

# The Ising Model

Henry Daniels-Koch and Boyang Zhang

October 26, 2015

## 1 Introduction

In this project, we look at a two dimensional Ising model, which explains the behavior of Ferro magnets around a critical temperature, where the magnetization suddenly drops from 1 to 0. Spins inside of a Ferro magnet allow the magnet to have as many states as the square of the number of spins. Each state has a probability determined by the Boltzmann factor. The Boltzmann factor is determined by the temperature and the energy. The energy is very complicated since it involves the interaction between the spins. However, we can simplify it by only considering interactions between nearest neighbors. In order to determine the exact equilibrium for different temperatures, we will need to calculate the energy of all of the states at this particular temperature. For large number of spins, again, the process will be very complicated. Therefore, to approximately calculate the equilibrium, we implement the Metropolis algorithm. This algorithm randomly chooses spins for calculation and avoids calculating states with extremely low probability by using the Boltzmann factor within the random choosing process.

## 2 Testing and Problems

After we successfully implement the model in code, we encounter several problems. The first problem is choosing temperatures. While testing different temperatures in a realistic range, somewhere around room temperature 300K, the magnetization always goes to zero. Therefore, we extend the testing range for  $T$  and find that the critical temperature is between 2 and 3. The reason for such small  $T$  is because we have set constants  $k$  and  $J$  to 1, so the  $T$  no longer has units of kelvin. Another problem we encounter is the fluctuation of the equilibrium related to the spins switched after each sweep. Initially we set each sweep to randomly choose  $N$  spins and use the Metropolis algorithm to determine the value for each spin. However, the results give large fluctuation around the equilibrium, especially between temperatures of 2.3 and 2.5. We observed that when the temperature is close to the critical temperature, it takes longer for  $M$  and  $E$  to reach equilibrium. Previously, the method we used for finding the average equilibrium value used an arbitrary range for all  $T$ . This results in

larger fluctuations, especially around the critical temperature. The new method implemented for calculating the average equilibrium uses ranges specific for different  $T$ . This gives a much better result. We used two different methods to test the code. The first one is to test whether the initial conditions will alternate the result. While keeping all the other conditions the same, switching spins in the initial array from all 1 to all -1 and also randomly assigned 1 or -1 provide the same result, although the number of sweeps required for reaching equilibrium differ. Therefore, the results are independent of the initial spins. We also check whether the code will work on arrays of different sizes and how this influences the results. Initially the code was for a  $64 \times 64$  array. Testing the code on arrays with size  $128 \times 128$ ,  $8 \times 8$  and  $16 \times 16$  gives similar  $M$  vs  $T$  results. However, the number of sweeps required to reach equilibrium differs for different size of the array.

### 3 Implementation of Code

To implement the two-dimensional Ising model, we create a class that stores an  $m \times m$  array with a total number of  $N$  spins where  $N = m^2$ . The initial parameters consist of a range of temperatures at which to analyze, the increment at which we analyze, the size of  $M$ , and the number of sweeps to be conducted. We find a critical temperature of about 2.39 and thus choose a range of temperatures from 1.5 to 2.7 and increment by 0.01. All temperatures below 1.5 quickly reach an equilibrium of 1.0 or very close to it. All temperatures above 3.5 quickly reach an equilibrium of 0.0 or very close to it. We choose  $M$  to be 128 due to limited computational time. We choose 1000 sweeps because all equilibria occur before 1000 sweeps.

For each temperature, we call the function "sweep" which initiates an array of spins with each spin set to +1. At each temperature we find the average energy and average magnetization in equilibrium by "sweeping" though the array until the array reached equilibrium.

For each sweep of the array, we call the function "sweep". "sweep" generates  $N$  randomly selected  $x$  and  $y$  coordinates of the array and determines whether to flip the spin by calling the function "determine\_Flip". In order for "determine\_Flip" to decide whether or not flip the spin, it must learn what spins its neighbors have. Using periodic boundary conditions, we initialize the right, left, top, and bottom neighbors of the spin. We calculate what the change in energy would be if we decided to switch the sign of the spin  $dE$ . We then compare a randomly generated number to  $\beta$  where  $\beta = 1/kT$ . In actuality  $k$  is the Boltzmann constant, but this would cause the random number to be altered accordingly and thus we simplify the problem by setting  $m$  equal to 1. If the random number is less than difference between the two energy states  $e^{-\beta dE}$ , we flip the sign. Otherwise, the sign stays the same.

Next we find the magnetization and energy after each sweep of the array with the function "find\_Mag\_Energy". To determine the magnetization of the

array the function we use the following expression

$$\frac{\sum s_{i,j}}{m^2}$$

where  $s$  is the spin. To determine the energy of array the function uses periodic boundary conditions to sum of the energies of pair to pair interactions from nearest neighbors and divide by the number of spins:

$$\frac{-J \sum s_{ij}(s_{i+1,j} + s_{i,j+1})}{m^2}$$

These values are stored in the class. The functions “input\_Mag” and “input\_Eng” store the magnetism and energy of the spins into two arrays. Both the number of sweeps and the magnetism are written to a file while the number of sweeps and energy is written to a separate file.

After all sweeps are complete, the average equilibria magnetism and energy are calculated. To find the equilibrium value for each T, we compare each M or E in each sweep with the average value of previous 100 Ms or Es. If the M or E is smaller than our acceptable error (0.01), the M or E has reached equilibrium for this temperature. Thus, the average M or E is the equilibrium value of M or E. If the method does not return a value for the temperature, there are not enough sweeps to reach equilibrium. To calculate standard deviations we use the functions ”mag\_Std” and ”eng\_Std”. These functions sum up the squares of the values of (magnetism/energy) - (the avg magnetism/energy) for each of the 100 values before equilibrium. The standard deviations are the square roots of these two sums divided by N-1:

$$\frac{\sqrt{\sum (M - \langle M \rangle)^2}}{N - 1}$$

$$\frac{\sqrt{\sum (E - \langle E \rangle)^2}}{N - 1}$$

Lastly, we write the average magnetism and average energy for each temperature to a file. For each temperature we print out the average magnetism/energy and standard deviation in magnetism/energy.

## 4 Results

In Figure 1, we see that at low temperatures such as 1.5 and below, the equilibrium magnetization is 1 or slightly less than 1 and approaches its equilibrium quickly. At high temperatures such as 3.0 and above, the equilibrium approaches 0 quickly. Near the critical temperature (about 2.5), The magnetism after each sweep approaches lies at some equilibrium between 0 and 1. These temperatures take longer to reach an equilibrium.

In Figure 2, we see how quickly the energy after a each sweep approaches equilibrium. Near the critical temperature (at 2.5 on the graph), the energy

takes the longest to approach equilibrium, just as magnetism did. At higher temperatures, energies approach an average equilibrium of about -0.2. At lower temperatures, energies approach an average equilibrium of about -2.0.

In Figure 3, while there is noticeable fluctuation, we see how the average magnetization drops very quickly to arrive at 0 at the observed critical temperature of 2.5.

In Figure 4, we see how the average energy increases from near -2 faster and faster until around the critical temperature the slope lessens and evens out. In Figure 5 we observe how the average energy approaches -0.2. This curve resembles the form of  $\tanh(x)$ .