# HW3

# Ising Model Simulation

## 陳凱騫

### 2024-12-15

# **Table of contents**

| 簡介      |  |  | <br> |  |  |  |  |  |  |  |  |  |  |  | <br> |  |  |  | - |
|---------|--|--|------|--|--|--|--|--|--|--|--|--|--|--|------|--|--|--|---|
| R 模擬與分析 |  |  | <br> |  |  |  |  |  |  |  |  |  |  |  | <br> |  |  |  | • |
| 模型設定 .  |  |  | <br> |  |  |  |  |  |  |  |  |  |  |  | <br> |  |  |  | • |
| R 程式碼區段 |  |  | <br> |  |  |  |  |  |  |  |  |  |  |  | <br> |  |  |  | • |

## 簡介

2D Ising 模型是統計物理中經典的模型,用於研究磁性材料的相變現象。本次模擬將使用**Metropolis** 演算法,來分析能量、磁化率、比熱等隨溫度變化的行為。

## R 模擬與分析

#### 模型設定

• Ising Hamiltonian 定義如下:

$$H = -J \sum_{\langle i,j \rangle} S_i S_j$$

- $S_i$ : 自旋,取值  $\pm 1$
- J: 自旋間的交互作用強度
- $\langle i,j \rangle$ : 最近鄰居的交互作用

### R 程式碼區段

#### 載入套件

library(ggplot2) library(gridExtra)

#### 初始參數與自旋配置

L <- 16 # 格點大小 J <- 1 # 交換作用強度

T\_seq <- seq(1.5, 3.5, length.out = 40) # 溫度範圍

```
steps_eq <- 1000 # 平衡步數
steps_mc <- 1000 # 蒙地卡羅步數

initial_state <- function(L) {
    matrix(sample(c(-1, 1), L * L, replace = TRUE), L, L)
}
```

### Metropolis 演算法

```
calc energy <- function(spin matrix, J) {
 E <- 0
 L <- nrow(spin_matrix)
 for (i in 1:L) {
 for (j in 1:L) {
   Si <- spin_matrix[i, j]
   nb <- spin_matrix[(i %% L) + 1, j] +
       spin_matrix[(i - 2) \%\% L + 1, j] +
       spin_matrix[i, (j %% L) + 1] +
       spin_matrix[i, (j - 2) %% L + 1]
   E <- E - J * Si * nb
 }
 return(E / 2)
metropolis_step <- function(spin_matrix, beta, J) {
 L <- nrow(spin_matrix)
 for (k in 1:(L * L)) {
 i <- sample(1:L, 1)
 j \leftarrow sample(1:L, 1)
  Si <- spin_matrix[i, j]
  nb <- spin_matrix[(i %% L) + 1, j] +
     spin_matrix[(i - 2) \%\% L + 1, j] +
     spin_matrix[i, (j %% L) + 1] +
     spin_matrix[i, (j - 2) %% L + 1]
  dE <- 2 * J * Si * nb
  if (dE \le 0 || runif(1) \le exp(-dE * beta)) {
   spin_matrix[i, j] <- -Si
}
 return(spin_matrix)
```

#### 模擬與結果

```
simulate_ising <- function(L, T_seq, steps_eq, steps_mc, J) {
    E_list <- M_list <- numeric(length(T_seq))
    for (t in 1:length(T_seq)) {
        T <- T_seq[t]
        beta <- 1 / T
        spin_matrix <- initial_state(L)
        for (step in 1:steps_eq) spin_matrix <- metropolis_step(spin_matrix, beta, J)
        E <- M <- 0
```

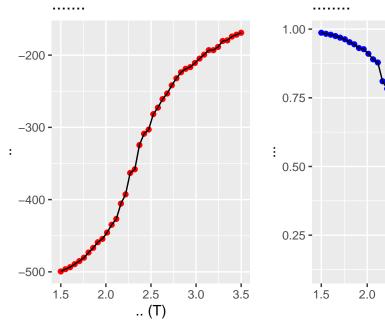
```
for (step in 1:steps_mc) {
    spin_matrix <- metropolis_step(spin_matrix, beta, J)
    E <- E + calc_energy(spin_matrix, J)
    M <- M + abs(sum(spin_matrix))
    }
    E_list[t] <- E / steps_mc
    M_list[t] <- M / (steps_mc * L * L)
}
return(data.frame(T = T_seq, Energy = E_list, Magnetization = M_list))
}
results <- simulate_ising(L, T_seq, steps_eq, steps_mc, J)</pre>
```

## 視覺化

```
p1 <- ggplot(results, aes(x = T, y = Energy)) +
    geom_point(color = "red") + geom_line() +
    labs(title = "能量隨溫度變化", x = "溫度 (T)", y = "能量")

p2 <- ggplot(results, aes(x = T, y = Magnetization)) +
    geom_point(color = "blue") + geom_line() +
    labs(title = "磁化率隨溫度變化", x = "溫度 (T)", y = "磁化率")

grid.arrange(p1, p2, ncol = 2)
```



3.0

3.5

2.5

.. (T)