AA 2021/2022 Computational Methods
Lesson 9 - Monte Carlo Methods

Almaz Khabibrakhmanov[*], Mario Galante[†], Alexandre Tkatchenko

Theoretical Chemical Physics group, FSTM,
Campus Limpertsberg, University of Luxembourg

Part 1 - Area of a Circle

1. Implement the pseudocode below and execute it.

Algorithm 1 Monte Carlo integration

Set $N = 1000$, the counters `n_in = 0, n_out = 0` and the empty arrays `x_in, y_in, x_out, y_out`.
for $i = 0, 1, ..., N-1$ **do**
    Generate random x and y within the interval $[-1, 1]$ using `numpy.random.uniform`
    if $\sqrt{x^2 + y^2} \leq 1$ **then**
        `n_in += 1`
        Append x and y to `x_in` and `y_in`
    else
        `n_out += 1`
        Append x and y to `x_out` and `y_out`
    end if
end for
$S_{circle} = S_{square} \cdot n_{in}/(n_{in} + n_{out})$

2. Plot your results: first draw a unit circle, then plot the points $(x_{in},\ y_{in})$ and $(x_{out},\ y_{out})$ on the same figure (in different colors).

3. Express the value of π from the obtained S_{circle}. Is it close to the real value? Do you think the current sampling is dense enough for our purpose?

4. Execute your code 5 times and calculate the average value of π for this N.

5. Try to gradually increase the density of points taking $N = 10^4, 10^5, 10^6, 10^7$. Check the calculated value of π and create the scatter plot of the sampling points. For every N repeat the calculations 5 times and obtain the corresponding average value π_N.

6. Compute the deviation from the exact result $\varepsilon = |\pi - \pi_N|$. Perform a log-log plot of this error as a function of N and show that the data can be fit to a straight line of a slope $-1/2$.

Part 2 - The Solution of 1D Ising Model

The goal of this part is to explore the Ising model through the Metropolis algorithm. Consider a chain made of N spins σ_i that each take the value 1 or -1 marking if the spin at site i is up or down. We will assume that the chain is *periodic*, therefore identifying $\sigma_{N+1} = \sigma_1$. The nearest-neighbour Ising model assumes that interactions only exists for the adjacent spins; the energy of a particular spin configuration $\{\sigma_i\}_{i=1}^N$ is then given by:

[*]almaz.khabibrakhmanov@uni.lu
[†]mario.galante@uni.lu

$$E(\{\sigma\}) = -J\sum_{i=1}^{N}\sigma_i\sigma_{i+1}, \quad J > 0, \tag{1}$$

and its magnetization is:

$$M(\{\sigma\}) = \sum_{i=1}^{N}\sigma_i. \tag{2}$$

The Metropolis algorithm is a widely used Monte Carlo method that allows to simulate the dynamics of the relaxation of a given spin configuration towards thermal equilibrium. Below is the pseudocode describing how to use the Metropolis algorithm to solve the 1D Ising model.

Algorithm 2 Monte Carlo equilibration of 1D Ising model

Generate a random initial configuration of spins $\{\sigma_i\}_{i=1}^{N}$ using `numpy.random.choice`
Calculate the energy E and magnetization M of this configuration. You may want to write a separate function for that. Remember also about the periodic boundary conditions (PBC) at the ends of the chain.
for $j = 0, 1, ..., it$ **do**
    Choose a random index k and flip the corresponding spin ($\sigma_k \to -\sigma_k$)
    Determine the nearest neighbours of the chosen spin (do not forget to check PBC!)
    Calculate the difference in energy ΔE generated by the spin flip.
    Calculate the associated transition probability $p = exp(-\Delta E/T)$.
    if $\Delta E \leq 0$ **then**
        Accept the spin flip, update the energy E and magnetization M
    end if
    if $\Delta E > 0$ **then**
        Generate a random number $r \in [0, 1]$. Use `numpy.random.uniform` for this.
        if $r \leq p$ **then**
            Accept the spin flip, update the energy E and magnetization M
        else
            Reject the spin flip and return it back to the previous state ($\sigma_k \to -\sigma_k$).
        end if
    end if
    Duplicate the values of E and M to the corresponding arrays to store the values at all steps.
end for

1. Set the following parameters: $N = 200,\ T = 2,\ J = 0.1,\ it = 10^3 \cdot N$ (high temperature limit).

2. Implement the Metropolis algorithm. Normalize the energy and magnetization per number of spins in chain.

3. Run your code 5 times. Every time plot the results: 1) energy vs. step number; 2) magnetization vs. step number; 3) the scatter plot of spins in the chain. Observe the equilibration of the system. Also plot the analytical results: $\langle E\rangle = -J \cdot tanh(J/T),\ \langle M\rangle = 0$. Are these values achieved, as expected, in your simulations?

4. Now run the simulations in the low temperature limit, change parameters: $T = 0.1,\ J = 1$. Repeat (3). How did scatter plot of spins change? Do your simulations agree with the analytical predictions about the possibility of ferromagnetic transition within the 1D Ising model? If there is a difference, can you figure out the possible cause?