

# Quinta série de exercícios

## Tópicos de Mecânica Estatística

### Transições de Fases - 2020

Renata Biaggi Biazzi

22 de maio de 2020

## 1

Com base no Hamiltoniano:

$$\mathcal{H} = -J \sum_{(i,j)} \sigma_i \sigma_j$$

Podemos escrever a função de partição:

$$\mathcal{Z} = \sum_{\sigma_i} \prod_{B_{ij}} e^{\beta J \sigma_i \sigma_j}$$

com  $K = \beta J$  e  $B_{ij}$  sendo cada vizinho mais próximos  $ij$  da rede. O número total de ligações (entre vizinhos mais próximos) é  $B = Nq/2$ , em que  $q$  é a coordenação da rede e  $N$  o número de sítios. As únicas possibilidades de valores para o termo  $e^{K\sigma_i\sigma_j}$  são:  $e^K$  ou  $e^{-K}$ , a depender de  $\sigma_i\sigma_j = \pm 1$ . Então:

$$e^K = \frac{1}{2} (e^K + \sigma_i \sigma_j e^K) \quad \text{se } \sigma_i \sigma_j = 1$$

$$e^{-K} = \frac{1}{2} (e^{-K} - \sigma_i \sigma_j e^{-K}) \quad \text{se } \sigma_i \sigma_j = -1$$

então:

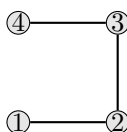
$$e^{K\sigma_i\sigma_j} = \frac{1}{2} (e^K + e^{-K}) + \frac{\sigma_i\sigma_j}{2} (e^K - e^{-K}) = \cosh K + \sigma_i\sigma_j \sinh K = \cosh K (1 + \sigma_i\sigma_j \tanh K)$$

chamando  $t = \tanh K$ , na rede quadrada  $q = 4$  temos:

$$\mathcal{Z} = (\cosh K)^{\frac{Nq}{2}} \sum_{\sigma_i} \prod_{B_{ij}} (1 + t\sigma_i\sigma_j) = (\cosh K)^{\frac{N}{2}} \sum_{\sigma_i} (1 + t\sigma_1\sigma_2)(1 + t\sigma_2\sigma_3)\dots(1 + t\sigma_N\sigma_1)$$

$$= (\cosh K)^{\frac{N}{2}} (2^N) [1 + c_4 t^4 + c_6 t^6 + c_8 t^8 + \dots]$$

O termo  $2^N$  vem de que temos  $N$  spins  $\sigma_i$  e cada um pode ter 2 valores  $\pm 1$ . Então a soma sobre todas as configurações possíveis equivale a  $2^N$  possibilidades totais. As contribuições que não se anulam são aquelas em que há ligações fechadas. Ou seja, que cada sítio  $\sigma_i$  está conectado com pelo menos dois outros sítios, de forma que aparecerá um número par de vezes, e terá um expoente par. Na rede 2d com  $q = 4$ , só é possível fazer ligações fechadas se consideramos pelo menos 4 ligações, pois com menos do que isso a rede fica aberta:



Assim, a série começa em  $t^4$  e não terá termos ímpares porque qualquer soma sobre as configurações em que pelo menos um expoente de um spin é ímpar irá se anular (haverá sítios abertos). Vejamos porque:

No primeiro exemplo, com ambos os spins com expoente 1 ou somente com um spin:

$$\sum_{i=\pm 1} \sigma_i^1 = 1 - 1 = 0$$

$$\sum_{i=\pm 1; j=\pm 1} \sigma_i^1 \sigma_j^1 = (1.1) + (1.-1) + (-1.1) + (-1.-1) = 1 - 1 - 1 + 1 = 0$$

então para outros casos:

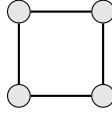
$$\sum_{i=\pm 1; j=\pm 1; k=\pm 1} \sigma_i^2 \sigma_j^2 \sigma_k^1 = \sum_{k=\pm 1} \sigma_k^1 = 0$$

$$\sum_{i=\pm 1; j=\pm 1; k=\pm 1} \sigma_i^1 \sigma_j^2 \sigma_k^1 = \sum_{i=\pm 1; k=\pm 1} \sigma_i^1 \sigma_k^1 = 0$$

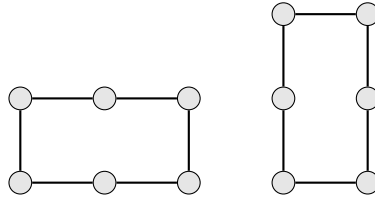
Ou seja, as únicas contribuições que não se anulam são para spins cujo expoente é par.

a)

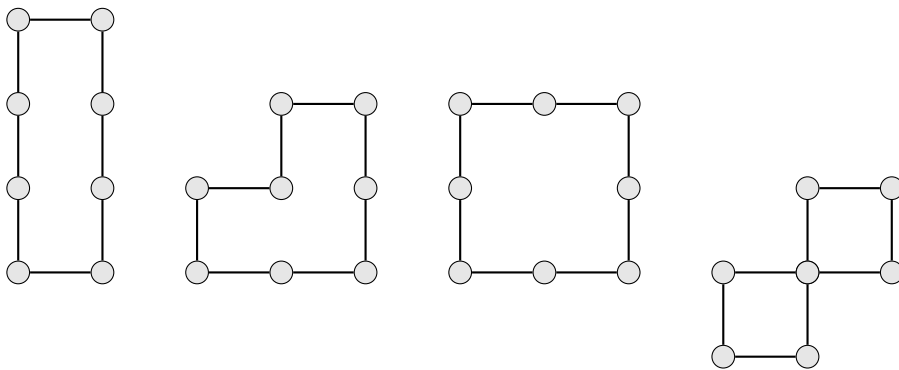
Para  $t^4$  o coeficiente  $c_4 = N$ , pois só há um único grafo possível com 4 ligações e ele pode aparecer em  $N$  posições diferentes na rede:



Para  $t^6$  só há um grafo possível com 6 ligações, porém rotacionando ele há duas formas possíveis de ele aparecer na rede. Assim, existem  $2N$  possibilidades de encontrar 6 ligações na rede, e  $c_6 = 2N$ :

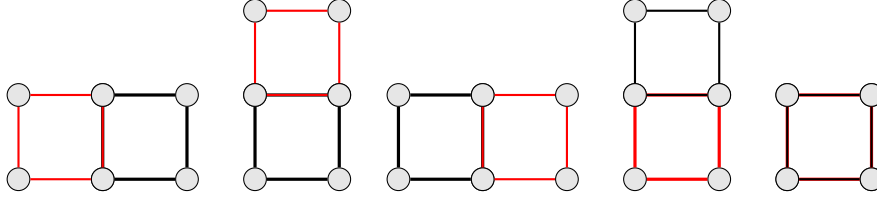


Para  $t^8$  os grafos possíveis são:



O primeiro grafo pode ser rotacionado, gerando  $2N$  possibilidades. O segundo grafo, ao ser rotacionado, gera  $4N$  possibilidades. O terceiro grafo é invariante a rotação, então só gera  $N$  possibilidades. Já o último grafo é mais complicado, vamos analisá-lo:

Esse último grafo deve ser entendido como dois quadrados independentes, cada um com quatro ligações, que devem aparecer juntos (para somar as 8 ligações) e cada um pode estar em qualquer lugar da rede. Assim, como cada um tem a possibilidade de aparecer em  $N$  lugares, e eles devem aparecer juntos, existem  $N^2$  possibilidades para ambos aparecerem. Entretanto, há 5 configurações que eles não podem estar (pois não concervam 8 ligações totais), elas são:



Devemos então subtrair de  $N^2$  as  $5.N$  configurações que não mantêm 8 conexões. Além disso, esses dois quadrados são indistinguíveis, então o número de configurações possíveis deve ser  $N^2 - 5N$  dividido por dois, para não contarmos duas vezes a mesma configuração. Assim, o número total de possibilidades de configurações para esse grafo de dois quadrados independentes (mantendo 8 conexões totais) é  $\frac{N^2 - 5N}{2}$ . Então, o coeficiente  $c_8$  é a soma dessas configurações todas:

$$c_8 = 2N + 4N + N + \frac{N^2 - 5N}{2} = 7N + \frac{N^2 - 5N}{2} = \frac{N(N + 9)}{2} = \frac{N(N - 9)}{2} + 9N$$

Assim, até o termo  $t^8$ , a série fica:

$$\mathcal{Z} = (\cosh K)^{\frac{N}{2}} 2^N \left[ 1 + Nt^4 + 2Nt^6 + \frac{N(N + 9)}{2}t^8 + \dots \right]$$

**b)**

A energia livre por spin no limite termodinâmico é calculada da seguinte forma:

$$g = \frac{-1}{\beta} \lim_{N \rightarrow \infty} \frac{1}{N} \log \mathcal{Z}$$

Sendo:

$$G = \log \mathcal{Z} = 2N \log [\cosh K] + N \log [2] + \log \left[ 1 + Nt^4 + 2Nt^6 + \frac{N(N + 9)}{2}t^8 + \dots \right]$$

No limite  $t \rightarrow 0$ , podemos expandir o  $\log$  e como a energia livre é extensiva, fazemos a aproximação de que os termos de ordem superior a  $N$  são descartados:

$$G = 2N \log [\cosh K] + N \log [2] + \left[ Nt^4 + 2Nt^6 + \frac{9N}{2}t^8 + \dots \right]$$

Então, a energia livre por sítio é:

$$-\beta g = \lim_{N \rightarrow \infty} \frac{1}{N} G = 2 \log [\cosh K] + \log [2] + t^4 + 2t^6 + \frac{9}{2}t^8 + \dots$$

em concordância com o esperado.

## 2

Para analisar o comportamento da função  $f(x)$  ao redor do que seria o ponto crítico  $x_c$ , iremos fazer a aproximação de que para  $x$  perto do ponto crítico, ou seja  $x/x_c \ll 1$  essa função pode ser escrita na forma:

$$f(x) \approx A(x_c - x)^{-\gamma}$$

Assim, utilizando a expansão binomial, escrevemos:

$$A(x_c - x)^b = \sum_{n=0,1}^{\infty} A_n x^n$$

com

$$A_n = \binom{b}{n} x_c^{b-n} = \frac{b!}{(b-n)!n!} x_c^{b-n}$$

O método da razão utiliza a seguinte relação: se uma série converge pelo método da razão, temos que seu raio de convergência  $r_c$  será dado pelo limite da razão dos coeficientes:

$$\frac{1}{r_c} = \lim_{n \rightarrow \infty} \frac{A_n}{A_{n-1}}$$

utilizaremos essa propriedade para extrapolar na nossa série com termos finitos. Fazendo:

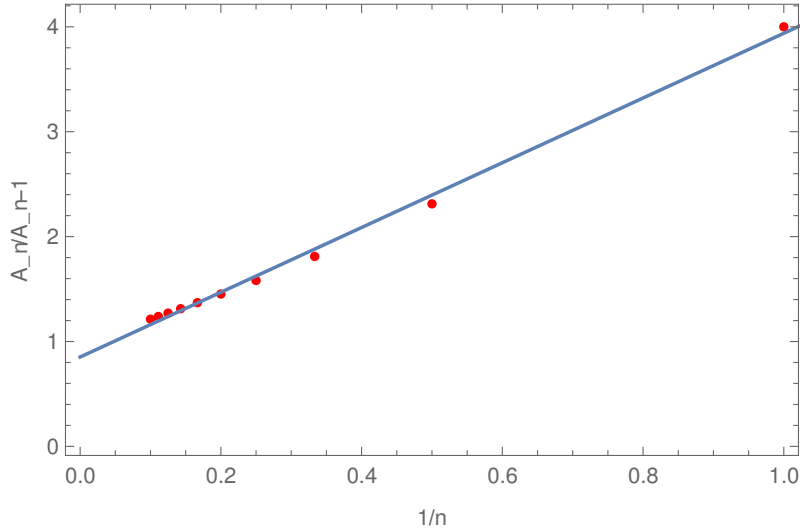
$$\frac{A_n}{A_{n-1}} = \frac{\binom{b}{n} x_c^{b-n}}{\binom{b}{n-1} x_c^{b-n+1}} = \frac{\frac{b!}{(b-n)!n!} x_c^{b-n}}{\frac{b!}{(b-n+1)!(n-1)!} x_c^{b-n+1}} = \frac{(b-n+1)}{n x_c} = \frac{1}{x_c} \left( \frac{b+1}{n} - 1 \right)$$

Iremos fazer um ajuste com  $y = \frac{A_n}{A_{n-1}}$  e  $x = 1/n$  nos pontos do tipo:

$$y = \alpha x + \beta$$

$$\frac{A_n}{A_{n-1}} = -\frac{b+1}{x_c} \frac{1}{n} + \frac{1}{x_c}$$

então teremos que pelo  $\lim_{n \rightarrow \infty} \frac{A_n}{A_{n-1}}: x_c = \frac{1}{\beta}$ . Já o parâmetro  $\gamma = -b$  é dado por  $-b = \alpha x_c + 1$ . Então temos o ajuste e os coeficientes:



$$\alpha = 3.08614 \quad \beta = 0.852466$$

Então obtemos a seguinte aproximação pelo método da razão:

$$\boxed{x_c \approx 1.1731} \quad \boxed{\gamma \approx 4.6203}$$

### 3

A série de baixas temperaturas para o modelo de Ising 2d  $\sigma = \pm 1$  pode também ser construída por argumentos geométricos. Da mesma forma que na série de altas temperaturas, começamos com a função de partição para o modelo de Ising 2d a campo nulo:

$$\mathcal{Z} = \sum_{\sigma_i} \prod_{B_{ij}} e^{\beta J \sigma_i \sigma_j}$$

com  $K = \beta J$  e  $B_{ij}$  sendo cada vizinho mais próximos  $ij$  da rede. O número total de ligações (entre vizinhos mais próximos) é  $B = Nq/2$ , em que  $q$  é a coordenação da rede e  $N$  o número de sítios. Nessas condições, o estado de energia mais baixo é dado pelo estado com todos os spins alinhados, seja  $\sigma = 1$  ou  $\sigma = -1$ , que é um estado degenerado de  $\mathcal{H}_{ef} = -2JN$ . Assim, temos a função de partição do estado fundamental:

$$\mathcal{Z}_{ef} = \sum_{\sigma_i} \prod_{Nq/2} e^{\beta J \sigma_i \sigma_j} = \sum_{\sigma_i} e^{\beta J 2N \sigma_i \sigma_j} = 2e^{\beta J 2N} = 2e^{K 2N}$$

Para baixas temperaturas, o termo  $K = \frac{J}{k_B T}$  é grande, então as configurações típicas podem ser enumeradas como pequenas flutuações ao redor do estado fundamental, mudando alguns spins de direção e contando a energia de ligação desses estados. Por exemplo, o próximo termo é definido se mudamos apenas a orientação de um spin. Nesse caso, na rede quadrada o custo para fazer isso é de  $8J$ , pois ele está conectado com quatro vizinhos mais próximos, e a diferença de energia com a mudança de orientação do spin com cada um dos seus vizinhos é de  $2J$ , pois a diferença de energia entre dois vizinhos próximos com essa mudança é calculada como:

$$\Delta \mathcal{H}_{ij}^{\sigma_i \rightarrow -\sigma_i} = -J \Delta(\sigma_i \sigma_j)^{\sigma_i \rightarrow -\sigma_i} = -J((- \sigma_i) \sigma_j - \sigma_i \sigma_j) = -J((-1) - 1) = 2J$$

Podemos mudar esse spin em  $N$  lugares, e chamaremos a função de partição dessa primeira flutuação de  $Z_4$  devido ao expoente 4 na mudança de energia, e temos:

$$Z_4 = N e^{2KN - 8K} = Z_{ef} N (e^{-2K})^4$$

O próximo termo é devido a mudarmos a orientação de dois spins vizinhos. O número de conexões com spins opostos que os dois têm juntos é 6, e podemos colocá-los em  $N$  lugares, assim como rotacioná-los em 2 orientações, então temos:

$$Z_6 = N 2 e^{2KN - 12K} = Z_{ef} 2N (e^{-2K})^6$$

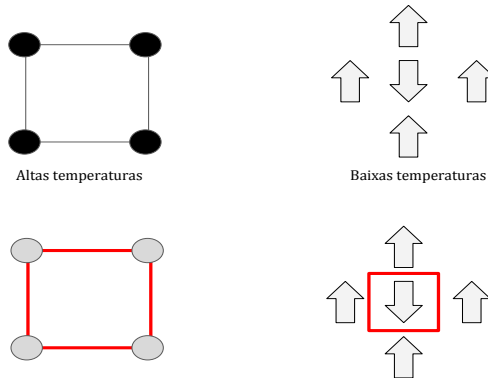
Com esses termos, já podemos começar a escrever a série de baixas temperaturas:

$$\mathcal{Z}_{\text{baixas}} = Z_{ef} [1 + N(e^{-2K})^4 + 2N(e^{-2K})^6 + \dots]$$

que tem uma forma muito semelhante com a de altas temperaturas:

$$\mathcal{Z}_{\text{altas}} = (\cosh K)^{\frac{N}{2}} 2^N \left[ 1 + N t^4 + 2N t^6 + \frac{N(N+9)}{2} t^8 + \dots \right]$$

Essa correspondência vem de um paralelo que podemos fazer da contagem dos grafos. No caso das séries de altas temperaturas, estamos contando todos as possibilidades de grafos fechados com  $x$  ligações. No caso da série de baixas temperaturas, estamos contando todas as conexões que um conjunto de spins que foi invertido têm com um conjunto de  $y$  spins vizinhos. O número de combinações possíveis para as  $x$  ligações é o mesmo que o número de todas as possibilidades para as  $y$  conexões. Um exemplo para simplificar essa ideia é visto nessa representação do termo de ordem quarta:



Em ambos os casos estamos contando todas as possibilidades em que essas linhas vermelhas podem aparecer, como as duas figuras em linhas vermelhas são geometricamente equivalentes na rede, o número de possibilidades em que elas podem aparecer é o mesmo. É dessa relação que vem a semelhança das séries para altas e baixas temperaturas. Sendo assim, com a mudança de variáveis:

$$t = \tanh K = e^{-2K^*}$$

que também pode ser escrita de outra maneira:

$$e^{2K} - e^{-2K} = \frac{\cosh(K^*)}{\sinh(K^*)} - \frac{\sinh(K^*)}{\cosh(K^*)} = \frac{\cosh^2(K^*) - \sinh^2(K^*)}{\cosh(K^*) \sinh(K^*)} = \frac{2}{\sinh(2K^*)}$$

$$\sinh(2K) \sinh(2K^*) = 1$$

resgatamos a equivalência das séries. A singularidade, conforme o trabalho de Hendrik Kramers e Gregory Wannier (H. A. Kramers and G. H. Wannier (1941). "Statistics of the two-dimensional ferromagnet", Physical Review 60: 252–262), ocorre quando essas variáveis são iguais, em que definimos a temperatura crítica como o valor "autodual" em que ocorre a transição entre a fase "altas temperaturas" e a fase "baixas temperaturas" em  $K = K^* = K_c$ :

$$\exp\left(\frac{-2J}{k_B T_c}\right) = \tanh\left(\frac{J}{k_B T_c}\right)$$

Dessa maneira, fica assegurada a mesma temperatura crítica para as duas séries. Isso é a dualidade: a simetria em relação a  $T_c$  entre duas temperaturas diferentes (de séries diferentes). Resolvendo essa relação, escrita com  $K_c = J\beta_c$ , chegamos em:

$$e^{2K_c} = \frac{\cosh K_c}{\sinh K_c} = \frac{e^{2K_c} + 1}{e^{2K_c} - 1} = 1 + \frac{2}{e^{2K_c} - 1}$$

Então:

$$(e^{2K_c} - 1)^2 = 2$$

De onde chegamos na temperatura crítica dada por:

$$K_c = \frac{1}{2} \log(1 + \sqrt{2})$$

Esse é o resultado fornecido pela solução exata de Onsager, que no seu artigo "A Two-Dimensional Model with an Order-Disorder Transition", Physical Review, 1944; chega na mesma relação:  $\sinh(2K_c) \sinh(2K_c^*) = 1$  como condição para temperatura crítica. Já com modelos de campo médio:  $J\beta_c^{CM} = 1/4$ , um resultado consideravelmente diferente.

## 4

Partimos da hipótese de escala em que, sendo a energia livre de Gibbs uma função que pode ser dividida entre a soma de uma função bem comportada com uma função singular:  $g(t, \pi) = g_0(t, \pi) + g_s(t, \pi)$ .

Então, a parte singular da energia livre obedece a relação de escala:

$$g_s(t, \pi) = \lambda g_s(\lambda^a t, \lambda^b \pi)$$

como  $\lambda$  é arbitrário, podemos escolhê-lo tal que:  $\lambda^a t = 1$ , então  $\lambda = t^{-1/a}$ , e

$$g_s(t, \pi) = t^{-1/a} g_s(1, t^{-b/a} \pi) = t^{-1/a} F\left(\frac{\pi}{t^{b/a}}\right)$$

e definindo  $a$  tal que  $-1/a = 2 - \alpha$  e chamando  $\Delta = b(\alpha - 2)$ :

$$g_s(t, \pi) = t^{2-\alpha} F\left(\frac{\pi}{t^\Delta}\right)$$

A energia livre por partícula na representação de Gibbs é:

$$dg = -sdT + vdp$$

Assim, iremos calcular os expoentes críticos das grandezas termodinâmicas seguindo o caminho  $\pi = 0$  no diagrama de fases  $p - T$ .

i)

Lembrando que  $t = \frac{T - T_c}{T_c}$ . Nessas condições do diagrama de fases, a entropia é calculada como:

$$\begin{aligned} s(t, \pi = 0) &= -\left(\frac{\partial g_s}{\partial T}\right)_{p=p_c} = -\frac{1}{T_c} \left(\frac{\partial g_s}{\partial t}\right)_{\pi=0} = \frac{-1}{T_c} \left[(2 - \alpha)t^{1-\alpha} F\left(\frac{\pi}{t^\Delta}\right) - t^{2-\alpha} \Delta t^{-\Delta-1} \pi F'\left(\frac{\pi}{t^\Delta}\right)\right] \Big|_{\pi=0} \\ &= \frac{-1}{T_c} [(2 - \alpha)t^{1-\alpha} F(0)] \end{aligned}$$

Então, o expoente crítico da entropia, associado a quando nos aproximamos de  $t = 0$  por  $\pi = 0$ , é  $1 - \alpha$ :

$$s(t, \pi = 0) \propto t^{1-\alpha}$$

ii)

A compressibilidade isotérmica mede a variação relativa do volume com a pressão à temperatura fixa:

$$\kappa_T = \frac{-1}{v} \left( \frac{\partial v}{\partial p} \right)_T$$

Então, primeiro temos que obter o volume a partir da energia livre:

$$v = \left( \frac{\partial g}{\partial p} \right)_T = \frac{1}{p_c} \left( \frac{\partial g}{\partial \pi} \right)_t = \frac{1}{p_c} t^{2-\alpha-\Delta} F' \left( \frac{\pi}{t^\Delta} \right)$$

e então:

$$\begin{aligned} \kappa_T &= \frac{-1}{v} \left( \frac{\partial v}{\partial p} \right)_T = \frac{-1}{v p_c} \left( \frac{\partial v}{\partial \pi} \right)_t = \frac{-t^{-(2-\alpha-\Delta)}}{F' \left( \frac{\pi}{t^\Delta} \right)} \frac{1}{p_c} t^{2-\alpha-2\Delta} F'' \left( \frac{\pi}{t^\Delta} \right) \\ &= \frac{-t^{-\Delta}}{p_c} \frac{F'' \left( \frac{\pi}{t^\Delta} \right)}{F' \left( \frac{\pi}{t^\Delta} \right)} \end{aligned}$$

assim, quando nos aproximamos do ponto crítico por  $\pi = 0$ , o expoente crítico da compressibilidade é  $-\Delta$ :

$$\kappa_T(t, 0) \propto t^{-\Delta}$$

iii)

O calor específico a volume constante é dado por:

$$c_v = T \left( \frac{\partial s}{\partial T} \right)_v = T \left[ \left( \frac{\partial s}{\partial T} \right)_p \left( \frac{\partial v}{\partial p} \right)_T - \left( \frac{\partial s}{\partial p} \right)_T \left( \frac{\partial v}{\partial T} \right)_p \right] \left( \frac{\partial p}{\partial v} \right)_T$$

relação essa em que foram usadas as transformações jacobianas. Resolvendo essa equação fazendo todas as derivadas, com  $\pi = 0$ , e usando a relação  $\left( \frac{\partial s}{\partial p} \right)_T = - \left( \frac{\partial v}{\partial T} \right)_p = -v\alpha_p$ , usando os resultados obtidos nos outros itens temos:

$$c_v = c_p + \frac{T_c(t+1)v\alpha_p^2}{\kappa_T}$$

Chegamos então que a dependência de  $c_v$  com  $t$  quando  $\pi = 0$  e  $t \rightarrow 0$  o termo que domina faz com que o expoente crítico dominante seja:

$$c_v \propto t^{-\alpha}$$

iv)

O calor específico a pressão constante é calculado por:

$$\begin{aligned} c_p &= T \left( \frac{\partial s}{\partial T} \right)_p = t \left( \frac{\partial s}{\partial t} \right)_p + \left( \frac{\partial s}{\partial t} \right)_p \\ &= (t+1) \left( \frac{\partial}{\partial t} \frac{-1}{T_c} \left[ (2-\alpha)t^{1-\alpha} F \left( \frac{\pi}{t^\Delta} \right) - t^{1-\alpha-\Delta} \Delta \pi F' \left( \frac{\pi}{t^\Delta} \right) \right] \right)_p \end{aligned}$$

Então, usando a função da entropia calculada em i:

$$\begin{aligned} c_p(t, \pi = 0) &= \frac{-(t+1)}{T_c} \left[ (2-\alpha)(1-\alpha)t^{-\alpha} F \left( \frac{\pi}{t^\Delta} \right) + (2-\alpha)t^{-\alpha-\Delta} \pi F' \left( \frac{\pi}{t^\Delta} \right) + \mathcal{O}(\pi) + \mathcal{O}^2(\pi) \right] \\ &= \frac{-(t+1)}{T_c} \left[ (2-\alpha)(1-\alpha)t^{-\alpha} F(0) \right] \end{aligned}$$

$$c_p(t, \pi = 0) \propto t^{1-\alpha} + t^{-\alpha}$$

temos que o expoente crítico, quando nos aproximamos do ponto crítico  $t \rightarrow 0$  é dado pelo maior termo, que corresponde a descartar o termo de ordem maior, de forma que o expoente crítico é  $(-\alpha)$ :

$$c_v(t \rightarrow 0, \pi = 0) \propto t^{-\alpha}$$

v)

O coeficiente de dilatação térmica mede a dilatação relativa de um sistema a pressão constante e é dado por:

$$\alpha_p = \frac{1}{v} \left( \frac{\partial v}{\partial T} \right)_p = \frac{1}{v T_c} \left( \frac{\partial v}{\partial t} \right)_p$$

$$= \frac{p_c}{T_c} t^{-(2-\alpha-\Delta)} \frac{1}{F' \left( \frac{\pi}{t^\Delta} \right)} \left[ \frac{1}{p_c} (2-\alpha-\Delta) t^{1-\alpha-\Delta} F' \left( \frac{\pi}{t^\Delta} \right) + \frac{1}{p_c} t^{2-\alpha-2\Delta} \pi F'' \left( \frac{\pi}{t^\Delta} \right) \right]$$

e temos que:

$$\alpha_p(t, \pi = 0) = \frac{(2-\alpha-\Delta)t^{-1}}{T_c}$$

então quanto  $t \rightarrow 0$ , definimos como o expoente crítico da dilatação térmica (-1):

$$\alpha_p(t, \pi = 0) \propto t^{-1}$$

## 5

Primeiro, iremos converter as temperaturas da tabela para Kelvin, a primeira linha da tabela fica:

T	621.93K	623.81K	625.68K	631.33K
---	---------	---------	---------	---------

em que o valor em vermelho marca a temperatura maior do que a temperatura crítica  $T_c = 627.4K$  fornecida pela questão. Com essa mudança, e utilizando a correção para o campo magnético de que cada ponto:  $H_i^{corrigido} = H_i - 39.3m_i$

T = 621.93K		T = 623.81K		T = 625.68K		T = 631.33K	
m	H	m	H	m	H	m	H
11,36	-16	10,45	19	8,11	111	-	-
13,79	348	-	-	-	-	-	-
14,31	793	12,65	858	10,63	937	4,24	1.189
15,65	2.615	14,32	2.667	12,75	2.729	7,57	2.933
16,94	5.349	15,78	5.395	14,52	5.444	10,31	5.610
18,21	9.354	17,27	9.391	16,20	9.433	12,78	9.568
19,21	13.455	18,35	13.489	17,40	13.526	14,39	13.645
19,98	16.990	19,14	17.023	18,29	17.056	15,50	17.166

Figura 1: Tabela de dados. Para melhor visualização, as casas decimais de  $H$  foram suprimidas. Entretanto, elas foram consideradas no plot, apesar de não serem relevantes para a escala do plot logXlog.

Vamos reescrever também os valores de  $T$  na variável  $|t| = \frac{|T-T_c|}{T_c}$ :

t	8.71852.10 <sup>-3</sup>	5.72202741.10 <sup>-3</sup>	2.74147274.10 <sup>-3</sup>	6.26394644.10 <sup>-3</sup>
---	--------------------------	-----------------------------	-----------------------------	-----------------------------

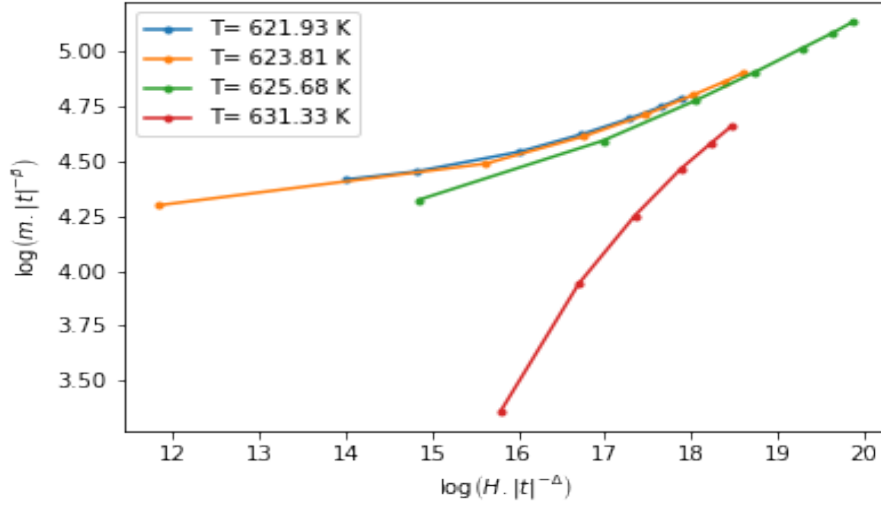
Os dados correspondentes a coluna em vermelho devem ter um comportamento diferente dos dados das outras colunas, pois estão acima da temperatura crítica. Dada a forma de escala e os valores dos expoentes críticos mencionados na apresentação da questão, os dados devem ser aproximados por curvas do tipo:

$$\frac{m}{|t|^\beta} = Y_\pm \left( \frac{H}{|t|^\Delta} \right)$$

Com os expoentes tendo os valores:  $\gamma = 1.34$ ,  $\beta = 0.378$ ,  $\delta = 4.58$  e  $\Delta = \beta(\delta - 1)$ . Esses valores estão de acordo com a relação de escala:  $\gamma = \beta(\gamma - 1)$ , pois:  $\beta(\delta - 1) = 0.378(3.58) = 1.353 \approx$



$1.34 = \gamma$ . Então, a partir dos dados da Figura 1: Tabela de dados, plotamos em um gráfico os valores de  $\log m|t|^{-\beta}$  por  $\log H|t|^{-\Delta}$ :

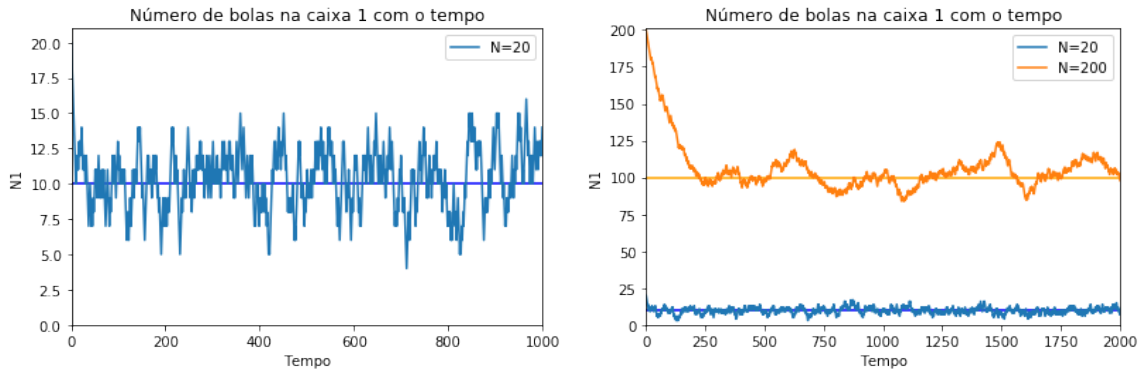


Como esperado, os dados abaixo da temperatura crítica tem o mesmo comportamento e colapsam aproximadamente em uma mesma curva. Já os dados acima da temperatura crítica (linha vermelha) tem um comportamento bastante distinto. Dessa forma, os dados se comportam conforme a relação de escala prevista.

## 6 Urna de Ehrenfest

a)

Usando Python, foi feito um programa para fazer as simulações (disponíveis no Apêndice A) com o qual foram geradas as duas figuras abaixo. Para ambos os casos *i*)  $N = 20$  e *ii*)  $N = 200$ , vemos que conforme o aumento de  $t$ , o sistema tende a flutuar ao redor da posição de equilíbrio  $N/2$ . Entretanto, as flutuações são em média da ordem de 10 bolas, então para o caso *i* (Figura 1), como há poucas bolas (somente  $N = 20$ ), as flutuações são da ordem do número de bolas. Assim, conforme  $N$  aumenta, as flutuações passam a ser cada vez mais desprezíveis e, na Figura 2, em que estão simulados os dois casos *i* e *ii*, fica explícito o comportamento, principalmente no caso *ii*, que em  $t \rightarrow \infty$  o sistema tende para  $N/2$ .



b)

Supondo que  $P(N_1, t + \Delta t)$  é a probabilidade de encontrar  $N_1$  bolas na urna 1 no instante de tempo  $t + \Delta t$ , podemos pensar que a evolução temporal desse sistema a cada passo de tempo  $\Delta t$  deve ser dada por dois termos:

i) um deles representando a probabilidade de no tempo  $t$  o sistema ter  $N_1 - 1$  bolas na urna 1 e ganhar uma bola da urna 2.

ii) O outro representando a probabilidade de no tempo  $t$  o sistema ter  $N_1 + 1$  bolas na urna 1 e perder uma bola da urna 1 para a urna 2.

Essas são as duas únicas possibilidades de se chegar no estado de  $N_1$  bolas na urna 1 em somente um passo de tempo  $\Delta t$ .

No primeiro caso, a probabilidade da urna 1 ter  $N_1 - 1$  bolas no tempo  $t$  é  $P(N_1 - 1, t)$ . A probabilidade de estando nesse estado ganhar uma bola da urna 2 é igual a probabilidade de se sortear um número que pertença a urna 2, como nesse estado a urna 2 tem  $N - (N_1 - 1)$  bolas, essa probabilidade é  $W_1 = \frac{N - (N_1 - 1)}{N}$ . Logo, o primeiro termo, associado a ter  $N_1 - 1$  bolas na urna 1 no tempo  $t$  E ganhar uma bola da urna 2 é  $P(N_1 - 1, t)W_1$ .

No segunda caso, a probabilidade da urna 1 ter  $N_1 + 1$  bolas no tempo  $t$  é  $P(N_1 + 1, t)$ . A probabilidade de estando nesse estado perder uma bola da urna 1 é igual a probabilidade de sortear uma bola que pertença a urna 1, como a urna 1 nesse estado tem  $N_1 + 1$  bolas, essa probabilidade é  $W_2 = \frac{N_1 + 1}{N}$ . Logo, o segundo termo, associado a ter  $N_1 + 1$  bolas na urna 1 no tempo  $t$  E perder uma bola da urna 1 é  $P(N_1 + 1, t)W_2$ .

Logo, a equação que representa essa evolução temporal na urna 1 é a soma desses dois termos:

$$P(N_1, t + \Delta t) = P(N_1 - 1, t)W_1 + P(N_1 + 1, t)W_2$$

com  $W_1 = \frac{N - (N_1 - 1)}{N}$  e  $W_2 = \frac{N_1 + 1}{N}$ .

A hipótese envolvida nessa formulação é de que a cada tempo necessariamente uma bola muda de caixa e, para decidir qual será essa bola, supõe-se que cada uma das  $N$  bolas tenha um número de 1 a  $N$  associado a ela (sem repetições), e que a cada intervalo de tempo  $\Delta t$  sorteia-se aleatoriamente um número de 1 a  $N$  que designará a bola que irá mudar de caixa.

c)

No estado de equilíbrio, a equação do modelo da urna corresponde a uma situação em que as probabilidade não dependem mais do tempo, então reescreveremos a equação de evolução temporal como:

$$\pi_{j+1} = \pi_j \frac{N - j}{N} + \pi_{j+2} \frac{j + 2}{N}$$

Ou seja, com  $p_j = \frac{N - j}{N}$  e  $q_{j+2} = \frac{j + 2}{N}$ :

$$\begin{aligned} \pi_0 &= 0 + \pi_1 q_1 \\ \pi_1 &= \pi_0 p_0 + \pi_2 q_2 \\ \pi_2 &= \pi_1 p_1 + \pi_3 q_3 \\ \pi_3 &= \pi_2 p_2 + \pi_4 q_4 \\ &\dots \\ \pi_j &= \pi_{j-1} p_{j-1} + \pi_{j+1} q_{j+1} \\ &\dots \\ \pi_N &= \pi_{N-1} p_{N-1} \end{aligned}$$

Resolvendo esse sistema por indução, usando a relação  $p_i + q_i = 1$ , e  $p_0 = 1$ , pois  $q_0 = 0$  chegamos em:

$$\begin{aligned} \pi_1 &= \frac{\pi_0}{q_1} \\ \pi_2 &= \frac{1}{q_2} (\pi_1 - \pi_0 p_0) = \frac{1}{q_2} \left( \frac{\pi_0}{q_1} - \pi_0 p_0 \right) = \frac{1}{q_2} \left( \frac{1}{q_1} - \frac{p_0 q_1}{q_1} \right) \pi_0 = \frac{1 - p_0(1 - p_1)}{q_2 q_1} \pi_0 = \frac{p_0 p_1 + 1 - p_0}{q_2 q_1} \pi_0 \\ &= \frac{p_0 p_1}{q_2 q_1} \pi_0 \\ \pi_3 &= \frac{1}{q_3} (\pi_2 - \pi_1 p_1) = \frac{1}{q_3} \left( \frac{p_0 p_1}{q_2 q_1} \pi_0 - \pi_1 p_1 \right) = \frac{1}{q_3} \left( \frac{p_0 p_1}{q_2 q_1} \pi_0 - \frac{\pi_0}{q_1} p_1 \frac{q_2}{q_2} \right) = \frac{1}{q_3} \left( \frac{p_0 p_1}{q_2 q_1} - p_1 \frac{(1 - p_2)}{q_2 q_1} \right) \pi_0 \\ &= \frac{(p_0 p_1 - p_1 + p_1 p_2)}{q_3 q_2 q_1} \pi_0 = \frac{p_1 (p_0 + p_2 - 1)}{q_3 q_2 q_1} \pi_0 = \frac{p_1 p_2}{q_3 q_2 q_1} \pi_0 \\ &= \frac{p_0 p_1 p_2}{q_3 q_2 q_1} \pi_0 \end{aligned}$$

dessa forma, por indução, chegamos que:

$$\pi_j = \frac{p_{j-1}p_{j-2}\dots p_0}{q_j q_{j-1}\dots q_1} \pi_0$$

que podemos escrever como:

$$\pi_j = \frac{[N - (j - 1)][N - (j - 2)]\dots[N]}{j(j - 1)\dots 1} \pi_0 = \frac{N!}{j!(N - j)!} \pi_0 = \binom{N}{j} \pi_0$$

que é a distribuição binomial. Para encontrar  $\pi_0$ , usamos a condição  $\pi_0 + \pi_1 + \dots + \pi_N = 1$ , então:

$$\sum_{j=0}^N \pi_j = 1 \quad \text{então} \quad \pi_0(1 + \dots) = \pi_0(2^N) = 1 \rightarrow \pi_0 = \frac{1}{2^N}$$

Assim, mostramos que a distribuição binomial:

$$P(N_1) = \frac{1}{2^N} \binom{N}{N_1}$$

é solução da equação de evolução temporal na situação de equilíbrio.

O valor médio pode ser calculado, fazendo a mudança de variável  $N'_1 = N_1 - 1$ , como:

$$\begin{aligned} \langle N_1 \rangle &= \sum_{N_1=0}^N N_1 P_N(N_1) = \sum_{N_1=0}^N N_1 \frac{N!}{N_1!(N - N_1)!} \frac{1}{2^N} = \sum_{N_1=1}^N \frac{N!}{(N_1 - 1)!(N - N_1)!} \frac{1}{2^N} \\ &= \sum_{N_1=1}^N \frac{N}{2} \frac{(N - 1)!}{(N_1 - 1)!(N - N_1)!} \frac{1}{2^{N-1}} = \frac{N}{2} \sum_{N_1=1}^N \frac{(N - 1)!}{(N_1 - 1)!(N - 1 - (N_1 - 1))!} \frac{1}{2^{N-1}} \\ &= \frac{N}{2} \sum_{N'_1=0}^{N-1} \frac{(N - 1)!}{(N'_1)!(N - 1 - N'_1)!} \frac{1}{2^{N-1}} = \frac{N}{2} \sum_{N'_1=0}^{N-1} P_{N-1}(N'_1) = \frac{N}{2} \end{aligned}$$

Como previsto na simulação, o valor esperado desse sistema é  $N/2$ . O cálculo de  $\langle (N_1 - \langle N_1 \rangle) \rangle$  é feito usando esse resultado:

$$\begin{aligned} \langle (N_1 - \langle N_1 \rangle) \rangle &= \sum_{N_1=0}^N (N_1 - N/2) P_N(N_1) = \sum_{N_1=0}^N N_1 P_N(N_1) - \frac{N}{2} \sum_{N_1=0}^N P_N(N_1) \\ &= \frac{N}{2} - \frac{N}{2} = 0 \end{aligned}$$

Faz sentido que o deslocamento médio em relação a média seja zero. Agora vamos calcular a variância  $\sigma^2$ , fazendo também uma mudança de variáveis  $N_1 - 2 = N'_1$ :

$$\begin{aligned} \sigma^2 &= \langle (N_1 - \langle N_1 \rangle)^2 \rangle = \langle N_1^2 + \langle N_1 \rangle^2 - 2N_1 \langle N_1 \rangle \rangle = \langle N_1^2 \rangle + \langle N_1 \rangle^2 - 2\langle N_1 \rangle^2 \\ &= \langle N_1^2 \rangle - \langle N_1 \rangle^2 = \sum_{N_1=0}^N N_1^2 P_N(N_1) - \frac{N^2}{4} \quad \text{então vamos resolver o primeiro termo:} \\ \langle N_1^2 \rangle &= \sum_{N_1=0}^N N_1^2 \frac{N!}{N_1!(N - N_1)!} \frac{1}{2^N} = \sum_{N_1=0}^N (N_1^2 - N_1) \frac{N!}{N_1!(N - N_1)!} \frac{1}{2^N} + \sum_{N_1=0}^N N_1 \frac{N!}{N_1!(N - N_1)!} \frac{1}{2^N} \\ &= \sum_{N_1=0}^N (N_1^2 - N_1) \frac{N!}{N_1!(N - N_1)!} \frac{1}{2^N} + \frac{N}{2} = \sum_{N_1=1}^N (N_1 - 1) \frac{N!}{(N_1 - 1)!(N - N_1)!} \frac{1}{2^N} + \frac{N}{2} \\ &= \sum_{N_1=2}^N \frac{N!}{(N_1 - 2)!(N - N_1)!} \frac{1}{2^N} + \frac{N}{2} = \sum_{N'_1=0}^{N-2} \frac{N(N - 1)(N - 2)!}{(N'_1)!(N - 2 - N'_1)!} \frac{1}{2^N} + \frac{N}{2} \\ &= \frac{N(N - 1)}{4} \sum_{N'_1=0}^{N-2} \frac{(N - 2)!}{(N'_1)!(N - 2 - N'_1)!} \frac{1}{2^{N-2}} + \frac{N}{2} = \frac{N(N - 1)}{4} \sum_{N'_1=0}^{N-2} P_{N-2}(N'_1) + \frac{N}{2} = \frac{N(N - 1)}{4} + \frac{N}{2} \end{aligned}$$

então:

$$\sigma^2 = \frac{N(N - 1)}{4} + \frac{N}{2} - \frac{N^2}{4} = \frac{-N}{4} + \frac{N}{2} = \frac{N}{4}$$

Então o coeficiente de variação, é:

$$\frac{\sqrt{\sigma^2}}{\langle N_1 \rangle} = \frac{\sqrt{N}/2}{N/2} = \frac{1}{\sqrt{N}}$$

d)

Utilizando a equação estocástica deduzida anteriormente, podemos escrever uma expressão para a evolução temporal do valor esperado  $\langle N_1 \rangle_t$ :

$$\langle N_1 \rangle_{t+\Delta t} = \sum_{N_1=0}^N N_1 P_{(N_1, t+\Delta t)} = \sum_{N_1=0}^N N_1 \left[ P_{(N_1-1, t)} \frac{N - (N_1 - 1)}{N} + P_{(N_1+1, t)} \frac{N_1 + 1}{N} \right]$$

Vamos analisar o primeiro termo da somatória:

$$S_1 = \sum_{N_1=0}^N N_1 \left[ P_{(N_1-1, t)} \frac{N - (N_1 - 1)}{N} \right]$$

quando  $N_1 = 0$ , devemos ter  $P_{(N_1-1, t)} = 0$ , pois não pode haver probabilidade de ter um número negativo de bolas. Então podemos reescrever a somatória começando em  $N_1 = 1$  e fazer uma mudança de variáveis  $N'_1 = N_1 - 1$ :

$$\begin{aligned} S_1 &= \sum_{N_1=1}^N N_1 \left[ P_{(N_1-1, t)} \frac{N - (N_1 - 1)}{N} \right] = \sum_{N'_1=0}^{N-1} (N'_1 + 1) \left[ P_{(N'_1, t)} \frac{N - N'_1}{N} \right] \\ &= \sum_{N'_1=0}^N (N'_1 + 1) \left[ P_{(N'_1, t)} \frac{N - N'_1}{N} \right] = \sum_{N'_1=0}^N P_{(N'_1, t)} \frac{NN'_1 - N'^2_1 + N - N'_1}{N} \end{aligned}$$

sendo que na segunda linha incluímos o termo  $N'_1 = N$  na somatória, pois ele é nulo. Já para o termo  $S_2$  temos:

$$S_2 = \sum_{N_1=0}^N N_1 \left[ P_{(N_1+1, t)} \frac{N_1 + 1}{N} \right]$$

Quando  $N_1 = N$ , o termo  $P_{(N_1+1, t)}$  deve ser nulo, pois não pode haver probabilidade de existir mais do que  $N$  bolas. Usando isso e fazendo uma mudança de variável:  $N_1 + 1 = N''_1$ , podemos escrever:

$$\begin{aligned} S_2 &= \sum_{N_1=0}^{N-1} N_1 \left[ P_{(N_1+1, t)} \frac{N_1 + 1}{N} \right] = \sum_{N''_1=1}^N (N''_1 - 1) \left[ P_{(N''_1, t)} \frac{N''_1}{N} \right] \\ &= \sum_{N''_1=0}^N N''_1 \left[ P_{(N''_1, t)} \frac{N''_1 - 1}{N} \right] = \sum_{N''_1=0}^N P_{(N''_1, t)} \frac{N''^2_1 - N''_1}{N} \end{aligned}$$

em que na última linha adicionamos o termo  $N''_1 = 0$ , pois ele é nulo. Juntando agora as duas somatórias, como as duas somam sobre termos iguais, podemos reescrevê-las com uma mesma variável, por exemplo  $N''_1 = N'_1 = N_1$  e obtemos:

$$\begin{aligned} \langle N_1 \rangle_{t+\Delta t} &= \sum_{N'_1=0}^N P_{(N'_1, t)} \frac{NN'_1 - N'^2_1 + N - N'_1}{N} + \sum_{N''_1=0}^N P_{(N''_1, t)} \frac{N''^2_1 - N''_1}{N} \\ &= \sum_{N_1=0}^N P_{(N_1, t)} \frac{NN_1 - N_1^2 + N - N_1}{N} + \sum_{N_1=0}^N P_{(N_1, t)} \frac{N_1^2 - N_1}{N} \\ &= \sum_{N_1=0}^N P_{(N_1, t)} \frac{NN_1 + N - 2N_1}{N} = \sum_{N_1=0}^N P_{(N_1, t)} \left( N_1 + 1 - \frac{2}{N} N_1 \right) \\ &= \langle N_1 \rangle_t + 1 - \frac{2}{N} \langle N_1 \rangle_t \end{aligned}$$

Então temos que:

$$\langle N_1 \rangle_{t+\Delta t} - \langle N_1 \rangle_t = 1 - \frac{2}{N} \langle N_1 \rangle_t$$

e para  $\Delta t \rightarrow 0$ , podemos pensar na forma diferencial dessa equação como:

$$\frac{\partial}{\partial t} \langle N_1(t) \rangle = 1 - \frac{2}{N} \langle N_1(t) \rangle$$

que é uma equação diferencial que tem como solução geral:

$$\langle N_1(t) \rangle = A e^{\frac{-2t}{N}} + \frac{N}{2}$$

Usando então como condição inicial que em  $t = 0$ :  $N_1(0) = N$ , igual ao que fizemos na simulação, conseguimos encontrar o parâmetro  $a$ :

$$N_1(0) = a + N/2 = N \rightarrow a = N/2$$

Logo chegamos na solução:

$$\langle N_1(t) \rangle = \frac{N}{2} e^{\frac{-2t}{N}} + \frac{N}{2}$$

que coincide com o que encontramos na simulação, que para  $t$  grande, quando  $t \rightarrow \infty$ , temos:

$$\langle N_1(t) \rangle \rightarrow \frac{N}{2}$$

Assim, para  $t$  grande a solução tende para o valor médio  $N/2$ , tal como vimos na simulação.

## 7 Método de Monte Carlo - Ising 2D

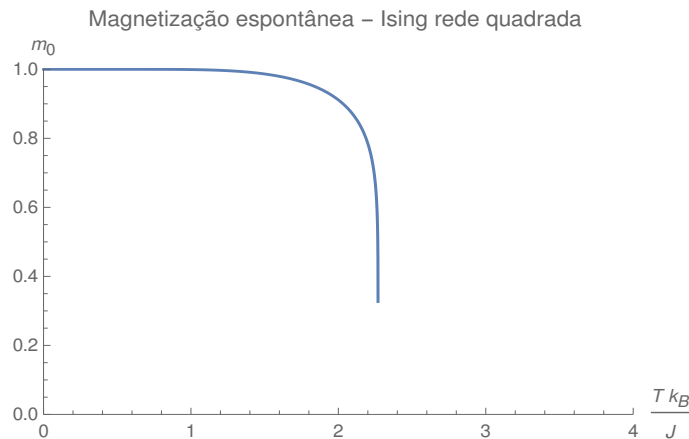
Na Mecânica Estatística nós estamos interessados em calcular as médias canônicas de uma grandeza  $A$ , tal que  $\langle A \rangle$  é definida através da soma de todas as configurações microscópicas **possíveis** de um sistema com  $N$  sítios. Em sistemas grandes, a medida que  $N$  aumenta, essa soma se torna numericamente impraticável. A ideia do método de Monte Carlo consiste em obter uma estimativa para a média das grandezas  $\langle A \rangle$  através de uma média aritmética simples em que somamos através de poucas configurações, porém selecionadas de forma a serem significativas, pois são representativas do equilíbrio.

Aqui iremos simular uma rede quadrada de lado de tamanho  $N$  com condições periódicas de contorno para obter uma estimativa do comportamento da magnetização espontânea. Pelo resultado obtido por Onsager, a magnetização espontânea é dada pela seguinte expressão:

$$m_0 = \left[ 1 - \left( \sinh \frac{2}{T} \right)^{-4} \right]^{\frac{1}{8}}$$

em que essa temperatura é definida como  $T = \frac{1}{\beta J}$ . A temperatura crítica é definida pelo ponto em que  $m_0 = 0$ , ou seja:  $\sinh \frac{2}{T} = 1$ , que define o valor:

$$T_c = 2.269185 \frac{J}{k_B}$$



O algoritmo que iremos simular é o **Algoritmo de Metrópolis**, e o código feito para simulá-lo e gerar as figuras foi feito em Python 3 e está disponível no Apêndice B. Começamos com a condição inicial corresponde a todos os spins para baixo ( $\sigma = -1$ ). Então percorremos todos os sítios da rede, passando por todos da primeira linha, depois todos da segunda linha, e assim por diante. Em cada sítio, dado pelas coordenadas  $(i, j)$ , calculamos a diferença de energia entre sua orientação inicial  $\sigma_{i,j}^i$  e sua orientação invertida  $\sigma_{i,j}^f = -\sigma_{i,j}^i$ :

$$\Delta E = -J(\sigma_{i,j}^f - \sigma_{i,j}^i)(\sigma_{i-1,j} + \sigma_{i+1,j} + \sigma_{i,j-1} + \sigma_{i,j+1})$$

essa diferença de energia corresponde a soma das diferenças de energia da mudança de orientação do sítio  $(i, j)$  com cada um dos seus 4 vizinhos mais próximos, com quem tem ligação.

Se  $\Delta E < 0$ , isso significa que essa mudança diminui a energia do sistema, ou seja, é "energeticamente favorável", e portanto é mais provável de ocorrer, levando o sistema para o estado de equilíbrio; é por isso que nesse caso aceitamos a mudança sempre, o sítio  $(i, j)$  passa a ter orientação  $\sigma_{i,j}^f$ , e seguimos adiante para analisar outro sítio.

Se  $\Delta E \geq 0$ , isso significa que essa mudança aumenta ou não muda a energia do sistema, ou seja, é "energeticamente desfavorável", e portanto mais improvável de ocorrer espontaneamente em um sistema. Nesse caso, nós calculamos  $r = \exp(-\frac{\Delta E}{T})$  e geramos um número uniformemente aleatório no intervalo  $[0, 1[$ , que chamamos de  $z$ . Se  $r > z$  aceitamos a mudança e o sítio  $(i, j)$  passa a ter orientação  $\sigma_{i,j}^f$ , ou seja, quanto maior  $r$  maior a chance de ele ser aceito. Isso significa que quanto menor for a relação  $\frac{\Delta E}{T}$ , maior é a chance de se aceitar a mudança. Ou seja, se  $T$  é grande temos bastante chance de aceitá-la, e podemos entender isso pensando que se a temperatura na rede é grande, há mais energia no sistema, aumentam as flutuações e a probabilidade de haver uma mudança de orientação "custosa" em um sítio é maior. Seguindo o mesmo raciocínio, se  $\Delta T$  é positivo, porém pequeno, as chances de você aceitar a mudança também aumentam, pois apesar de essa ser uma mudança de energia "custosa", o custo dela não é tão alto. Obviamente então, se  $r < z$  não aceitamos a mudança, e o sítio  $(i, j)$  continua sendo  $\sigma_{i,j}^i$ . Isso porque esse é o caso inverso do que acabamos de analisar:  $\Delta E$  é grande (mudança de energia custosa) e/ou a temperatura  $T$  é pequena, havendo pouca energia no sistema.

Fazemos essa análise de  $\Delta E$  passando por todos os  $N \times N$  sítios do sistema fazendo, e fazemos isso  $K$  vezes no sistema.  $K$  é o número de passos/iterações de Monte Carlo, e nossa hipótese é que depois de  $K > 100$  o sistema deve convergir para um estado de equilíbrio, ou seja,  $K > 100$  é suficiente para sair da região transiente associada à condição inicial. Nós fizemos no total  $K = 300$  iterações de Monte Carlo, ou seja, geramos 300 configurações diferentes, cada uma delas sendo identificada como  $k$ , com  $k = 0, 1, 2, \dots, 299$ .

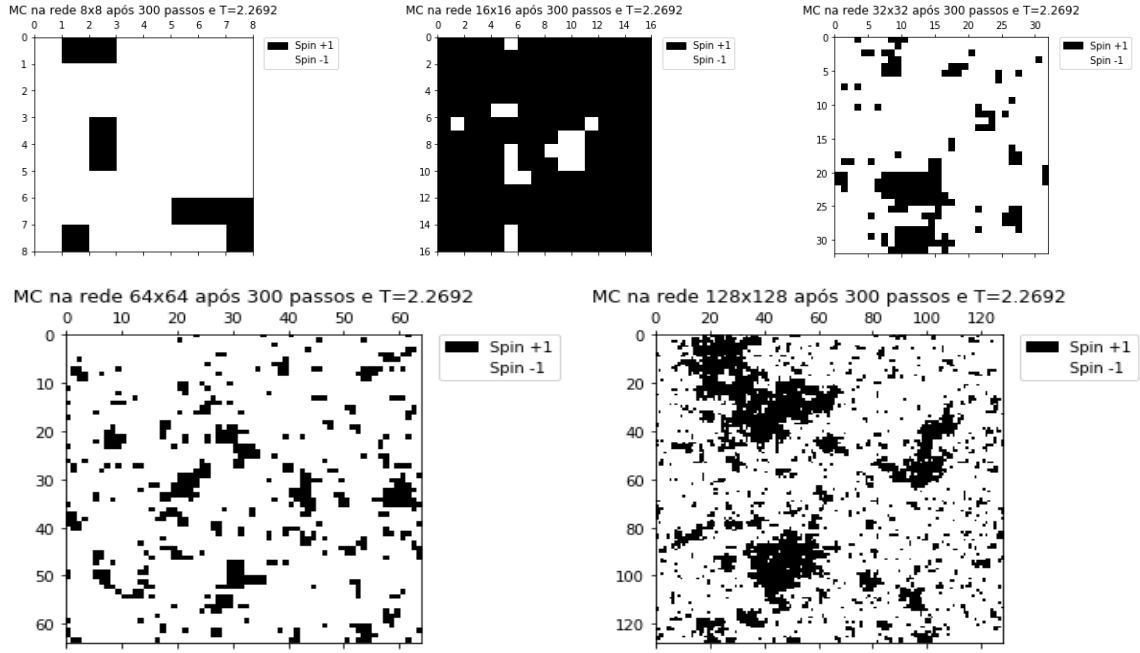


Figura 2: Visualização de redes de diferentes tamanhos  $N$ , todas elas foram simuladas para mesma temperatura e usando o algoritmo descrito acima, com mesma condição inicial. Todas elas correspondem a configuração  $k = 299$ .

A variável que queremos estudar é a magnetização, ou seja, queremos calcular a magnetização média para cada rede de tamanho  $N$ . Para isso iremos considerar somente as configurações entre  $100 \leq k \leq 299$ , que designamos as configurações representativas da rede, pois consideramos que representam o estado de equilíbrio.

O módulo da magnetização de uma configuração, para cada uma das  $k$  configurações "representativas" de uma rede de tamanho  $N$ , com  $k = 100, \dots, 299$ , é calculado como:

$$|m_k(T)| = \left| \sum_i^N \sum_j^N \sigma_{(i,j)} \right|$$

Com esses valores, fazemos a média da magnetização para a rede de tamanho  $N$ , ou seja:

$$m_N(T) = \sum_{k=100}^{299} |m_k(T)|$$

Nas figuras abaixo (Figura 3) vemos que conforme  $N$  aumenta as flutuações estatísticas diminuem. Além disso, plotamos os gráficos para três temperaturas, uma delas muito abaixo da  $T_c$ :  $T = 1.2$  (cor azul), outra muito acima:  $T = 4$  (cor verde) e uma ao redor de  $T_c$ :  $T = 2.2692$  (laranja). Com isso, vemos também que para valores ao redor da temperatura crítica, as flutuações estatísticas também são maiores.

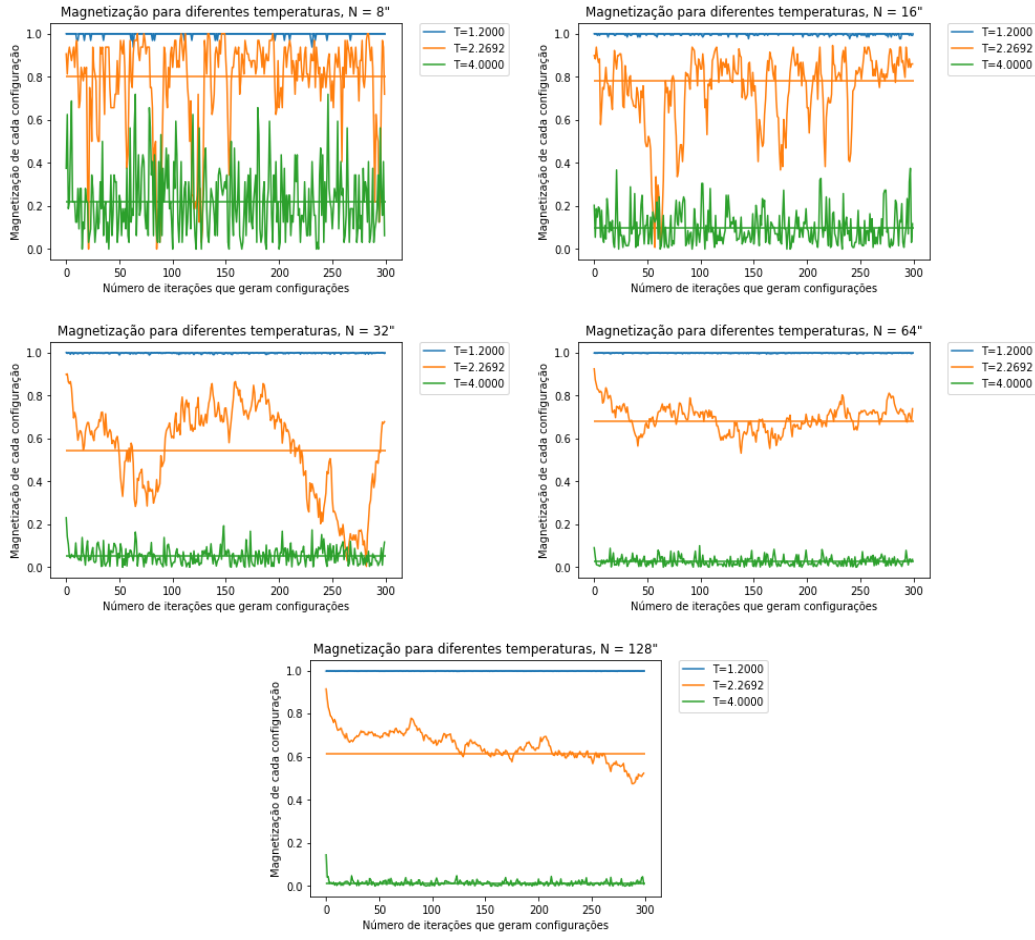
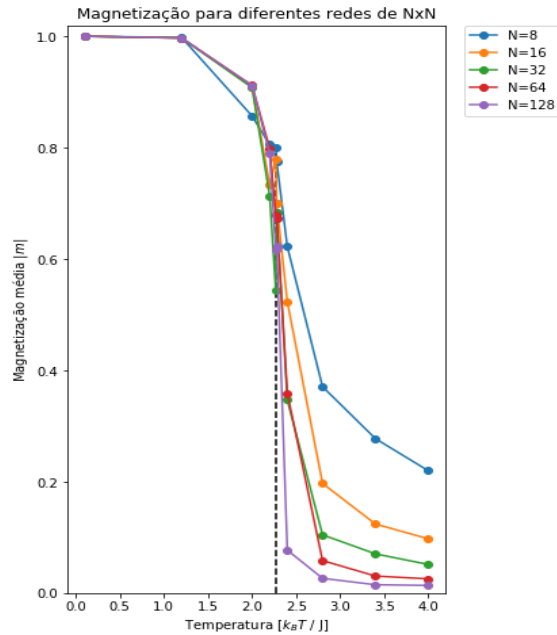


Figura 3: Magnetização de cada configuração. O eixo x é a variável  $k$ , e o y a variável  $|m_k(T)|$ . As retas horizontais são as médias para cada  $N$ ,  $m_N(T)$ , definida acima como das configurações  $100 \leq k \leq 299$ .

Então, escolhemos alguns valores de temperatura e calculamos a média  $m_N(T)$  para cada uma das redes  $N$ , obtendo a relação esperada:





A linha preta tracejada representa a temperatura crítica  $T_c$ . Como esperado, quanto maior a rede, melhor é a aproximação de Monte Carlo, obtendo um comportamento mais próximo da relação teórica obtida por Onsager. Além disso, apesar de ser difícil de visualizar devido a alta densidade de pontos e linhas, ao redor do ponto  $T_c$ , próximo de  $m_N(T) \sim 0.7$ , vemos que há uma flutuação maior dos pontos, que oscilam muito devido à proximidade do ponto crítico.

Para visualizar esse resultado na rede, geramos a imagem da configuração  $k = 299$  da rede  $N = 128$  para algumas temperaturas. Para uma temperatura muito baixa, os spins devem estar alinhados, não havendo ( $T = 0.1$ ) ou havendo pouquíssimos ( $T = 1.2$ ) spins com orientação diferente. Nesse caso, com alta ordenação da rede, obtemos magnetização média  $m_N(T) \sim 1$ , como esperado.

Para temperaturas muito altas ( $T = 4$ ), os spins estão completamente desordenados, e obtemos magnetização média  $m_N(T) \sim 0$ .

Para temperaturas ao redor da temperatura crítica, há magnetização diferente de zero, porém de uma configuração para outra há grande flutuação.

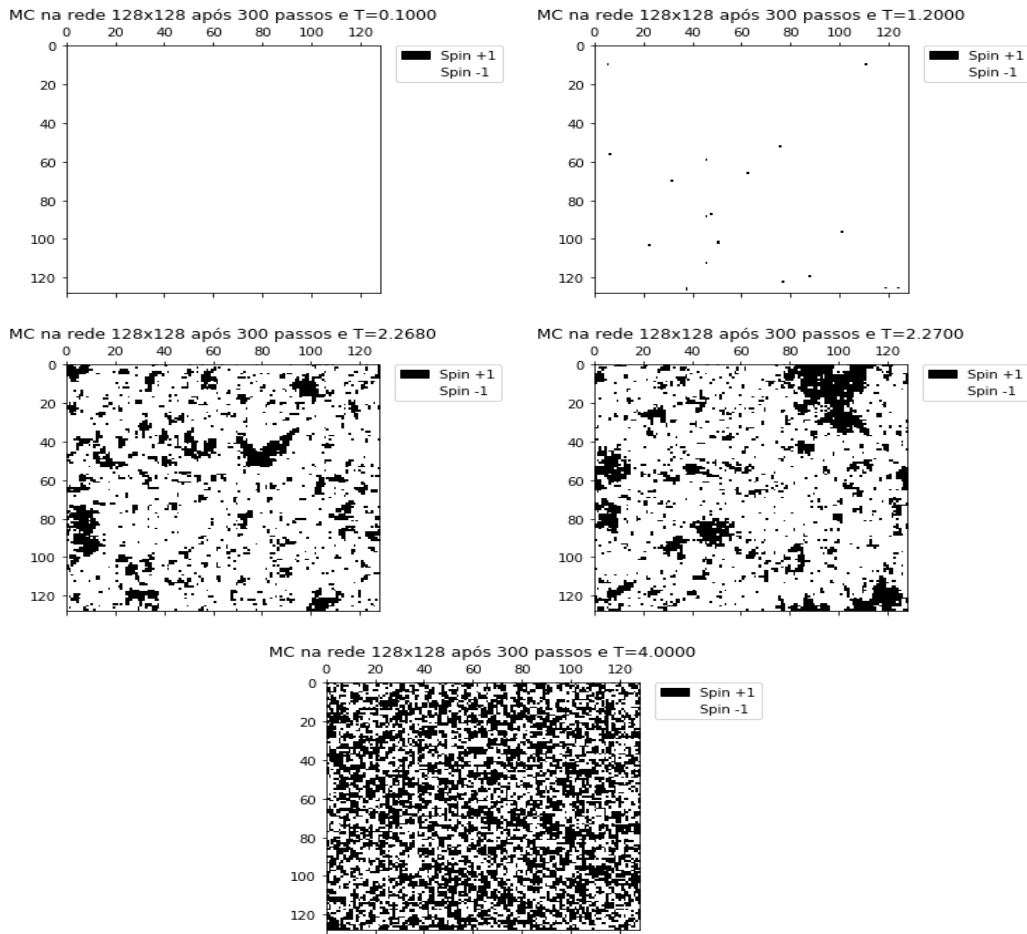


Figura 4: Rede  $128 \times 128$ ,  $k = 299$  para diferentes temperaturas, mostrando visualmente o comportamento esperado.

## APÊNDICE A

```
from matplotlib import pyplot as plt
import random as rand

#Simulação
def urna_T(N,T):
    N1 = N
    N2 = N - N1
    n1 = [N1]
    n2 = [N2]
    for t in range(T):
        r = rand.uniform(0,1)
        if (r <= N1/N):
            N1 = N1 - 1
            N2 = N2 + 1
        else:
            N1 = N1 + 1
            N2 = N2 - 1

        n1.append(N1)
        n2.append(N2)

    return n1,n2

#Parâmetros
N = 200 #200
T = 2000

#Figura
n1_20,n2 = urna_T(20,T)
n1_200,n2 = urna_T(200,T)
t = [i for i in range(T+1)]

plt.plot(t,n1_20,'-',label="N=20")
plt.plot(t,n1_200,'-',label="N=200")
plt.hlines(100,0,T,'orange')
plt.hlines(10,0,T,'blue')
plt.xlabel('Tempo')
plt.ylabel('N1')
plt.ylim(0,N+1)
plt.xlim(0,T+1)
plt.legend()
plt.title('Número de bolas na caixa 1 com o tempo')
```

## APÊNDICE B

```
import math
import numpy as np
import matplotlib.pyplot as plt
import matplotlib.patches as mpatches

FUNÇÕES QUE REALIZAM SIMULAÇÃO:

#Função que analisa deltaE e devolve o sigma novo para cada sítio da rede

def deltaE_analysis(i:int,j:int,Matrix:tuple,T:float):

    sigma_i = Matrix[i][j]
    sigma_f = Matrix[i][j]*(-1) #Muda o valor do spin

#Condições de contorno:
if (i == N-1 and j == N-1):
    deltaE = - (sigma_f - sigma_i)*(Matrix[i-1][j] + Matrix[0][j] + Matrix[i][j-1] + Matrix[i][0])
    #na unidade de J
elif(i == N-1):
    deltaE = - (sigma_f - sigma_i)*(Matrix[i-1][j] + Matrix[0][j] + Matrix[i][j-1] + Matrix[i][j+1])
    #na unidade de J
elif(j == N-1):
    deltaE = - (sigma_f - sigma_i)*(Matrix[i-1][j] + Matrix[i+1][j] + Matrix[i][j-1] + Matrix[i][0])
    #na unidade de J
else:
    deltaE = - (sigma_f - sigma_i)*(Matrix[i-1][j] + Matrix[i+1][j] + Matrix[i][j-1] + Matrix[i][j+1])
    #na unidade de J

#Análise do deltaE
if (deltaE<0):
    return sigma_f
else:
    r = math.exp(-deltaE/T)
    z = np.random.uniform(0,1)
    if (r>z):
        return sigma_f
    else:
        return sigma_i

#Simulação de Monte Carlo, passa por todos os sítios da rede,
#e faz isso com um número de iterações de
#Monte carlo dado por "n_configs", que será o número de configurações simuladas.
#A simulação é feita para uma temperatura e um tamanho de
#rede específicos.
#Devolve em Matrix a última configuração da rede (para visualização apenas)
#Devolve em mag o valor absoluto da magnetização média de
#cada configuração das "n_configs" configurações

def montecarlo(N,orientation,n_configs,T):
    mag = []
    Matrix = [[orientation for i in range(N)] for j in range(N)] #Matrix[i][j] (i=linha, j=coluna)
    for p in range(n_configs):
        for i in range(N):
            for j in range(N):
                Matrix[i][j] = deltaE_analysis(i,j,Matrix,T)

    mag.append(np.absolute(np.sum(Matrix))/(N*N))
```

```

    return Matrix,mag

    #Simulação para diferentes temperaturas e UM tamanho de rede:
    #Devolve em Matrix a última configuração da rede (para visualização apenas) de cada temperatura
    #Devolve em mag o valor absoluto da magnetização média de cada configuração das "n_configs" config

def montecarlo_variasT(N,orientation,n_configs,T):
    Matrix = []
    mag = []

    for i in range(len(T)):
        mat,b = montecarlo(N,orientation,n_configs,T[i])

        #é a Matriz da última configuração para cada T
        #para visualização
        Matrix.append(mat)

        #é o vetor com a magnetização para cada
        #uma das configurações em passos, para cada T
        mag.append(b)

    return Matrix, mag

    # Calcula e devolve a magnetização média para cada temperatura em um vetor de tamanho T
    #Essa média é calculada com todas as configurações a partir de init iterações
    #(descartamos as primeiras init configurações)

def mag_media(mag,n_configs,T,init):
    return [sum(mag[i][init:]/(n_configs-init) for i in range (len(T)))]

TIPOS DE PLOTS PARA VIZUALIZAÇÃO:

    #Plot para ver a última configuração simulada por T e por N: você vê a rede quadrada

    #Chamar da seguinte maneira: fig_rede(Matrix_N[i][j],N_plot[i],n_configs,T[j]),

    #i = índice que define o N; j = índice que define a T

def fig_rede(Matrix,N,n_configs,T):
    #Mostra a matriz como uma figura
    plt.matshow(Matrix,cmap='Greys',origin='upper',extent=(0, N, N, 0))

    #Legenda para as marcações branca e preta na rede
    patches = [mpatches.Patch(color='black',label="Spin +1"),
                mpatches.Patch(color='white',label="Spin -1")]
    #Se todos os spins estiverem alinhados igualmente,
    # o mapa será branco independente de ser -1 ou 1
    #devido ao modo como a função colormap distribue cores.

    plt.legend(handles=patches,bbox_to_anchor=(1.05, 1), loc='upper left', borderaxespad=0.)

    plt.title('MC na rede %1.fx%1.f após %1.f passos e T=%1.4f' %(N,N,n_configs,T),y=1.08)

    plt.savefig('Simulação T=%1.3f e rede N=%1.f .png' %(T,N),bbox_inches="tight" )

    plt.show()

    return

```

```

#Plot para ver um gráfico da magnetização para
#todas as configurações, Y:mag/config X:n_configs
#Chamar da seguinte maneira:
#fig_mag_por_config(mag_N[i],n_configs,mag_med_N[i],T,posicoes,N_plot[i])
#o índice i chama o N certo
#Ela define vetores T_plot, mag_plot, mag_med_plot que terão somente as T,mag,mag_med que vc
#quer que sejam plotada, definidas pelas posições
#Pois se plotar todas as T fica muito poluído

def fig_mag_por_config(mag,n_configs,mag_med,T,posicoes,N):
    plt.figure(figsize=(6,4))

    x_plot = [i for i in range(n_configs)]

    T_plot = []
    mag_plot = []
    mag_med_plot = []

    for i in (posicoes):
        T_plot.append(T[i])
        mag_plot.append(mag[i])
        mag_med_plot.append(mag_med[i])

    color = ['C0','C1','C2','C3','C4','C5','C6']

    for i in range(len(posicoes)):
        plt.plot(x_plot,mag_plot[i],label="T=%1.4f"%T_plot[i])
        plt.hlines(mag_med_plot[i],0,n_configs,color[i])

    plt.legend(bbox_to_anchor=(1.05, 1), loc='upper left', borderaxespad=0.)
    plt.xlabel('Número de iterações que geram configurações')
    plt.ylabel('Magnetização de cada configuração')
    plt.title('Magnetização para diferentes temperaturas, N = %i"%N)

    plt.savefig('Simulação varias T e rede N=%1.f .png' %(N),bbox_inches="tight" )

    plt.show()

# Plot para ver magnetização média por temperatura para diferentes redes de tamanho N:
#Chamar da seguinte maneira: fig_mag_por_T(mag_med_N,T,N)
#Em que N é um vetor com os tamanhos de rede simulados, em ordem, também salvar nos vetores:
#mag_med_N a magnetização média para cada simulação de rede tamanho N

def fig_mag_por_T(mag_med_N,T,N):
    plt.figure(figsize=(5,9))

    for i in range(len(N)):
        plt.plot(T,mag_med_N[i],'-o',label="N=%i"%N[i])

    plt.vlines(2.269185,0,0.8,colors='k',linestyles='dashed')

    lgd = plt.legend(bbox_to_anchor=(1.05, 1), loc='upper left', borderaxespad=0.)
    plt.xlabel('Temperatura [ $k_B T$  / J]')
    plt.ylabel('Magnetização média  $|m|$ ')
    plt.ylim(0,1.02)
    plt.title('Magnetização para diferentes redes de  $N \times N$ ')

    plt.savefig('Simulação varias T e varios N .png',bbox_inches="tight")

```

## PARÂMETROS

```
#Parametros fixos sempre
up = 1 ; down = -1
orientation = down

#Parametros fixos para todas as simulações
np.random.seed(1)
T = [0.1,1.2,2,2.2,2.269185,2.28,2.4,2.8,3.4,4] #Temperatura crítica é Tc = 2.269185
n_configs = 300 #número de configurações geradas/iterações de Monte Carlo

#Quero as seguintes posições de T para plotar no gráfico:
posicoes = [T.index(1.2),T.index(2.269185),T.index(4)]

#Parâmetros que devem ser modificados por simulação:
#N = ; #Tamanho de uma lateral da rede quadrada

N_plot = [0 for i in range(5)]
mag_N = [0 for i in range(5)]
Matrix_N = [0 for i in range(5)]
mag_med_N = [0 for i in range(5)]

    EXEMPLO DE COMO CHAMAR FUNÇÕES:

#Para simular uma rede de tamanho N
N = 8
i = 0
N_plot[0] = N

Matrix_N[0], mag_N[0] = montecarlo_variasT(N,orientation,n_configs,T)

init = 100
#Número de configurações iniciais (antes do equilíbrio) que serão descartadas para a média

mag_med_N[0] = mag_media(mag_N[0],n_configs,T,init)

#Para vizualizar:
fig_mag_por_config(mag_N[0],n_configs,mag_med_N[0],T,posicoes,N_plot[i])

fig_rede(Matrix_N[0][4],N_plot[0],n_configs,T[4])

#Para gerar o plot para ver magnetização média por temperatura para diferentes
#redes de tamanho N, deve-se primeiro simular os 5 tamanhos de rede.
```