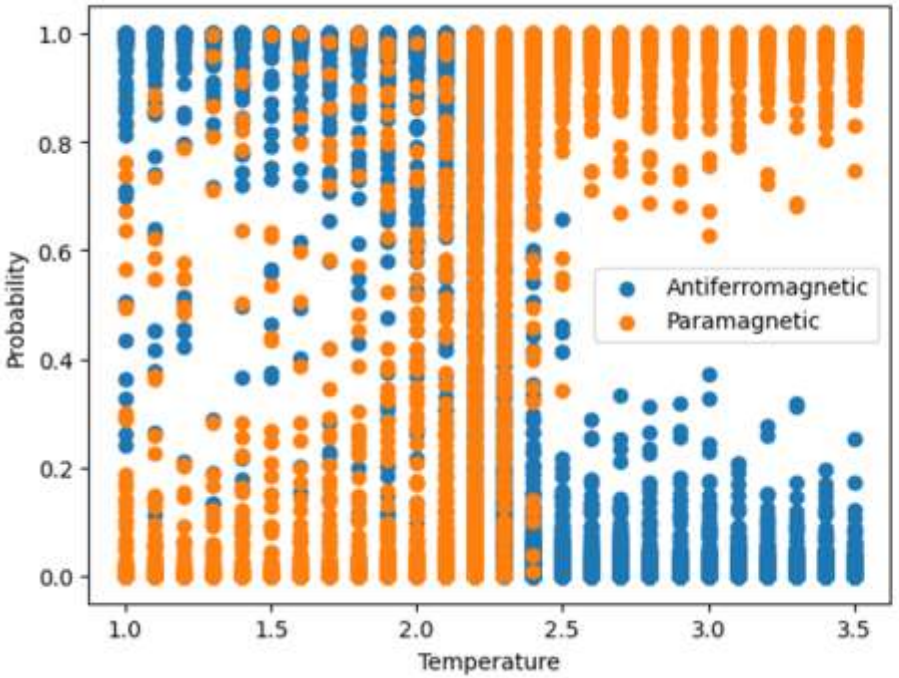
Initialization of the two fully connected layers FC1 and FC2 in the classifier with pretrained weights affects the results

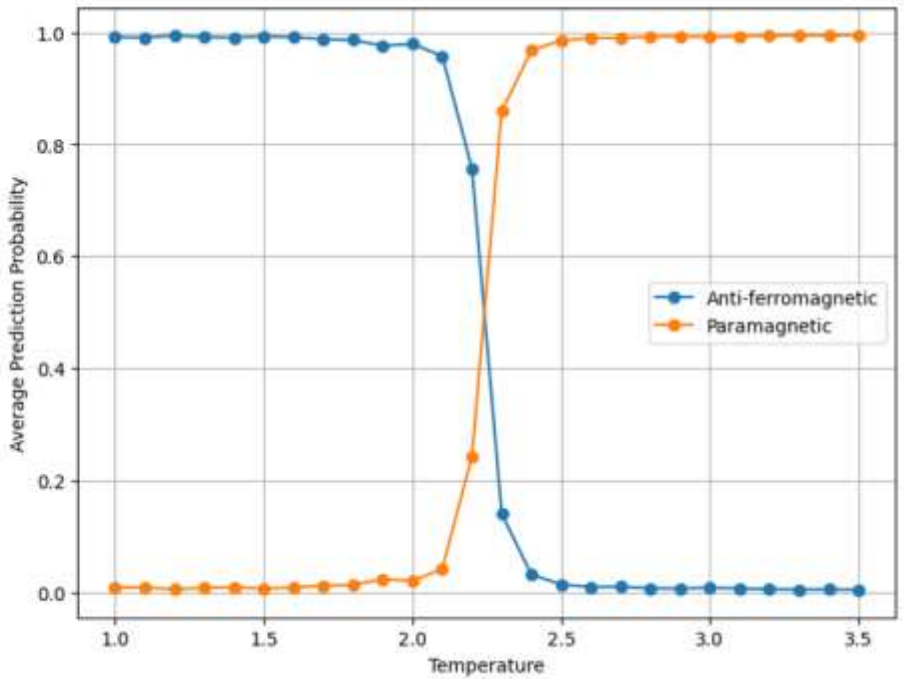


Weights are kept frozen for the feature extractor;
They are initialized with the same weights as the pre-trained model

Weights are updated during training for the classifier;
They are may or may not be initialized with the

Model trained to predict anti-ferromagnetic to paramagnetic transition





Predicted probability of a phase vs temperature plots for CNN trained for anti-ferromagnetic to paramagnetic phase transition prediction.

Accuracy values for different cases of initialization

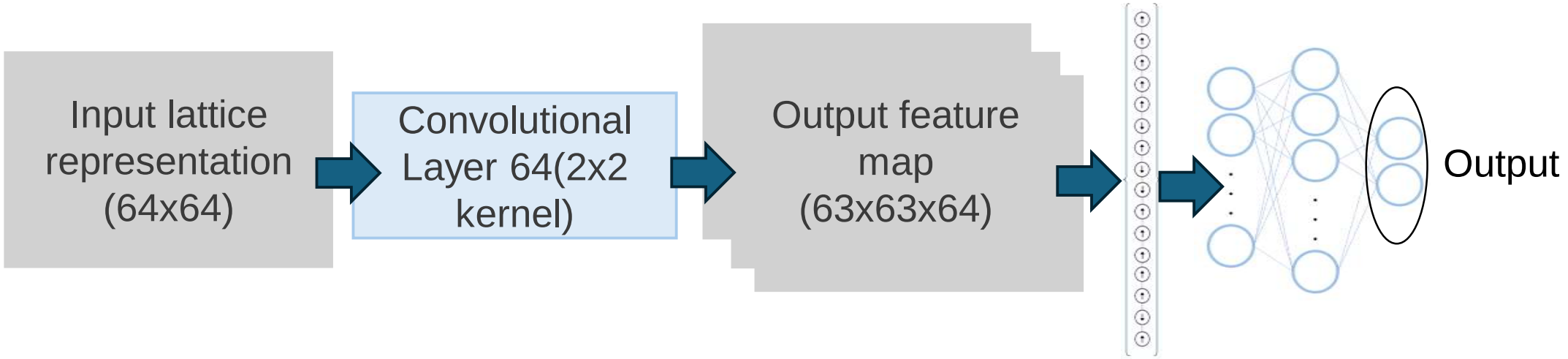
Model tested on	Model trained (accuracy)	Transfer Learning (accuracy)			
		Both initialized	FC1 initialized	FC2 initialized	Uninitialized
Antiferromagnetic samples	**98.1%**	50%	51.5%	**93.9%**	**98.2%**
Ferromagnetic samples	**87%**	50.5%	52.1%	**97.6%**	**98%**

Task 5. Fine-tuning the pre-trained CNN to perform phase classification on NNN interaction case with R=+2 where the model

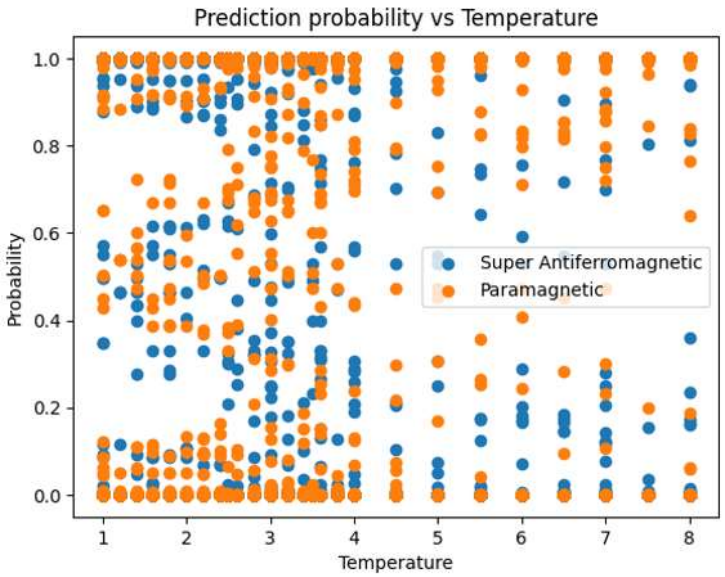
(for the first time in literature, to the best of my knowledge).

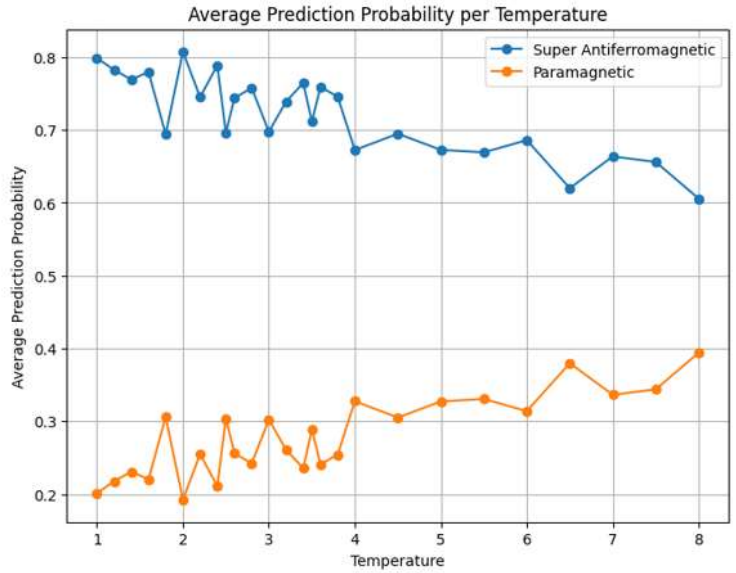
We will perform the task on 9000 lattice configurations for R=2 NNN Ising model.

Model 1:



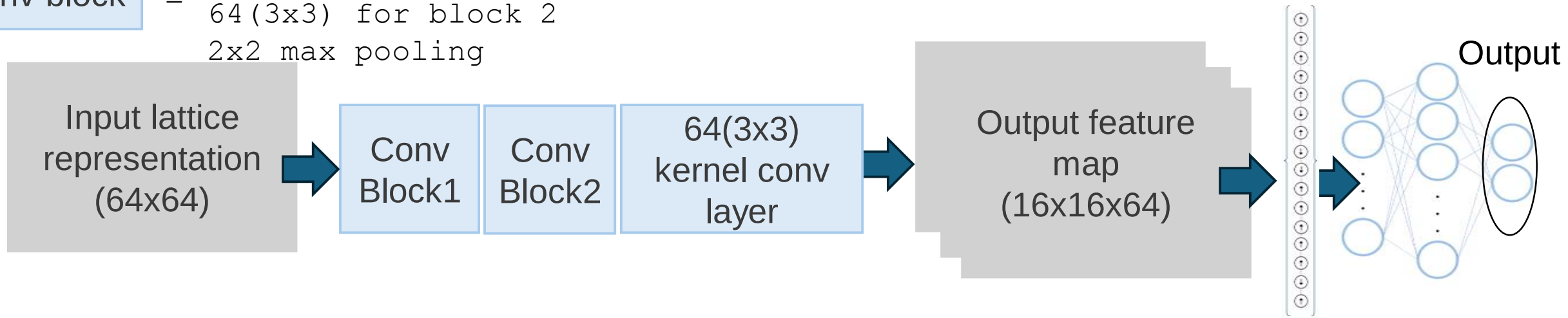
CNN1 architecture fails to train on R=2 data.



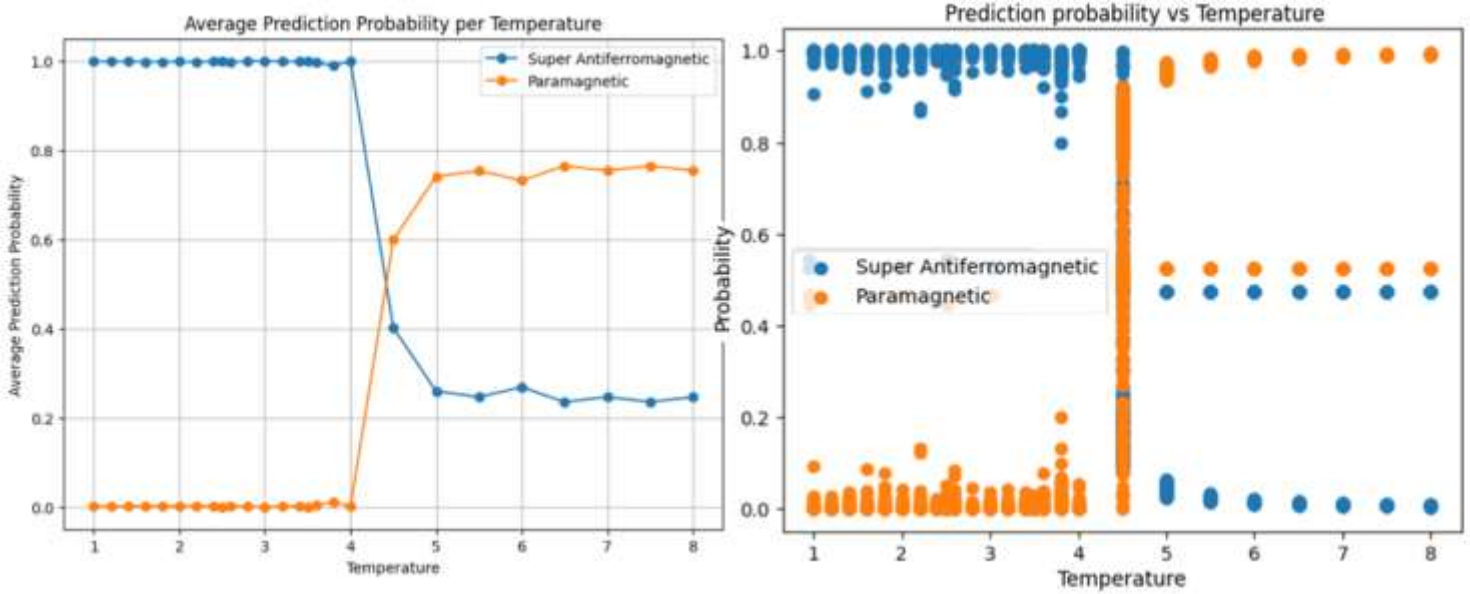


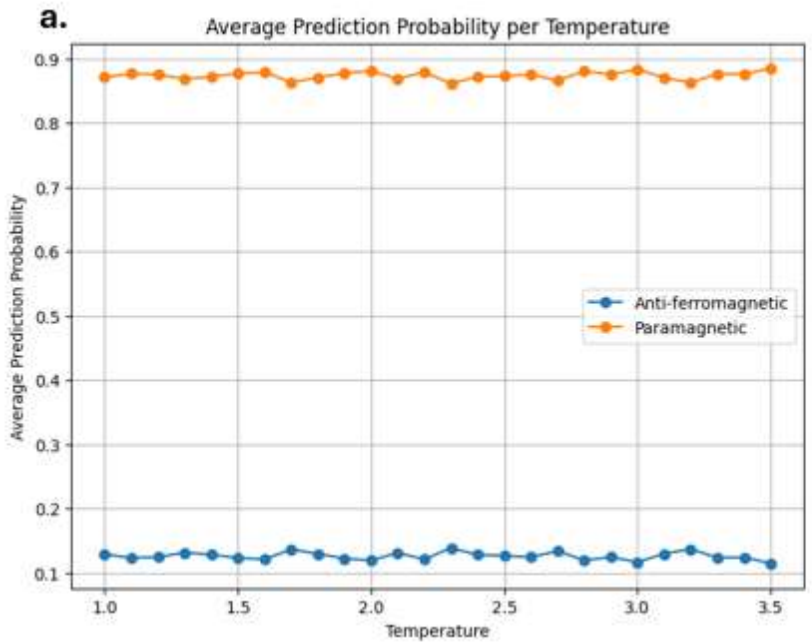
Model 2:

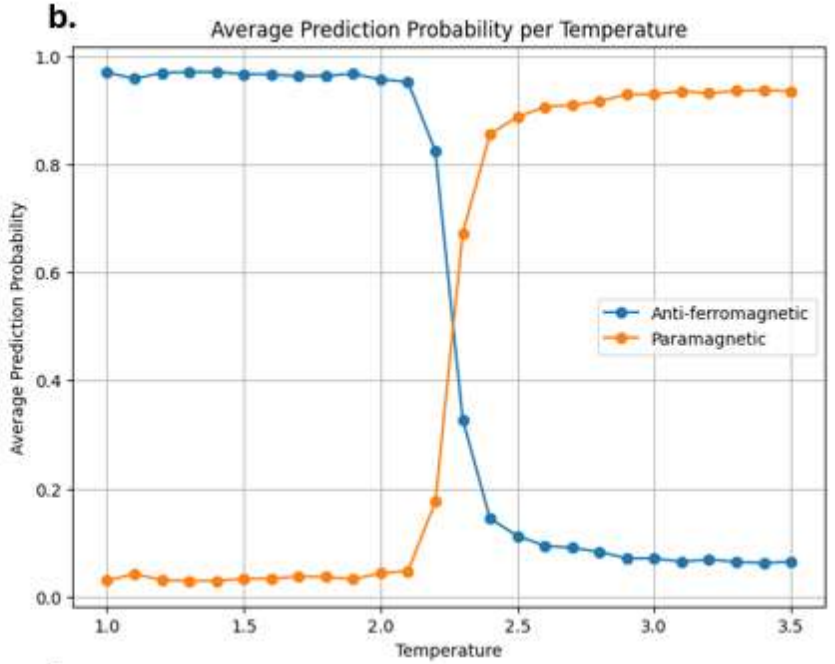
Conv block = 32(3x3) kernel for block 1,
64(3x3) for block 2
2x2 max pooling

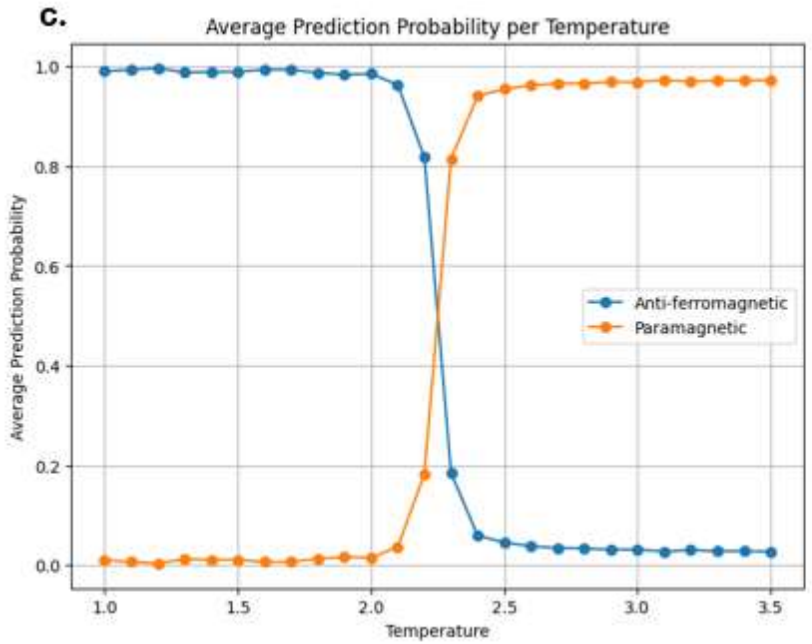


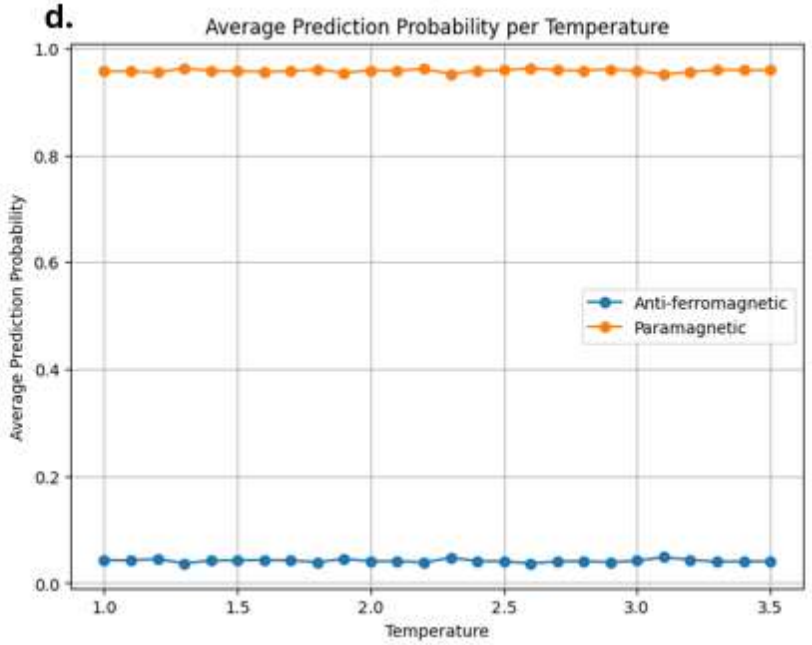
The modified CNN2 architecture on training on R=2 data achieved 95.5% accuracy.











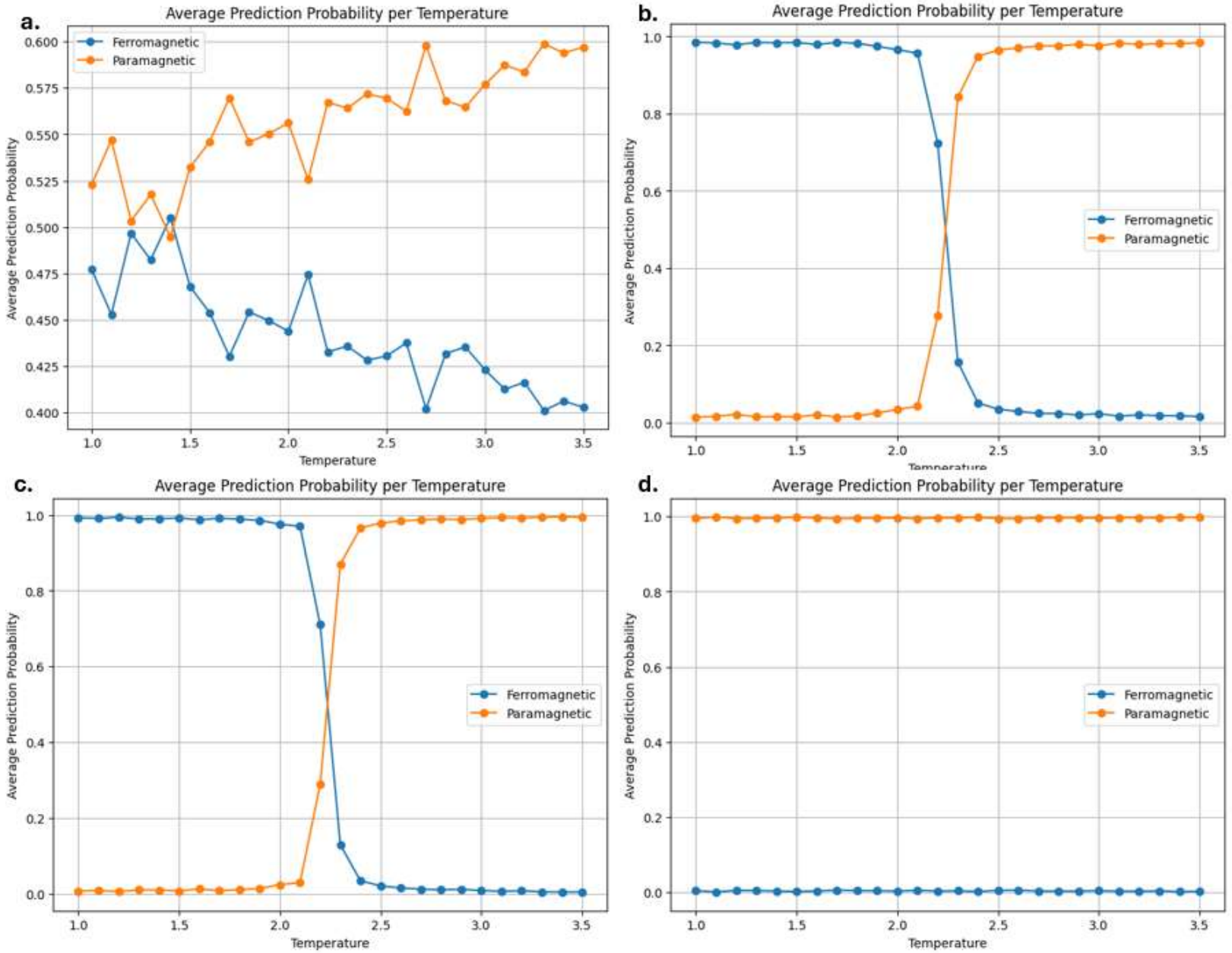
Phase transition prediction by transfer learning for on anti-ferromagnetic to paramagnetic phase transition with:

(a.) FC1 initialized,

(b.) FC2 initialized,

(c.) Both FC1 and FC2 un-initialized,

(d.) Both initialized, with weights of pre-trained network.



Phase transition prediction by transfer learning for on ferromagnetic to paramagnetic phase transition with

(a.) FC1 initialized,

(b.) FC2 initialized,

(c.) Both FC1 and FC2 un-initialized,

(d.) Both initialized, with weights of pre-trained network.

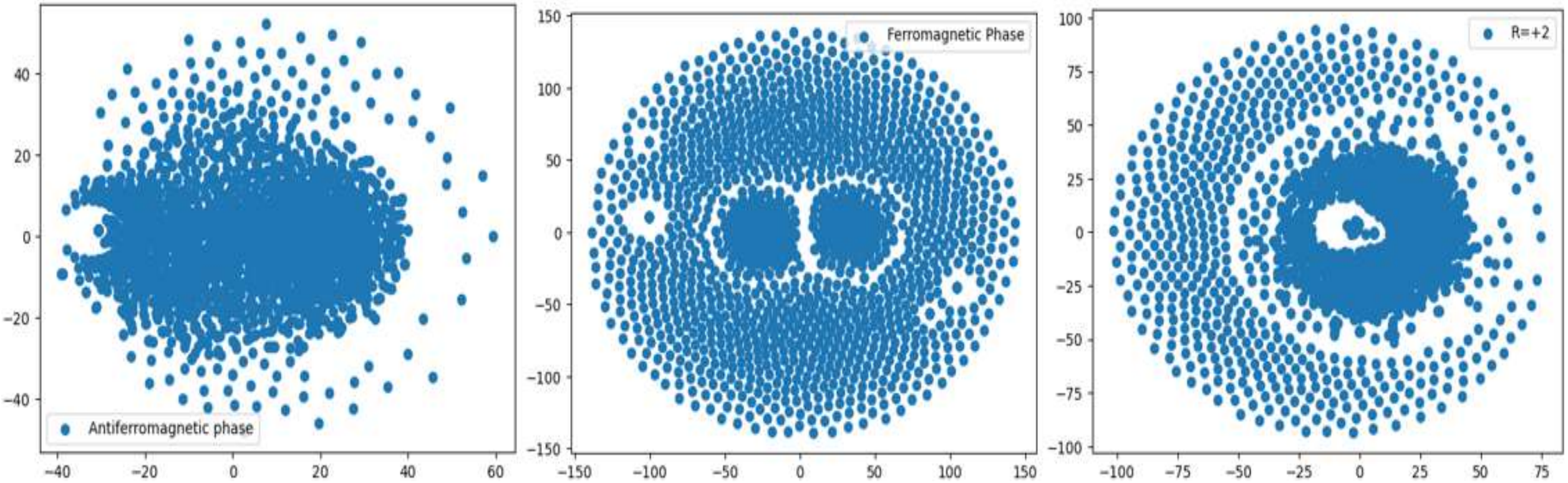
Transfer learning using CNN2 architecture produces reliable results

Model trained on	Model trained (accuracy)	Transfer Learning (accuracy)			
		Both initialized	FC1 initialized	FC2 initialized	Uninitialized
Antiferromagnetic samples	**95%**	40.3%	55%	**70.8%**	49%
Ferromagnetic samples	**93%**	38%	52%	**84.6%**	**86.4%**

Observation: Transfer learning works better when the initial model is trained on ferromagnetic samples.

We will use t-SNE to check the similarity in the anti-ferromagnetic, ferromagnetic and super-antiferromagnetic samples

T-SNE plots for ferromagnetic, anti-ferromagnetic and super-antiferromagnetic phases



Future Work

- To explore initializing weights of feature extractor with the model trained on ferromagnetic to paramagnetic transition and the weights of classifier with the ones of the model trained on anti-ferromagnetic to paramagnetic transition
- To explore more CNN based models (such as VGG) and perform hyper-parameter tuning to perform better at phase transition learning for R=2 in NNN Ising Model.
- Create pre-trained models and leverage them for the prediction of complicated phase transitions.
- Explore generative networks such as GANs as they be very efficient in learning generalizable features in the training data.
- Apply transfer learning to the LRIM case.

THANK YOU