

Introdução à Física Computacional II (4300318)

Prof. André Vieira
apvieira@if.usp.br
Sala 3120 – Edifício Principal

Aula 14

Aplicação: o modelo de Ising

O modelo e suas interações

- O modelo é definido em uma rede regular, sobre cujos sítios situam-se “spins” (átomos com um momento magnético intrínseco) que podem apontar apenas “para cima” e “para baixo” ao longo de uma única direção.
- Para tentar descrever o ferromagnetismo dos ímãs, vamos supor interações que favoreçam o alinhamento de spins vizinhos com a mesma orientação. Vamos escrever a energia de interação de um par de spins vizinhos como

$$E_{\text{par}} = -J s_1 s_2, \quad J > 0, \quad s_1, s_2 \in \{-1, 1\}.$$

O modelo e suas interações

- O modelo é definido em uma rede regular, sobre cujos sítios situam-se “spins” (átomos com um momento magnético intrínseco) que podem apontar apenas “para cima” e “para baixo” ao longo de uma única direção.
- Com essa escolha, e incluindo uma interação dos spins com um campo magnético externo, a energia total do modelo, para uma certa configuração $\{s_i\}$ de spins, é escrita como

$$E(\{s_i\}) = -J \sum_{\text{pares}} s_i s_j - B \sum_{\text{sítios}} s_i.$$

O modelo e suas interações

$$E(\{s_i\}) = -J \sum_{\text{pares}} s_i s_j - B \sum_{\text{sítios}} s_i.$$

- Por que esse modelo poderia descrever o ferromagnetismo? Em campo externo nulo ($B=0$), a configuração que minimiza a energia corresponde a todos os spins apontando para cima ou todos os spins apontando para baixo, o que sugere a possibilidade de que o sistema possua uma magnetização espontânea em baixas temperaturas, como um ímã permanente.

O modelo e suas interações

$$E(\{s_i\}) = -J \sum_{\text{pares}} s_i s_j - B \sum_{\text{sítios}} s_i.$$

- A magnetização por spin é definida como

$$m = \left\langle \frac{1}{N} \sum_{\text{sítios}} s_i \right\rangle,$$

sendo N o número de sítios.

- A magnetização espontânea é definida como

$$m_0 = \lim_{B \rightarrow 0} m.$$

O modelo em 1D

- Em uma rede unidimensional, podemos representar uma configuração dos spins, por exemplo, para uma rede com $L=10$ spins, como



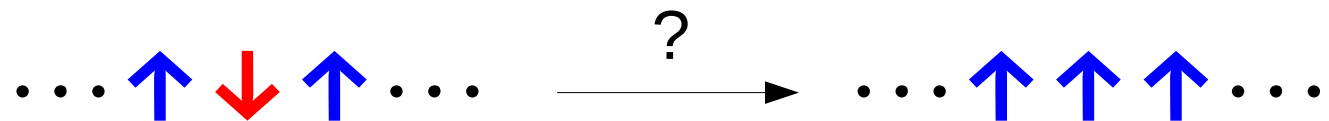
- A energia associada a uma configuração qualquer seria dada por

$$E(\{s_i\}) = -J \sum_i s_i s_{i+1} - B \sum_i s_i.$$

- Em particular, a energia da configuração na figura seria $E = -2J + 2B$ (com condições periódicas de contorno; o último spin interage com o primeiro).

O modelo em 1D

- A dinâmica da simulação envolve construir uma cadeia de Markov de estados, e vamos passar de um estado ao outro por um conjunto de possíveis movimentos que correspondem a inverter um único spin por vez, como ilustrado na figura abaixo:



- Se vamos propor a inversão do spin no i -ésimo sítio, s_i , a parte relevante da energia é aquela que envolve esse spin, ou seja,

$$-J s_{i-1} s_i - J s_i s_{i+1} - B s_i = -[J(s_{i-1} + s_{i+1}) + B] s_i$$

O modelo em 1D

- Se vamos propor a inversão do spin no i -ésimo sítio, s_i , a parte relevante da energia é aquela que envolve esse spin, ou seja,

$$-J s_{i-1} s_i - J s_i s_{i+1} - B s_i = -[J(s_{i-1} + s_{i+1}) + B] s_i$$

- Logo, se invertermos esse spin, a variação da energia será

$$\Delta E = \underbrace{\{-[J(s_{i-1} + s_{i+1}) + B](-s_i)\}}_{\text{Energia depois}} - \underbrace{\{-[J(s_{i-1} + s_{i+1}) + B]s_i\}}_{\text{Energia antes}}$$

$$\boxed{\Delta E = 2[J(s_{i-1} + s_{i+1}) + B]s_i}$$

O modelo em 1D

$$\Delta E = 2 \left[J (s_{i-1} + s_{i+1}) + B \right] s_i$$

- Em nosso código para a simulação, vamos tirar proveito do fato de que a variação da energia só depende do estado do spin a ser invertido e da soma dos estados dos seus vizinhos.

Aceleramos o cálculo definindo uma matriz de fatores de Boltzmann

$$b(s_i, S_v) = \exp(-\Delta E/T) = \exp \left[-2 (JS_v + B) s_i / T \right]$$

- Também implementamos condições de contorno periódicas, ou seja, tratamos os spins nos dois extremos como vizinhos entre si.

O modelo em 1D

- Vamos primeiro calcular a curva da magnetização como função do campo, a uma temperatura constante, e averiguar se surge uma magnetização espontânea quando a temperatura é suficientemente baixa.

Exemplo 1a

```
from math import exp,sqrt
from numpy import random,empty,full,sum
import matplotlib.pyplot as plt

# Implementa o cálculo da magnetização como função do campo externo,
# a uma temperatura fixa, para o modelo de Ising em 1D, através de
# uma simulação de Monte Carlo com o algoritmo de Metropolis.
# O estado de cada spin é representado por uma variável s[i] que pode
# assumir valores -1 ou +1. São utilizadas condições de contorno
# periódicas.

# Parâmetros
L = 2**7          # Número de spins
T = 1.0           # Temperatura da simulação
J = 1.0           # Constante da interação entre spins
Bmax = 1.0        # Valor máximo do campo
Bmin = -1.0       # Valor mínimo do campo
nB = 21           # Número de valores do campo entre Bmin e Bmax
Passos = 10000    # Número total de passos de Monte Carlo a realizar
Equilibra = 1000  # Número de passo de Monte Carlo para equilíbrio

# Criamos listas para armazenar os valores do campo (em ordem
# decrescente), da magnetização por spin e do erro em sua estimativa.
dB = (Bmax-Bmin)/(nB-1)
B_lista = [(Bmax - iB*dB) for iB in range(nB)]
m_sim, m_erro = [], [] # Magnetização por spin e seu erro

# Para implementar as condições de contorno periódicas, vamos criar
# vetores com os vizinhos de cada sítio.
ve, vd = empty(L,int), empty(L,int)
for i in range(L):
    ve[i] = i-1      # Vizinho à esquerda
    if i == 0:       # Correção para o primeiro spin
        ve[i] = L-1
    vd[i] = i+1      # Vizinho à direita
    if i == L-1:     # Correção para o último spin
        vd[i] = 0
```

Exemplo 1a

```
# Simulação. Definimos uma função para implementar um único passo
# de Monte Carlo. Buscamos coletar médias apenas após a estabilização.
s = full(L,1) # Spins inicialmente estão todos no estado +1

def passo_MC(s,boltz): # Função que implementa um passo de Monte Carlo
    z = random.rand(L) # Sorteamos L números aleatórios
    # Damos a cada spin a chance de mudar de estado, mas aqui fazemos
    # isso aleatoriamente para minimizar correlações entre mudanças.
    for i in random.permutation(L):
        si = s[i] # Estado do i-ésimo spin
        sv = s[ve[i]] + s[vd[i]] # Soma dos estados dos vizinhos
        if z[i] < boltz[si,sv]: # Mudança foi aceita?
            s[i] *= -1 # Registramos a mudança
    return s
```

Especialmente em 1D, as interações entre spins podem gerar correlações entre movimentos propostos sequencialmente. Por isso, escolhemos os spins no passo de MC aleatoriamente.

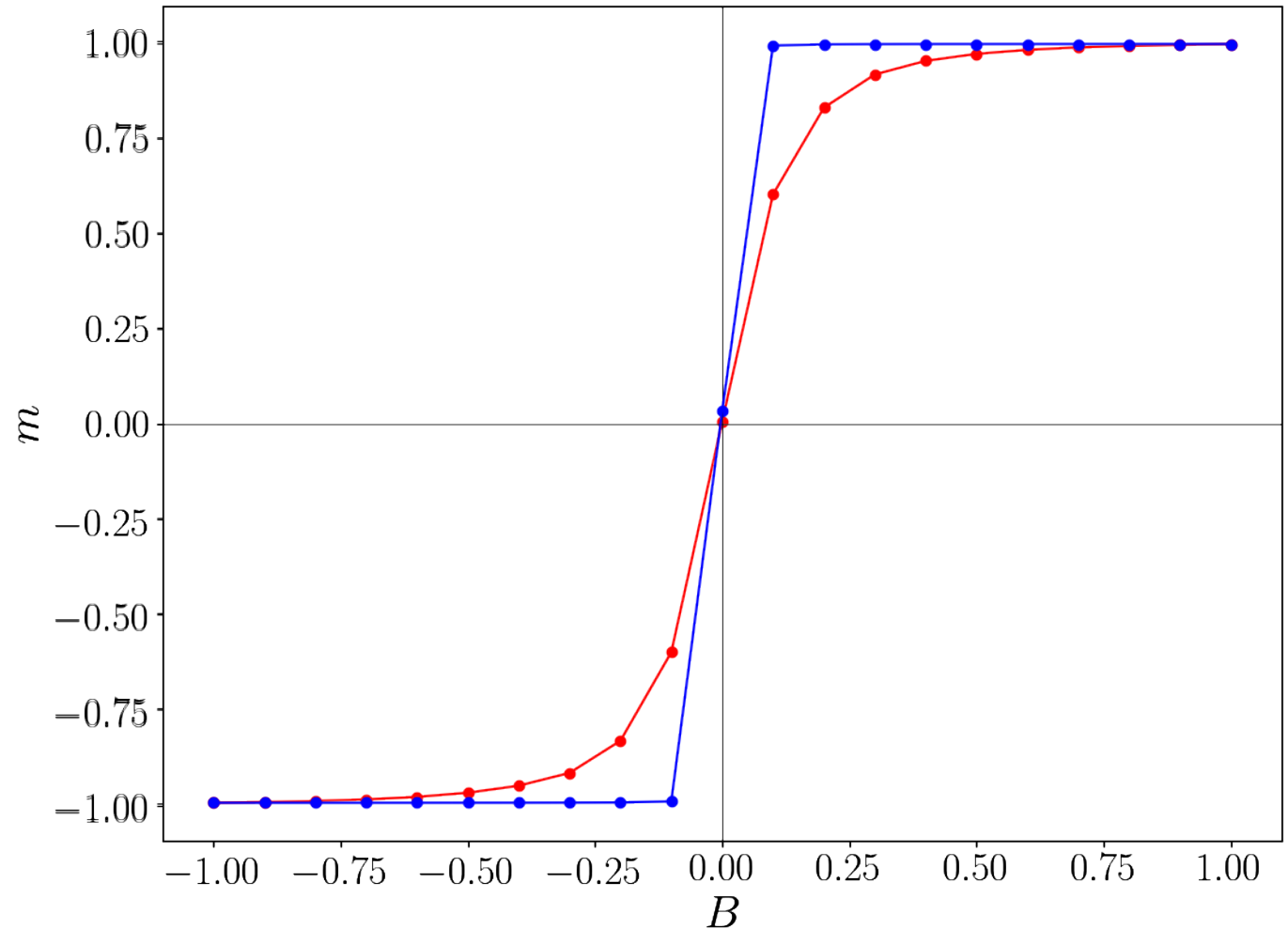
Exemplo 1a

```
medidas = Passos - Equilibra      # Número de medidas no equilíbrio
for iB in range(nB):
    B = B_lista[iB]
    # A variação de energia quando um spin é invertido depende apenas do campo
    # e da soma dos estados dos seus vizinhos. Vale a pena criar uma lista
    # para armazenar os valores possíveis do fator de Boltzmann correspondente.
    boltz = full([3,5],0.0)
    for si in range(-1,2):        # 0 estado do spin vai de -1 a 1
        for sv in range(-2,3):    # Soma dos estados dos vizinhos vai de -2 a 2
            dE = 2*(J*sv + B)*si  # Variação da energia na inversão do spin
            boltz[si,sv] = exp(-dE/T)
    # Primeiro aguardamos os passos de equilibração
    for passo in range(Equilibra):
        s = passo_MC(s,boltz)
    # Vamos começar a calcular médias
    acumula_M, acumula_M2 = 0.0, 0.0
    # Os passos restantes servem para cálculo das médias
    for passo in range(medidas):
        s = passo_MC(s,boltz)
        M = sum(s)
        acumula_M += M            # Acumulamos a magnetização
        acumula_M2 += M**2        # Acumulamos a magnetização quadrática
    # Agora registramos as médias
    M_medio = acumula_M/medidas
    M2_medio = acumula_M2/medidas
    m_sim.append(M_medio/L)
    m_erro.append(sqrt((M2_medio-M_medio**2)/L**2/(medidas-1)))
```

A introdução dos vetores **ve** e **vd** de vizinhos de um sítio e da matriz **boltz** que guarda os fatores de Boltzmann é importante para acelerar a execução do código, que pode demorar bastante.

Exemplo 1a

$L=128$



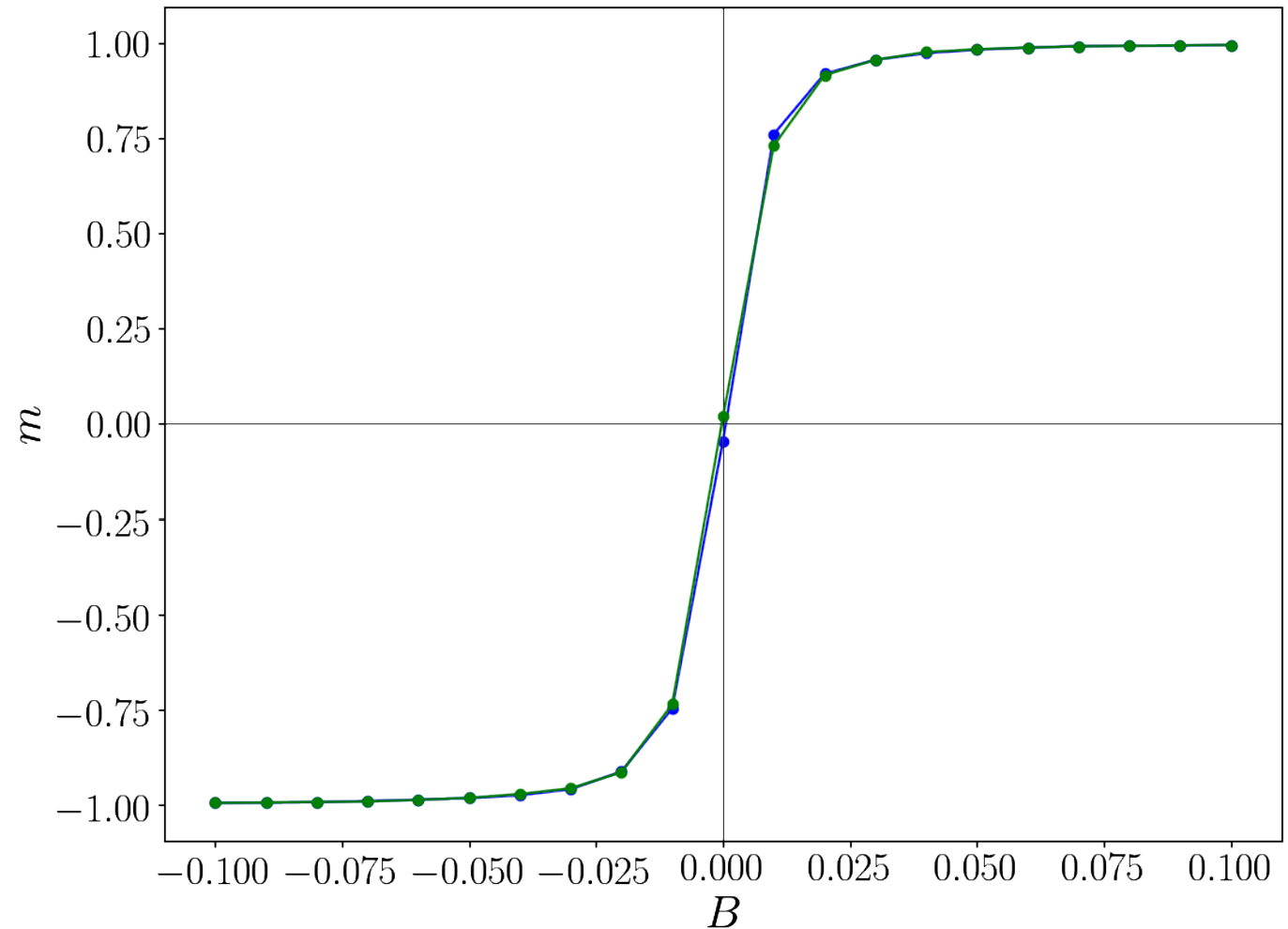
Em vermelho, a temperatura é $T=1.0$, enquanto em azul a temperatura é $T=0.5$. Haveria uma magnetização espontânea?

Exemplo 1a

$$T=0.5$$

$$L=128$$

$$L=256$$



Não parece haver diferença nas curvas quando duplicamos o tamanho do sistema, e a magnetização espontânea parece mais improvável.

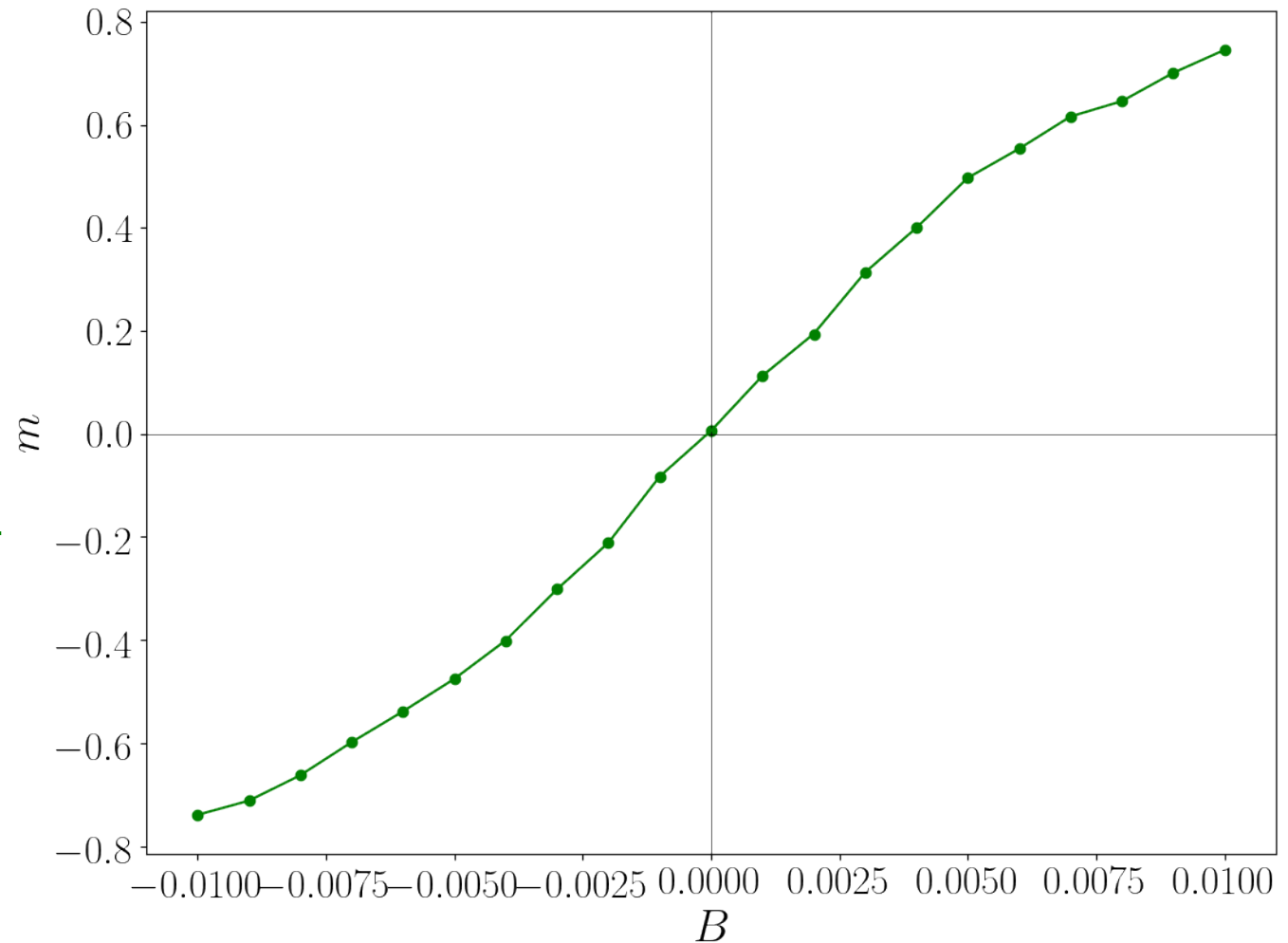
Exemplo 1a

$$T=0.5$$

$$L=256$$

$$\text{Passos} = 5 \times 10^4$$

$$\text{Equilibra} = 10^4$$



Não há de fato magnetização espontânea. O mesmo resultado é obtido em temperaturas mais baixas, e concorda com a solução analítica (que não discutiremos aqui).

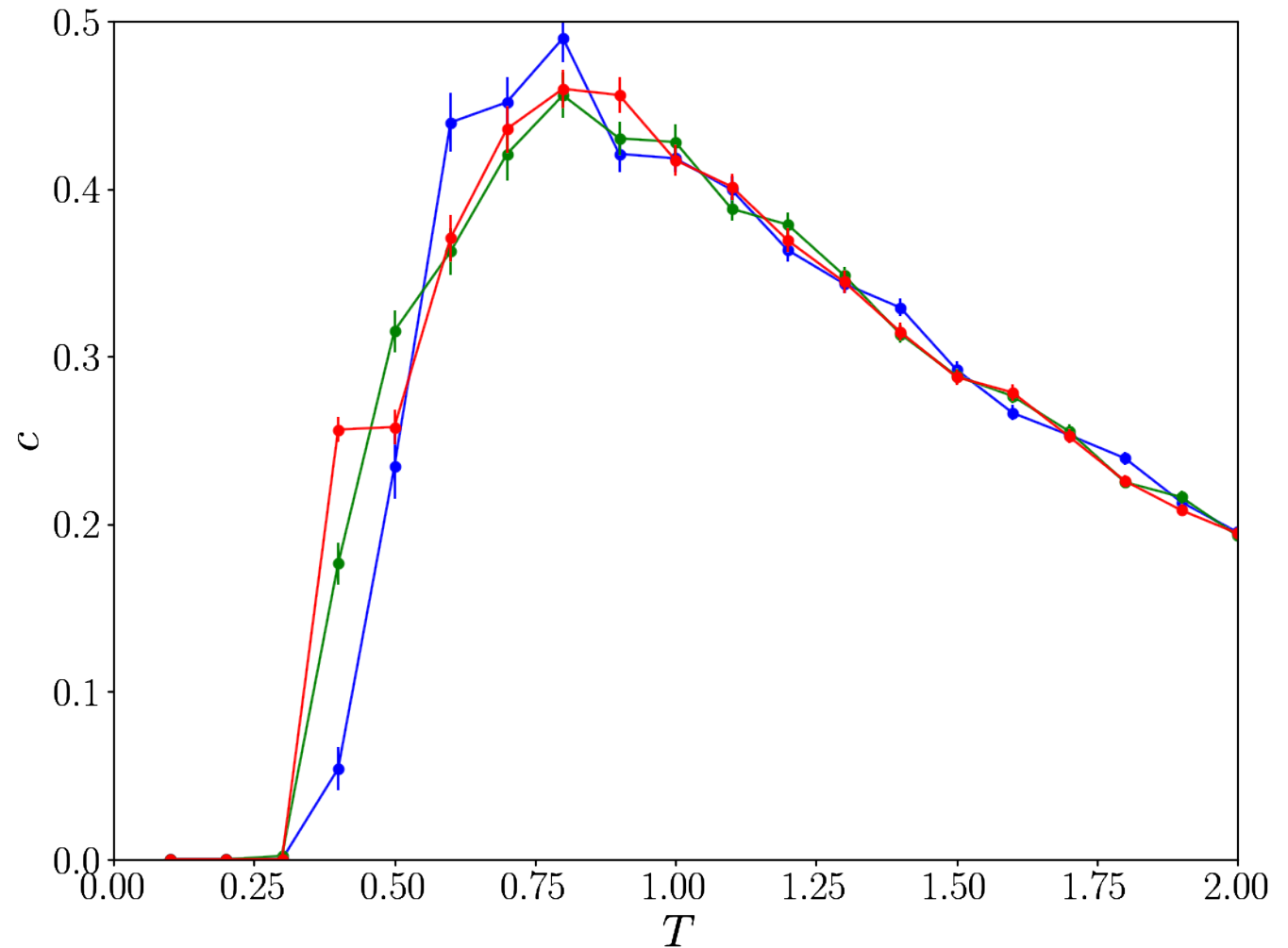
O modelo em 1D

- Agora vamos analisar o comportamento em função da temperatura, com campo nulo.
- Como em campo nulo há uma simetria entre configurações que se relacionam pela inversão de todos os spins, o valor esperado da magnetização média é nulo em qualquer sistema finito. Logo, para estimar o comportamento da magnetização no limite de tamanho infinito, utiliza-se nas simulações

$$m = \left\langle \left| \frac{1}{N} \sum_{\text{sítios}} s_i \right| \right\rangle.$$

Exemplo 1b

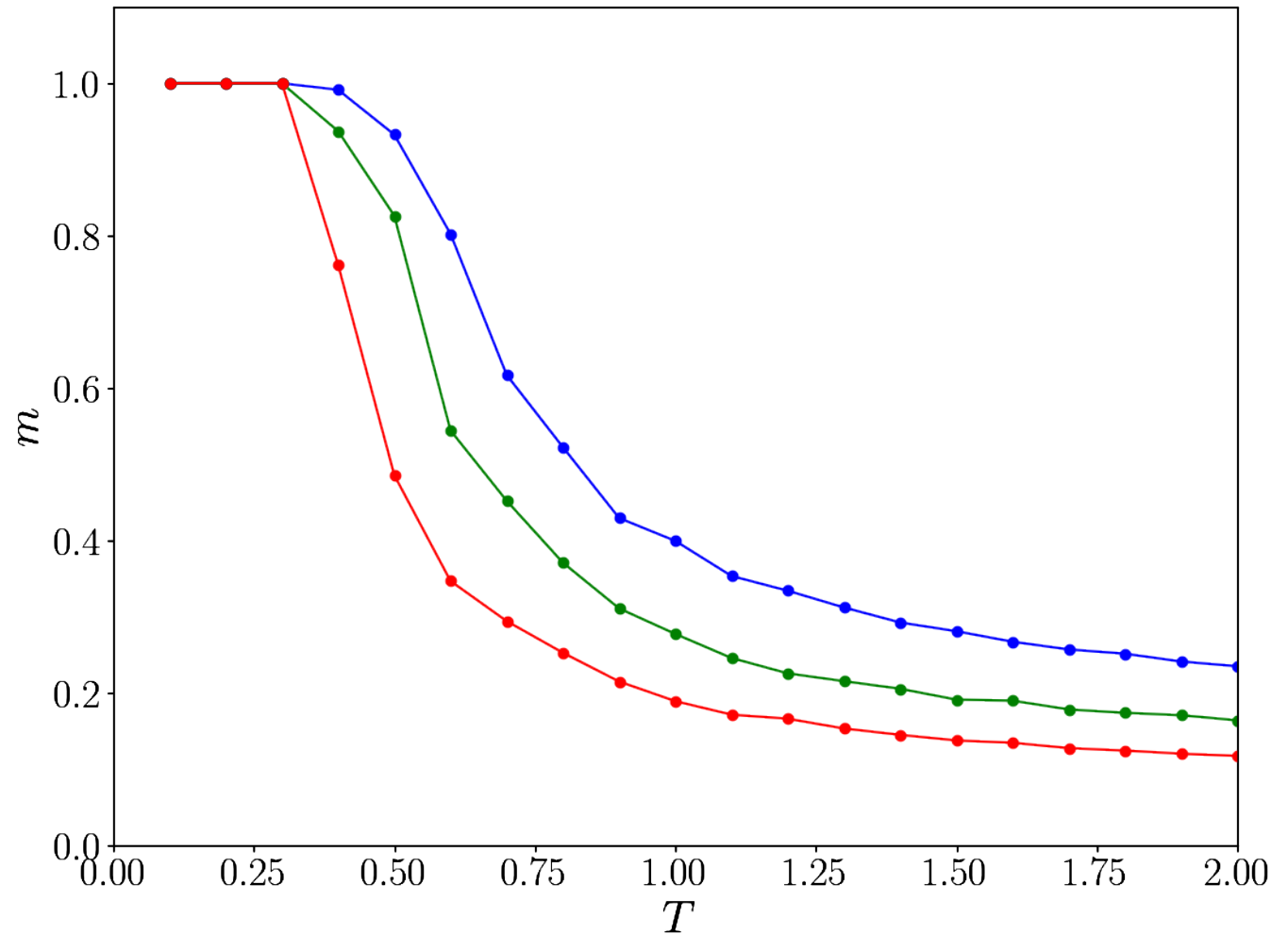
$B=0$
 $L=32$
 $L=64$
 $L=128$



O calor específico em função da temperatura não exibe uma dependência acentuada com o tamanho do sistema. O pico está associado à energia de interação J .

Exemplo 1b

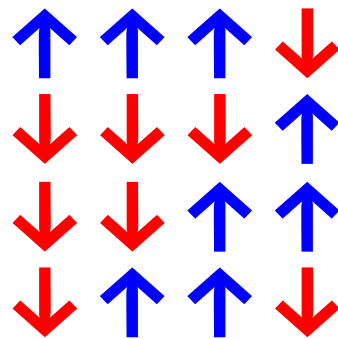
$B=0$
 $L=32$
 $L=64$
 $L=128$



Já a estimativa da magnetização claramente depende do tamanho do sistema. Na realidade, para sistemas suficientemente grandes essa estimativa tende a zero.

O modelo em 2D

- Em uma rede bidimensional, podemos representar uma configuração dos spins, por exemplo, para uma rede com $L=4$ spins, como

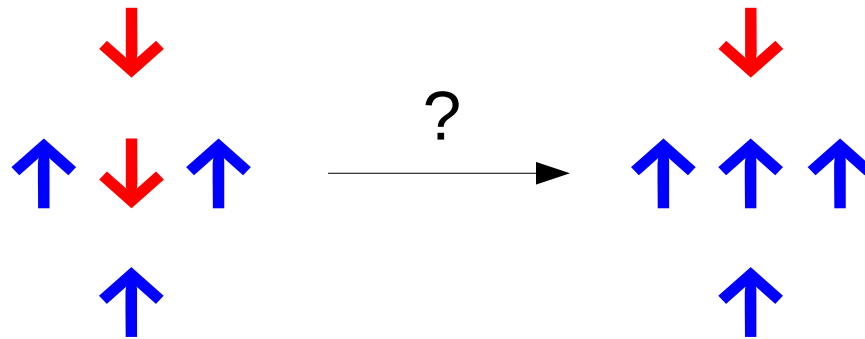


- A energia associada a uma configuração qualquer seria dada por

$$E(\{s_i\}) = -J \sum_{\text{pares}} s_i s_j - B \sum_{\text{sítios}} s_i.$$

O modelo em 2D

- Ao propormos inverter um spin durante a simulação, uma possível configuração local é:



- Se vamos propor a inversão do spin no i -ésimo sítio, s_i , a parte relevante da energia é aquela que envolve esse spin, ou seja,

$$-J s_i \left(\sum_{j \text{ viz. } i} s_j \right) - B s_i$$

O modelo em 2D

- Logo, se invertermos esse spin, a variação da energia será

$$\Delta E = \underbrace{\left\{ - \left[J \left(\sum_{j \text{ viz. } i} s_j \right) + B \right] (-s_i) \right\}}_{\text{Energia depois}} - \underbrace{\left\{ - \left[J \left(\sum_{j \text{ viz. } i} s_j \right) + B \right] s_i \right\}}_{\text{Energia antes}}$$

$$\Delta E = 2 \left[J \left(\sum_{j \text{ viz. } i} s_j \right) + B \right] s_i$$

Exemplo 2a

```
# Para implementar as condições de contorno periódicas, vamos criar
# uma matriz com os vizinhos de cada sítio.
v = empty((4,N),int)
for i in range(N):
    v[0,i] = i+1      # Vizinho à direita
    if (i+1)%L == 0:  # Correção para os spins na última coluna
        v[0,i] -= L
    v[1,i] = i+L      # Vizinho acima
    if v[1,i] > N-1:   # Correção para os spins na última linha
        v[1,i] -= N
    v[2,i] = i-1      # Vizinho à esquerda
    if i%L == 0:       # Correção para os spins na primeira coluna
        v[2,i] += L
    v[3,i] = i-L      # Vizinho abaixo
    if v[3,i] < 0:     # Correção para os spins na primeira linha
        v[3,i] += N

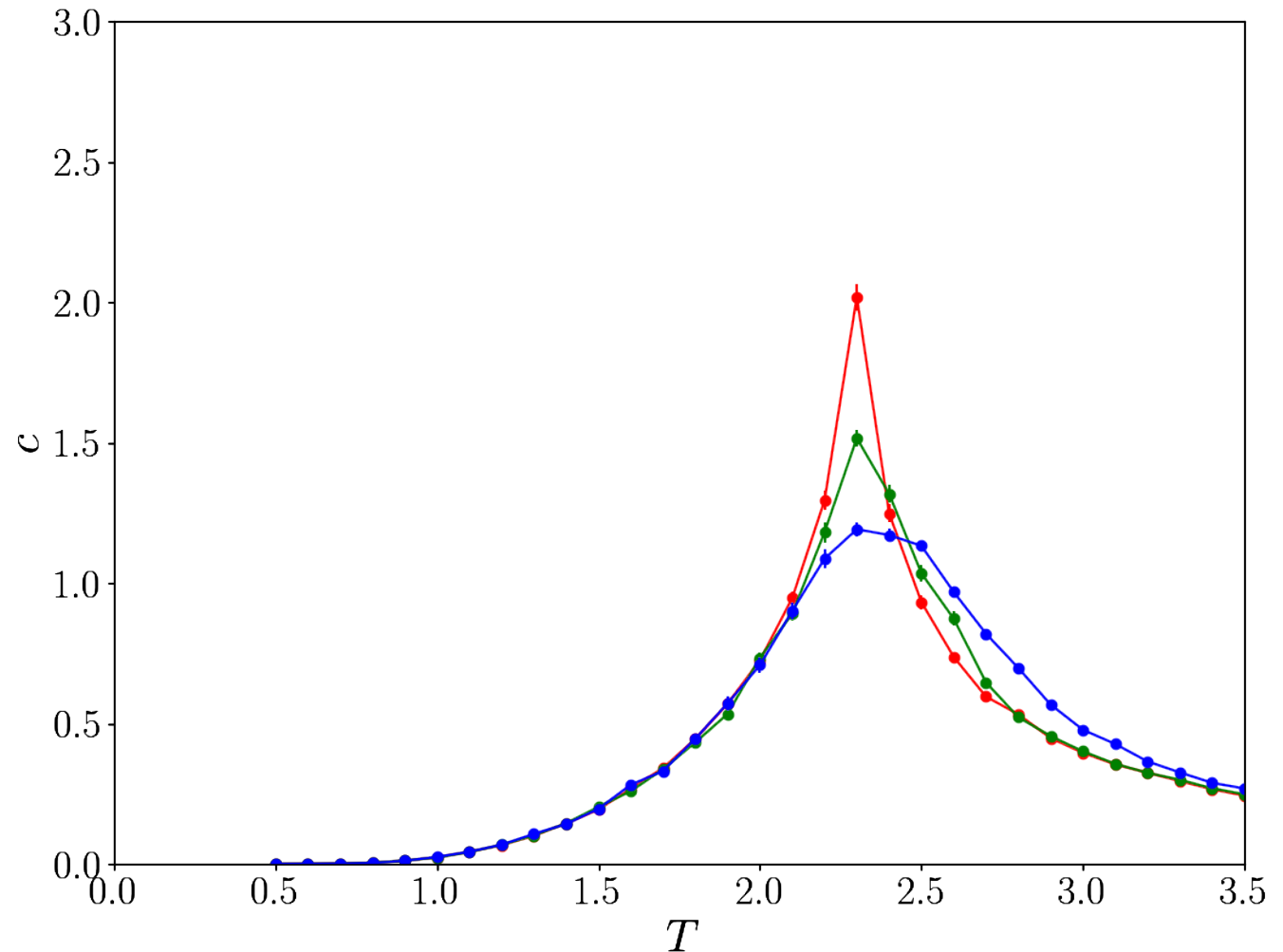
# Definimos uma função para calcular a energia de uma configuração
def Energia(s):
    soma = 0.0
    for i in range(N):
        soma += -(J*(s[v[0,i]]+s[v[1,i]]) + B)*s[i]
    return soma

# Simulação. Definimos uma função para implementar um único passo
# de Monte Carlo. Buscamos coletar médias apenas após a estabilização.
s = 1-2*random.randint(2, size=N)  # Estados iniciais aleatórios dos spins
```

As principais modificações no código envolvem a definição da matriz **v** de vizinhos (agora 4 por sítio) e o cálculo da energia. Note que ainda utilizamos um vetor para armazenar os estados dos spins.

Exemplo 2a

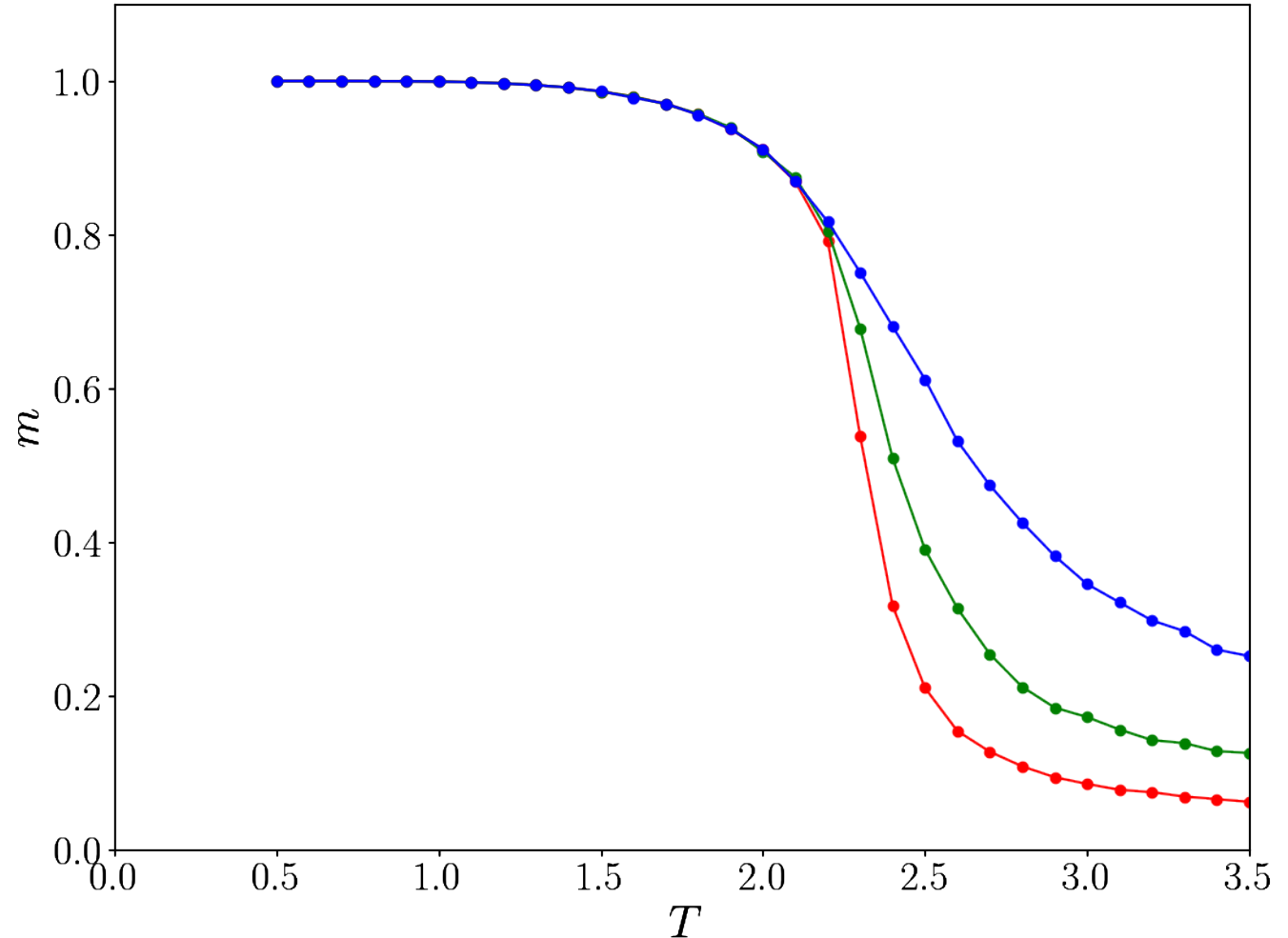
$L=8$
 $L=16$
 $L=32$



A rede contém $L \times L$ spins. Ao contrário do caso 1D, em 2D há uma variação perceptível do calor específico com o tamanho do sistema, especialmente em torno de $T=2.3$. Note como a altura do pico aumenta com L , o que se mantém indefinidamente.

Exemplo 2a

$L=8$
 $L=16$
 $L=32$



A estimativa da magnetização exibe duas tendências distintas com o aumento de L : para temperaturas abaixo de $T=2.3$, quase não há dependência, mas para temperaturas acima as curvas tendem a zero à medida que L aumenta.

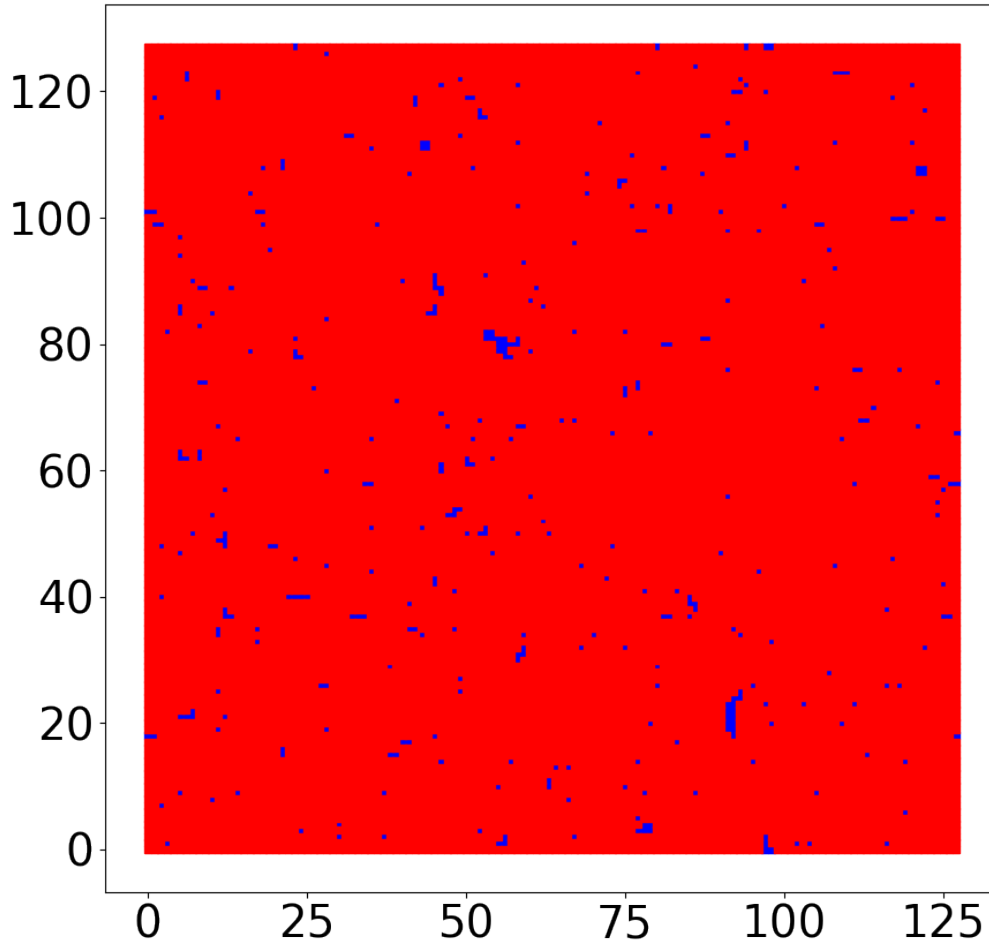
O modelo em 2D

- O que muda entre temperaturas mais altas e mais baixas?
- Para tentar entender, vamos observar as configurações dos spins em duas temperaturas.

Exemplo 2b

$$L = 128$$

$$T = 1.8$$

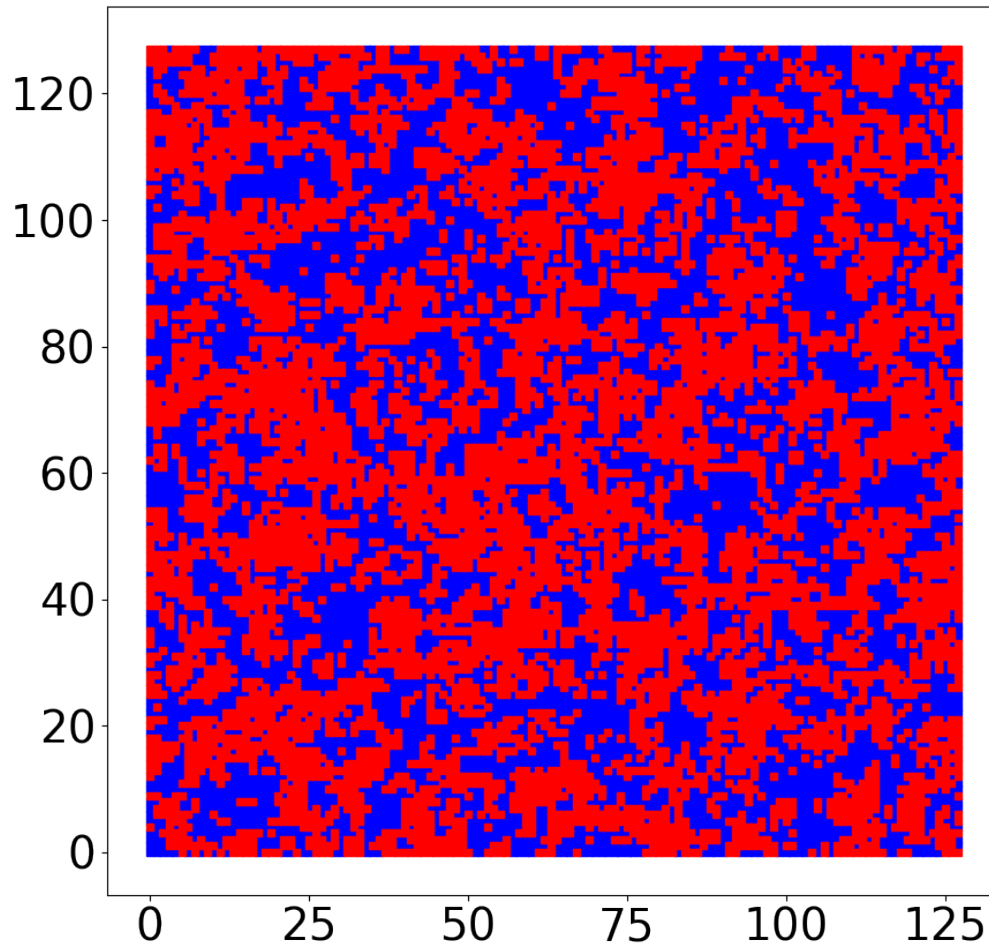


Em azul estão os spins “**para cima**”, e em vermelho, os spins “**para baixo**”. Note que aqui os spins estão majoritariamente para baixo. Em uma outra simulação, poderiam estar majoritariamente para cima.

Exemplo 2b

$$L = 128$$

$$T = 3.0$$



Em azul estão os spins “**para cima**”, e em vermelho, os spins “**para baixo**”. Note que agora há aproximadamente o mesmo número de spins em ambos os sentidos.

O modelo em 2D

- De fato há uma **transição de fase** entre uma fase ferromagnética e uma fase paramagnética em $T \simeq 2.3$. Isso é confirmado pela solução exata, e ocorre também em 3D (com outra T).
- A divergência do calor específico e o anulamento da magnetização nessa temperatura são condizentes com o que ocorre em ímãs reais (que, claro, não são 2D!).
- Existem técnicas mais avançadas que permitem a análise detalhada dessa transição de fase por meio de simulações.

Quinto EP

O prazo de entrega se encerra às 23h55 do dia
1 de julho.