

イジング模型における相検出

レポート提出日

2022 年 11 月 29 日

レポート提出者

05502211 北西優弥

1 平均場近似と数値解法

1.1 2次元イジング模型

イジング模型は、磁性体の性質を表すモデルの一つで、相転移を記述することができる。イジング模型は図 1 で示すように規則的に配列した粒子がそれぞれ、上向きか下向きのいずれかのスピン状態をとると考える。

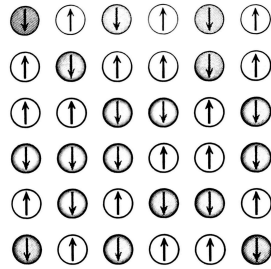


図 1: 2次元のイジング模型 [1]

隣接したスピン間にのみ相互作用が働いているとし、その相互作用エネルギーの大きさは等しくスピンが同じ向きするとき $-J$ 、逆向きするとき J とする。また、上向きスピンを $\sigma = +1$ 、下向きスピンを $\sigma = -1$ で表すとする。そうするとこの系のエネルギーは

$$H = -J \sum_{\langle i,j \rangle} \sigma_i \sigma_j \quad (1)$$

で表される。 (i, j) は格子上の点で、 $\langle i, j \rangle$ は最近接格子点を表す。 $i = \{0, \dots, n-1\}$, $j = \{0, \dots, m-1\}$ とすると、全スピンの配列が揃っているとき、 H は最小になり、スピンの配列が乱雑なほど H は大きくなる。

温度 T の熱平衡状態は、自由エネルギー

$$F = E - TS \quad (2)$$

を最小にする状態として決まるため、低温ではスピンの揃った秩序相が、高温ではスピンの向きが乱雑な無秩序相が実現する。このようにしてある温度で自発磁化をもつ秩序相から自発磁化を失う無秩序相へ転移が起こる物質のひとつが強磁性体である。

1.2 2次元イジング模型の解法

平均場近似によって得られた無撞着方程式の解析的近似解や数値計算による自由エネルギーの最適化などいくつかの方法で、磁化の温度依存性を導く。

イジング模型の分配関数は式 (1) のハミルトニアンを用いて

$$Z = \sum_{\{\sigma\}} e^{-\beta H} = \sum_{\{\sigma\}} \exp \left(\beta J \sum_{\langle i,j \rangle} \sigma_i \sigma_j \right) \quad (3)$$

である。これからヘルムホルツの自由エネルギー

$$F = -\frac{1}{\beta} \ln Z \quad (4)$$

が求まり、様々な物理量が得られる。しかし、式 (3) の式を見ると状態数は 2^{n+m} であり、格子が大きくなるとそのまま数値計算で解くのは難しくなる。2次元正方格子のイジング模型は外部磁場の無い場合には格子点数 $N \rightarrow \infty$ の極限で厳密に解けることが知られている。厳密解の解法としては転送行列法を用いたものがある。

2次元イジング模型の近似解法としては平均場近似法がある。平均場近似は、磁化で発生する磁場を外部磁場と同一視し、自己無撞着に解く近似法の一つである。これより Bragg-Williams 近似を用いて近似解を求める。

ミクロカノニカル集団の考え方を用いてエントロピーを求め、自由エネルギーと熱平衡状態の磁化を求める。全スピン数を N 、上向き・下向きスピンの数をそれぞれ、 N_+, N_- とする。このとき状態数は

$$W = \binom{N}{N_+} = \frac{N!}{N_+!(N - N_+)!} \quad (5)$$

である。したがってエントロピーは、Stirling の近似を用いると

$$S = k_B \ln W = -Nk_B \left\{ \frac{N_+}{N} \ln \left(\frac{N_+}{N} \right) + \left(1 - \frac{N_+}{N} \right) \ln \left(1 - \frac{N_+}{N} \right) \right\} \quad (6)$$

1 スピン当たりの磁化 $m \equiv (N_+ - N_-)/N$ を用いると

$$S = \frac{1}{2} Nk_B \{ 2 \ln 2 - (1 + m) \ln(1 + m) - (1 - m) \ln(1 - m) \} \quad (7)$$

となる。一方、エネルギーの期待値は各状態の実現確率と相互作用エネルギーを掛けて

$$E = \frac{1}{2} Nz \left\{ \left[\left(\frac{N_+}{N} \right)^2 + \left(\frac{N_-}{N} \right)^2 \right] + 2 \left(J \frac{N_+}{N} \cdot \frac{N_-}{N} \right) \right\} = -\frac{1}{2} Nz J m^2 \quad (8)$$

である。1つのスピンの隣接するスピン数を z とし、 $Nz/2$ はスピン総数である。これより自由エネルギー $F(m)$ は

$$F(m) = -\frac{1}{2} Nz J m^2 - \frac{1}{2} Nk_B T [2 \ln 2 - (1 + m) \ln(1 + m) - (1 - m) \ln(1 - m)] \quad (9)$$

m の熱平衡状態の磁化 m は $dF/dm = 0$ を満たす点であるから

$$\frac{dF}{dm} = 0 \implies -NzJm + \frac{1}{2} Nk_B T \ln \left(\frac{1+m}{1-m} \right) = 0 \quad (10)$$

$$\implies \frac{1}{2} \ln \left(\frac{1+m}{1-m} \right) = \frac{T_c}{T} m \quad \left(T_c \equiv \frac{zJ}{k_B} \right) \quad (11)$$

$$\implies m = \tanh \left(\frac{T_c}{T} m \right) \quad (12)$$

この自己無撞着な方程式を解くことで、各温度での平衡状態の m が得られる。しかし、この方程式はすべての温度領域を、解析的に解くことはできない。限られた温度領域で近似的に解くことはできる。これよりいくつかの方法でこの方程式の解を導く。

1.2.1 自己無撞着方程式の解法. 特定の温度領域での近似

式 (12) は $T \sim T_c$ では近似的に解が得られる. まず $m = 0$ は常に解を満たすことは明らかである. 実際には系の平衡点は極小であるから $d^2F/dm^2 < 0$ より

$$\frac{1}{(1-m)(1+m)} > \frac{zJ}{k_B} T = \frac{T_c}{T} \quad (13)$$

が得られる. したがって, $m = 0$ が解となるのは, $T_c/T < 1$ の範囲である. 次に低温極限を考える.

$\lim_{x \rightarrow \pm\infty} \tanh(x) = \pm 1$ であるから, $T \rightarrow 0$ の極限では, $m = \pm 1$ が解である. 次に $T \sim T_c$ 付近を考える. $T_c/T < 1$ で $m = 0$ であったからこの付近で $m \sim 0$ と考える. よって式 (12) をテイラー展開して

$$m = \tanh\left(\frac{T_c}{T}m\right) \simeq \frac{T_c}{T}m - \frac{1}{3}\left(\frac{T_c}{T}m\right)^3 \quad (14)$$

$T_c/T > 1$ の範囲を考えるので $m \neq 0$ であるから m で両辺を割り, m について解くと

$$m = \pm \sqrt{3\left(\frac{T}{T_c}\right)^2 \left(\frac{T_c - T}{T_c}\right)} \simeq \pm \sqrt{3\left(1 - \frac{T}{T_c}\right)} \quad (15)$$

が得られる.

まとめると, $T/T_c > 1$ では $m = 0$, $T \sim T_c$ では $m \sim \sqrt{3(1 - T/T_c)}$, $T \rightarrow 0$ で $m \rightarrow \pm 1$ となることが解析的な近似によって得られた.

1.2.2 自己無撞着方程式の解法. グラフによる解法

方程式 (12) を満たす解 m は, $m = (T/T_c)x$ と置いたとき, その直線と $m = \tanh(x)$ の交点である. 図 2 に直線 $m = (T/T_c)x$ と $m = \tanh(x)$ をプロットしたものを示す. グラフは python でプロットしたもので, ソースコードは ising_m-x.py である.

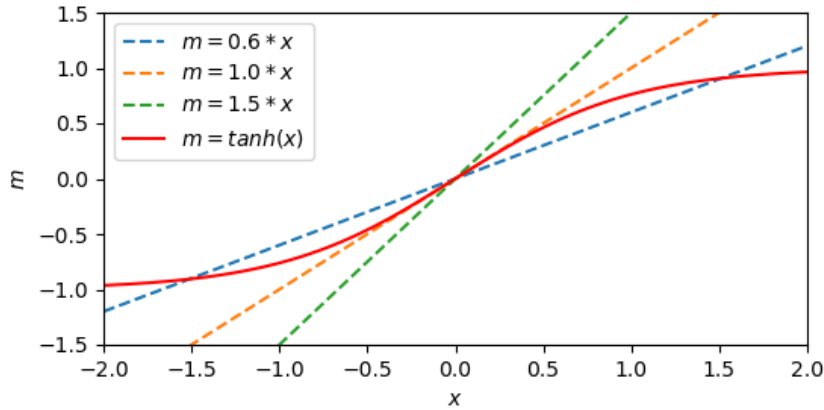


図 2: 自己無撞着方程式のグラフによる解法

それぞれ直線の傾きは $T/T_c = 0.6, 1.0, 1.5$ である. グラフから $T/T_c > 1$ のときには, 交点は $m = 0$ のときのみで, $T/T_c < 1$ では, 交点は $m = 0, \pm m_0$, $T/T_c \rightarrow 0$ では, 交点は $m \rightarrow \pm 1$ となることが分かる. これは先ほど解析的に解いた近似解の結果と一致する.

1.2.3 自己無撞着方程式の解法. 逆関数からの計算

式 (12) の自己無撞着方程式は $m = f\left(\frac{T}{T_c}\right)$ の形には解析的に求めることはできなかった. しかし, その上の式から $\frac{T}{T_c} = f(m)$ と逆関数の形として解析的に解くことができる.

$$\frac{T}{T_c} = 2m \left\{ \ln \left(\frac{1+m}{1-m} \right) \right\}^{-1} \quad (16)$$

磁化 m に対応する T/T_c が得られる。定義域は $m = 0$ を除く $-1 < m < 1$ である。これを用いてグラフにプロットした様子を図 3 に示す。グラフの作成には python を用い、ソースコードは `ising_m-t_inv.py` である。

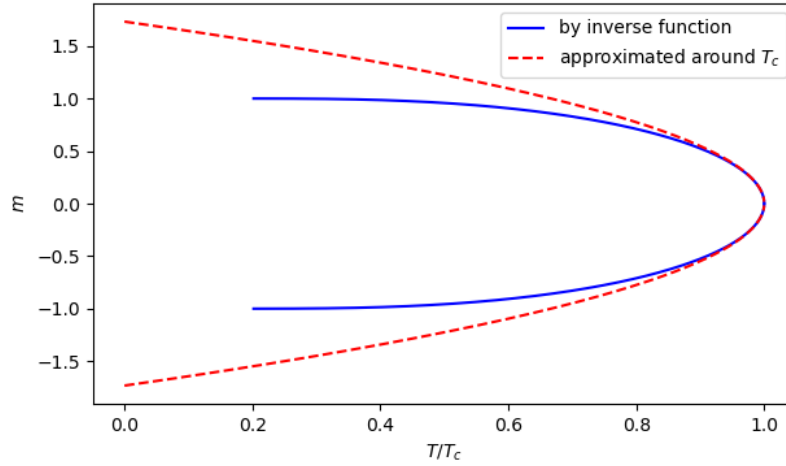


図 3: 磁化の温度依存性 (磁化 m の逆関数から計算)

赤の点線は上で導いた $T \sim T_c$ での解析解である。青の曲線が m の逆関数から計算して得られた解である。逆関数から得られた解は $T \sim T_c$ 付近では解析解の振舞いと一致する。また、 $T \rightarrow 0$ で $m \rightarrow \pm 1$ となっていることも確認できる。 T/T_c が小さい領域で曲線が途切れているが、これはプログラムで計算する際に、 \ln の定義域を超えないように $m = 1 - |\delta|$ までで打ち切ったことが原因である。図 3 は $\delta = 1e-4$ である。青の曲線は $T \sim 0$ で傾きがゼロにかなり近いため、 $T/T_c \sim 0$ 付近まで解を得るには、 δ の値をかなり小さくする必要がある。 $\delta = 1e-8$ でも $T/T_c > 0.1$ までしか解は得られなかった。

1.2.4 自己無撞着方程式の解法. 勾配降下法

以前ニューラルネットワークについて学習した際に用いられていた確率的勾配降下法を適用できるのではないかと思ったため、確かめてみた。

勾配降下法は関数の勾配をもとに極小値を見つけて最適化問題を解くアルゴリズムである。今回のイジング模型においては、自由エネルギー式 (9) のエネルギー最小化問題として考えられる。したがって、式 (9) の勾配がゼロになる磁化 m を見つける最適化問題である。各温度での自由エネルギー F の振舞いを図 4 に示す。グラフは python を用いて作成し、ソースコードは `ising_f-m.py` である。

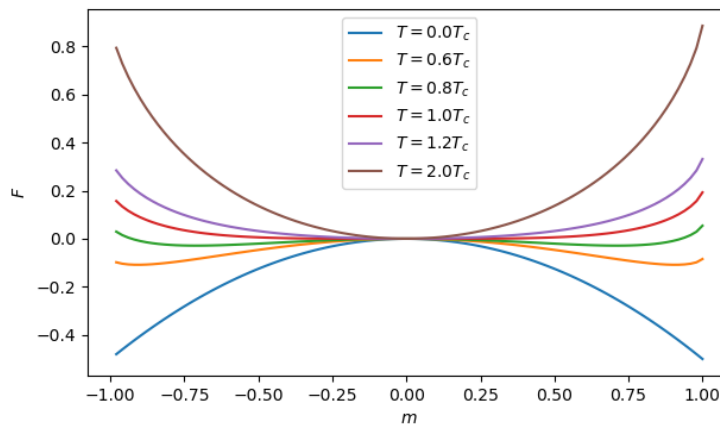


図 4: 各温度での自由エネルギーの磁化依存性. ($T_c = 1, N = 1$)

$m = 0$ で $f(m) = 0$ になるようにずらして描画している．各曲線の勾配がゼロになっている極小点が系の平衡時の磁化 m である． $T > T_c$ では極小点は $m = 0$ のみであり， $T < T_c$ では極小点は2点 $m = \pm m_0$ にあることが分かる．これらの極小点を勾配降下法によって求めることを考える．

時間ステップによって，自由エネルギーを極小点へと近づけていく．そのとき，系はエネルギー変化が負になるような方向へ作用が働く．したがってある定数 η を用いて更新は，

$$m \leftarrow m - \eta \frac{dF}{dm} \quad (17)$$

と表せられる．これは確率的勾配降下法 (stochastic gradient descent) の式と同じである． η は1ステップの更新の量を表す．実際にこのSGDによって，自由エネルギーが極小点へと更新される様子を図5に示す．更新時の勾配は解析的にも得られるが，今回は中心差分の数値微分によって行った．その勾配を計算するプログラムは mathutil.h 内の GradientDescent 名前空間内にある．複数のクラスのメンバ関数の計算に対応させるため，テンプレートを用いて実装した．ステップの更新を数値計算したプログラムは solve_selfconsistent.h 内の optimizeFreeEnergyBySGD() 関数である．

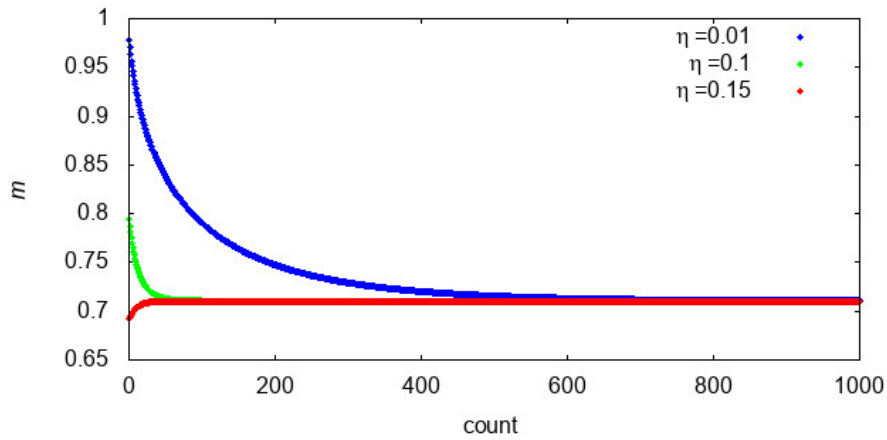


図 5: SGD による自由エネルギーの最適化 ($T = 0.8T_c$ ．初期値 $m = 1 - 1e-3$)

横軸は最適化を行ったステップ数であり，縦軸はそのときの磁化の値である． $T = 0.8T_c$ のときの様子で，初期値は $m = 1 - 1e-3$ から始めた．これは図4の緑の曲線で，右端から極小点を探していることに対応する．図5より更新を行うにしたがって，自由エネルギーが極小となるような磁化 m の値に収束している様子が確認できる．それぞれの曲線は学習率 η を変えており， η の値が大きいほど m は速く収束していくことが分かる．実際に式(17)によって自由エネルギーを最適化できることが確かめられた．

勾配降下法のアルゴリズムにはSGDの他にも様々な手法がある．Momentum と呼ばれる手法は次のように数式で表せられる．

$$\mathbf{v} \leftarrow \alpha \mathbf{v} - \eta \frac{\partial L}{\partial \mathbf{W}} \quad (18)$$

$$\mathbf{W} \leftarrow \mathbf{W} + \mathbf{v} \quad (19)$$

L は最適化を行う目的関数であるが，今回のイジング模型においては， L は自由エネルギー F であり， \mathbf{W} は磁化 m に相当する．SGD に比べて新しくパラメータ α が追加された． α は通常 1 より小さな値を指定する．SGD がそのまま勾配に比例した値で更新されたのに対し，この α によって同じ勾配でも減衰させていくことができる．

他に勾配降下法として以下の AdaGrad と呼ばれる手法がある．

$$\mathbf{h} \leftarrow \mathbf{h} + \frac{\partial L}{\partial \mathbf{W}} \odot \frac{\partial L}{\partial \mathbf{W}} \quad (20)$$

$$\mathbf{W} \leftarrow \mathbf{W} + \eta \frac{1}{\sqrt{\mathbf{h}}} \frac{\partial L}{\partial \mathbf{W}} \quad (21)$$

今のイジング模型の場合， L が自由エネルギー F ， \mathbf{W} は磁化 m に対応する． \odot はアダマール積で各要素ごとの積を表すが，今は F, m はスカラーであるから通常の積になる．これまでのSGDやMomentumと

比較すると、更新する際に $1/\sqrt{h}$ を乗算することで学習係数を減衰させていくことができる。しかし、無限に更新すると学習係数はゼロになり、更新されなくなってしまうという欠点がある。

これらのアルゴリズムはそれぞれに欠点があり、他にも様々な最適化手法が存在する。どの問題でも最適であるような手法はなく、問題によってアルゴリズムやパラメータを調整する必要がある。

実際にこれらのアルゴリズムを用いて、自由エネルギーの最適化を行い、磁化の温度依存性を数値計算した結果を図 6 と図 7 に示す。それぞれの最適化手法の実装は `mathutil.h` 内の `GradientDescent` スコープ内にある。磁化の温度依存性の数値計算のプログラムは `solve_selfconsistent.h` 内の `solveSelfConsistentBySGD` 関数内にある。

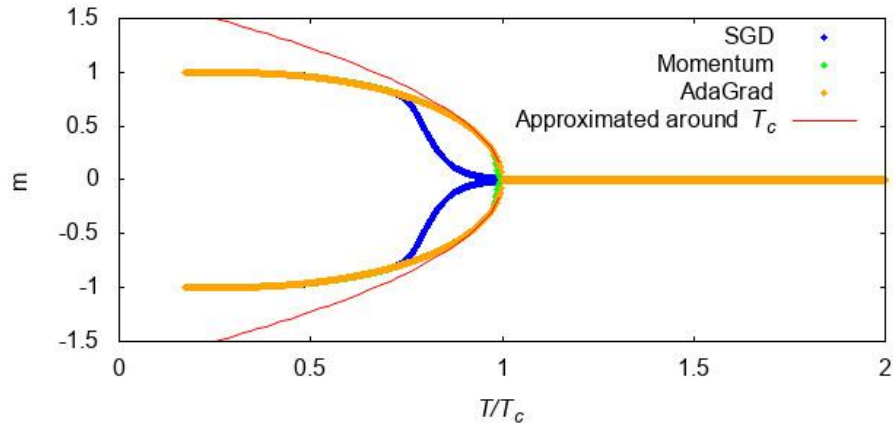


図 6: 勾配降下法から求めた磁化の温度依存性 (最大更新数 2000)

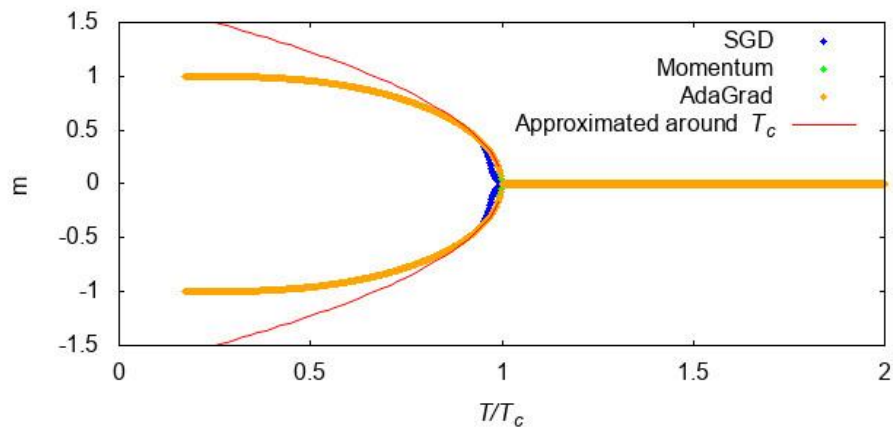


図 7: 勾配降下法から求めた磁化の温度依存性 (最大更新数 10000)

いずれの手法も $\eta = 0.01$, $\alpha = 0.9$ とした。自由エネルギーを最適化する際の磁化の初期値はいずれも $m = 1e-2$ である。図 6 が最大更新数が 2000 の様子で、図 7 が最大更新数が 10000 の様子である。 `mathutil.h` にあるように、更新幅 (勾配) が $1e-7$ 以下になれば、最適化を終えるようにしてある。赤は曲線が解析的に得られた $T \sim T_c$ 付近の振る舞いである。

最大更新数が 2000 の図 6 を見ると、Momentum, AdaGrad はこれまでに見てきたような磁化の振舞いを再現している。しかし、SGD で得られた磁化の振舞いはおおよそ $0.8 < T/T_c < 1$ 付近で大きく異なった。これは最大更新数 2000 ステップ以内に SGD が極小点を見つけることができなかったためである。SGD だけ T/T_c が 1 に近い範囲で最適化されなかった理由は、自由エネルギーの振舞い図 4 から分かるように、 $T/T_c \sim 1$ の振舞いは $m = 0$ 付近で勾配がほぼゼロの領域が広いためだと考えられる。SGD は式 (17) から、勾配が小さいところではほとんど更新されない。それに対し、Momentum や AdaGrad は勾配が小さくても、 v, h によってある程度更新される違いがある。図 7 の最大更新数 10000 のときには SGD でも極小点が得られていることが分かる。十分なステップ数があれば、勾配降下法によって、自由エネルギーの最

適化を行い、磁化の温度依存性の関係を得ることができた。いずれの図でも T/T_c が小さい領域では、磁化の値が得られていない。これは更新する際に、 m の値が、自由エネルギーの関数 (式 9) の定義域を超えるためである。数値計算で m が \ln の定義域を超えて $-\text{nan(ind)}$ が発生する。それを防ぐために、更新の際に $\min(1 - 1e7, m)$ のように磁化 m が 1 を超えないように工夫すれば、 T/T_c が小さい領域でも磁化の値が得られるだろう。

図 8 に $T/T_c = 0.8$ のときの各アルゴリズムが磁化を最適化していく様子を示す。プログラムは `solve_selfconsistent.h` 内の `compareOptimizer()` 関数にある。学習パラメータは $\eta = 0.01$, $\alpha = 0.9$ である。初期値はいずれも $1e-2$ から始めた。Momentum はおよそ 250 ステップには最適化されているのに対し、AdaGrad はおよそ 1500 ステップ、SGD は 3000 ステップほどで最適化されている。

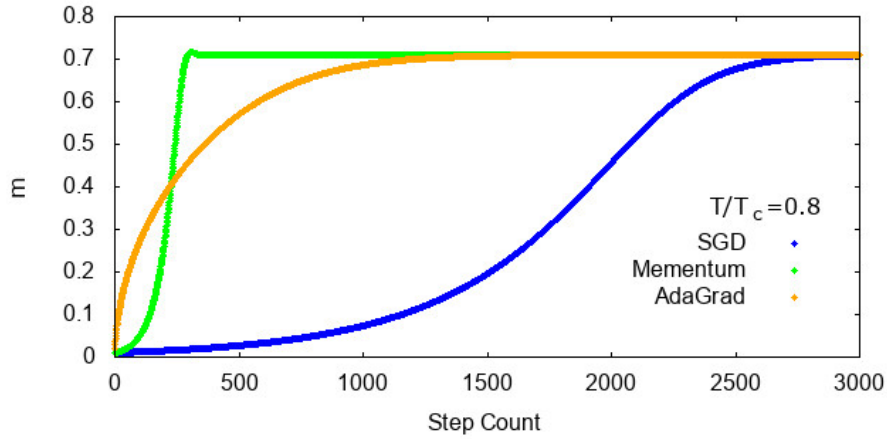


図 8: 磁化が最適化される様子のアルゴリズムの比較 ($T/T_c = 0.8$)

これまで自由エネルギーの式 (9) を目的関数として最適化してきたが、自己無撞着方程式 (12) を目的関数として考えることもできる。それは次の式がゼロになるような m を見つける最適化問題である。

$$L = \left(m - \tanh\left(\frac{T_c}{T}m\right) \right)^2 \quad (22)$$

これにより、磁化の温度依存性を数値計算した結果を図 9 に示す。プログラムのソースファイルは `solve_selfconsistent.h` 内の `solveSelfConsistentWithLoss()` 関数内にある。

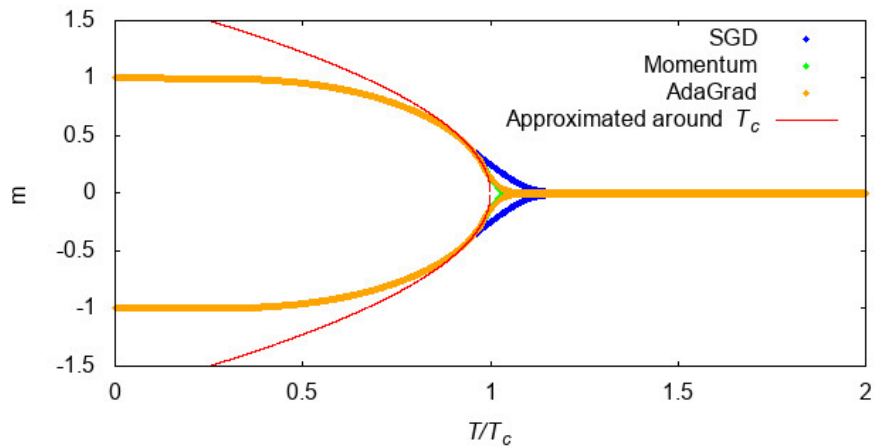


図 9: 自己無撞着方程式の二乗誤差を用いて勾配降下法により求めた磁化の温度依存性 (最大更新数 10000)

同じ最大更新数 10000 の先ほどの自由エネルギーを目的関数とする場合に比べ、 $T/T_c \sim 0$ 付近で解を得ることができたが、 T/T_c が 1 より少し大きい領域では更新数 10000 以内に極小点を見つけることができていない。図 9 は初期値を $m = 0.5 + 1e-2$ から始めたが、 $m = 1$ 付近から始めるともっと収束が悪くなっ

た。また、この損失関数で注意しなければならないのは、極小点が3点 $m = 0, \pm m_0$ あることで、初期値を $m = 0$ 付近から始めると、温度 T に依らず、ゼロに収束してしまうことである。

これまで見たように、勾配降下法によって磁化の温度依存性を数値計算で導くことができた。しかし、目的関数や更新のパラメータ値、初期値やアルゴリズムを適切に選ぶ必要がある。磁化の温度依存性を知るためだけなら、自己無撞着方程式の逆関数から数値計算した方が確実に計算量も大幅に抑えられる。

2 マルコフ連鎖モンテカルロ法によるスピン配位のシミュレーション

2.1 メトロポリス法

メトロポリス法はモンテカルロ法の一つで、特定の確率分布に収束させるサンプリング法である。マルコフ連鎖の遷移確率の選び方には自由度があるが、マルコフ連鎖が収束する十分条件が詳細釣り合いの原理である。ここでは、イジング模型の配位の古典統計力学的な確率分布を考え、メトロポリス法によって配位の遷移シミュレーションを試みる。

イジング模型のスピン配位 s_i が実現する確率は

$$P(s_i) = \frac{1}{Z} e^{-\beta H[s_i]} \quad (23)$$

で与えられる。ハミルトニアン H について、2つの状態 s_i, s_j について、 $H[s_i] < H[s_j]$ であるとする。この場合、低エネルギーである状態 s_i の方が高確率で実現されると考えられる。そこで遷移確率を

$$P(s_i | s_j) = 1 \quad (24)$$

とする。そうすると詳細釣り合いの原理

$$P(s_i | s_j) P(s_j) = P(s_j | s_i) P(s_i) \quad (25)$$

から、

$$P(s_j | s_i) = P(s_i | s_j) \frac{P(s_j)}{P(s_i)} = \exp(-\beta (H[s_j] - H[s_i])) \quad (26)$$

である。これより2状態の遷移確率をエネルギー変化によって表せられた。イジング模型の配位について、これらの遷移確率でスピンを変えていったとき、系はエネルギーの低い状態が実現されると考えられる。具体的には以下のようなステップで配位を更新していく。

1. ランダムに初期状態のスピン配位 s_0 を決める。
2. 今の配位 s_i のハミルトニアンを計算する。
3. ランダムに1サイトを選び、そのスピンを反転させる (s_j)。
4. 配位 s_j のハミルトニアンを計算する。
5. 配位の状態を以下のように更新する。
 - (a) $H[s_j] < H[s_i]$ の場合、配位を s_j に更新する。
 - (b) $H[s_j] \geq H[s_i]$ の場合、 $e^{-\beta(H[s_j]-H[s_i])}$ が乱数 $0 < r < 1$ より小さければ、配位を s_j に更新する。
 - (c) それ以外の場合、配位を変化させずに s_i のままとする。
6. ステップ2以降を繰り返す。

以上のメトロポリス法によって、イジング模型の配位をシミュレートした結果を図10から図12に示す。メトロポリス法のプログラムは mathutil.h 内の Metropolis クラスにある。MetropolisMethod クラスは汎用的にいくつかの関数を切り替えられるよう、テンプレートでメトロポリス法の骨組みだけを実装している。エネルギーの計算や更新は別のクラスで定義した関数を用いる。今回の例では、isingmodel.h にある

IsingModel クラスの関数を用いてメトロポリス法を計算する．シミュレーションの表示は GUI アプリをプログラムで作成したが，特定のフレームワークを含むため，それは省略する．

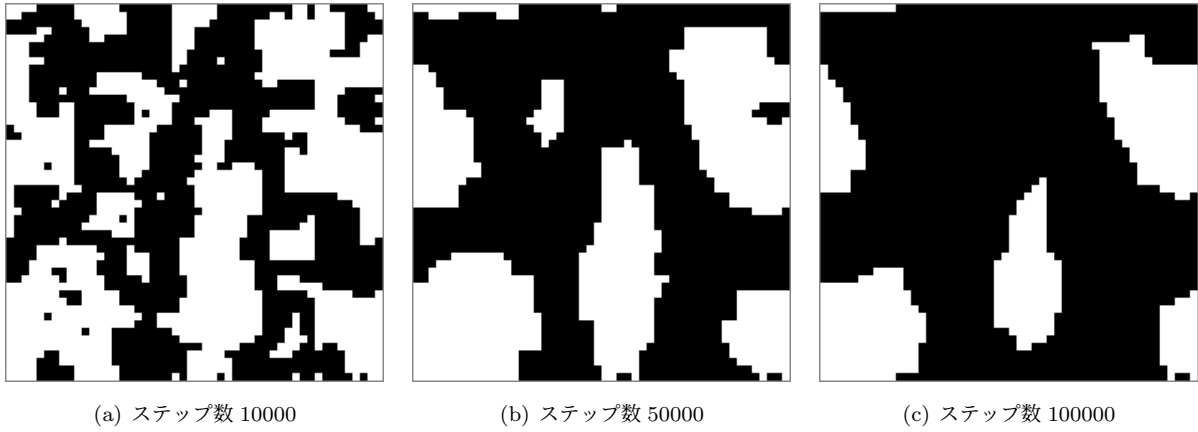


図 10: メトロポリス法による Ising 模型のシミュレーション ($k_B T = 0.1$, $J = 1$)



図 11: メトロポリス法による Ising 模型のシミュレーション ($k_B T = 4.0$, $J = 1$)

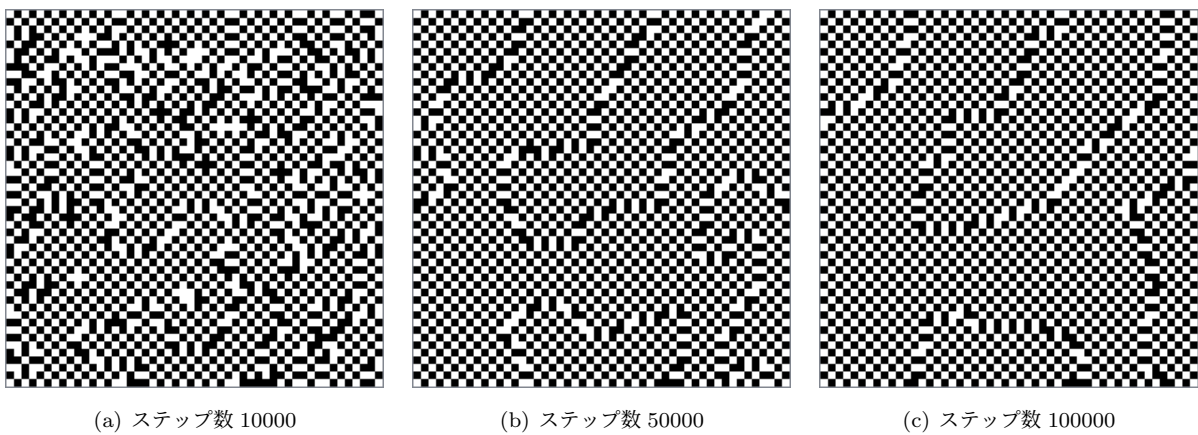


図 12: メトロポリス法による Ising 模型のシミュレーション ($k_B T = 0.1$, $J = -1$)

図 10 は $k_B T = 0.1$ の場合のスピン配位のシミュレーションである．格子数は 50×50 である．ステップ数が増えるにつれて，白と黒が分かれた島のようなものができていることが分かる．これは白と黒 (アップスピンとダウンスピン) が隣接するスピン対が徐々に減少しているということである．ここでは示していないが，ステップ数が十分上がると，全てのスピンのアップまたはダウンに揃うことも確認できた． $J = 1 > 0$ であるからスピンの向きが揃った強磁性を示すほうが安定するといった結果が再現されている．

図 11 は $k_B T = 4.0$ の場合のスピンのシミュレーションである。図 10 の $k_B T = 0.1$ の場合に比べてスピンは無秩序に配列している。1 スピンあたりの平均磁化 m はゼロに近い値だろうと予想できる。

図 12 は $k_B T = 0.1, J = -1$ の場合のスピンのシミュレーションである。縞々の模様の秩序が確認できる。これは $J < -1$ であるから、隣接するスピンが異なるほうが安定するという結果が再現されたと考えられる。

上の結果から、温度によってスピン配位の状態が変わることが確認できた。次に、各温度でスピン配位のシミュレーションをし、そのときの 1 スピンあたりの磁化 m を計算を行うことによって、磁化の温度依存性を数値計算してみる。途中で磁化の値が変わる転移温度があると予想できる。実際に計算した結果を図 13 に示す。プログラムは `isingspinconfig.h` にある。20x20 の格子を 100 万回メトロポリス法で更新し、それを 800 回各温度を計算するため、実行に時間がかかった。

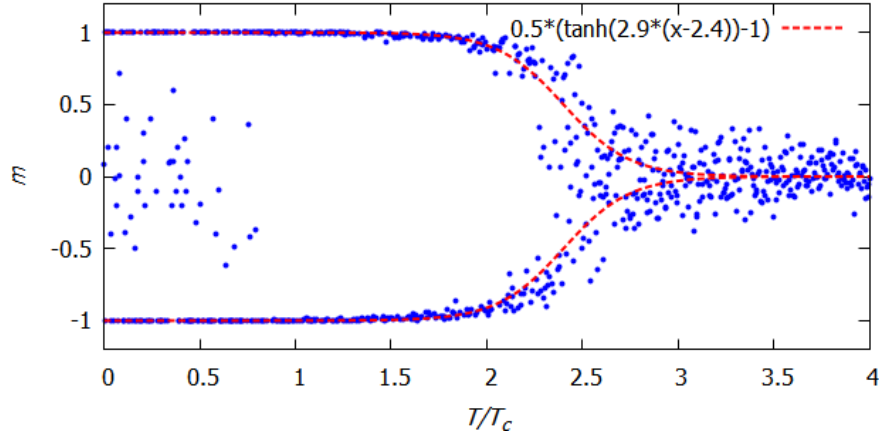


図 13: メトロポリス法によるスピン配位のシミュレーションから求めた磁化の温度依存性

スピン配位のサイズは 20x20 で、メトロポリス法の更新数は 10^6 回とした。図 13 をみるとおおよそ $T/T_c < 2$ の領域ではほとんどが $m = \pm 1$ であり、 $T/T_c > 2$ では徐々に $m = 0$ を中心に分布していく様子が確認できる。 $T/T_c \sim 0$ でも $m = 0$ 付近に分布する点もみられるが、これは 10^6 回の更新以内でスピンが最も安定した配位に実現されなかったものと思われる。更新数を十分大きくとれば $m = \pm 1$ のいずれかに収束すると考えられる。 $T/T_c > 2$ で $m = 0$ を中心とするばらつきがあるが、これは遷移の確率によるもので、格子サイズを無限に大きくとれば、ばらつきが無くなり $m = 0$ に収束すると思われる。

赤線は分布から予想して適当に描いた $m = \pm \frac{1}{2} \left\{ \tanh \left(2.9 \left(\frac{T}{T_c} - 2.4 \right) \right) - 1 \right\}$ 曲線である。磁化の分布とよく一致しているが、式とパラメータは適当に決めたものでこの式に意味があるのかは分からなかった。

それぞれメトロポリス法の更新時に 1 サイトの選び方はランダムであるが、格子上を順に走査するように更新した場合、各温度でスピン配位の収束が速くなることが確認できた。これはおそらくランダムに選ぶより順に選んだ方が偏りがなく効率が良いからだと考えられる。

サイズが無限の 2 次元 Ising 模型は Onsager によって解かれており、それによれば相転移温度 T^* は以下で与えられる。^[2]

$$T^* = \frac{2J}{k \ln(\sqrt{2} + 1)} \sim 2.269 \quad (27)$$

上でみたスピン配位のシミュレーションによる相転移温度はこれとおおよそ一致している。

2.2 熱浴法

熱浴法はメトロポリス法と同様にモンテカルロ法の一つで、特定の確率分布に収束させるサンプリング法である。ここでは、イジング模型のスピン配位の遷移確率を考え、熱浴法によって配位の遷移シミュレーションを行う。

メトロポリス法は系全体のエネルギーに着目して遷移確率が決定したことに対し、熱浴法では1つの自由度(1 スピン)に着目し、最近接部分を熱浴とみなして遷移確率を決定する。1スピンの存在確率を最近接部分の状態から求めて次の状態を決定する。

具体的には、状態 s_i から s_j の遷移確率を

$$P(s_j|s_i) = \frac{P(s_j)}{\sum_k P(s_k)} \quad (28)$$

のように $P(s_j)$ に比例するようにとる。これは詳細釣り合いの原理そのものであるので、マルコフ連鎖は収束する。実現確率を計算する際に、エネルギーを最近接部分を熱浴とみなすと、

$$P_i(s_i = +|*) = \frac{P(s_i = +)}{P(s_i = +) + P(s_i = -)} = \frac{\exp(-\beta R_i)}{\exp(-\beta R_i) + \exp(\beta R_i)} \quad (29)$$

$$P_i(s_i = -|*) = \frac{P(s_i = -)}{P(s_i = +) + P(s_i = -)} = \frac{\exp(\beta R_i)}{\exp(-\beta R_i) + \exp(\beta R_i)} \quad (30)$$

となる。ここで $*$ は任意の状態、 R_i はサイト i の局所エネルギー

$$R_i = -J \sum_{j \in \langle i, j \rangle} s_j \quad (31)$$

である。これをまとめると、熱浴法は次のようなステップでスピン配位を更新していく。

1. ランダムに初期状態のスピン配位 s_0 を決める。
2. 1 スピンを選択し、以下の遷移確率を計算する。

$$\omega^{(i)} = \frac{\exp(-\beta R_i)}{\exp(-\beta R_i) + \exp(\beta R_i)} \quad (32)$$

3. 一様乱数 $\xi \in [0, 1)$ を用いて次のようにスピンを更新する。

$$s_i^{\text{next}} = \begin{cases} 1 & (\xi < \omega^{(i)}) \\ -1 & \text{それ以外} \end{cases} \quad (33)$$

4. ステップ2以降を繰り返す。

$-\beta R_i = (J/k_B T) \sum s_i \equiv \alpha$ とおくと、

$$\omega^{(i)} = \frac{1}{1 + e^{-2\alpha}} = \frac{1}{2}(\tanh(\alpha) + 1) \rightarrow \begin{cases} \frac{1}{2} & (\alpha = 0) \\ 1 & (\alpha \rightarrow \infty) \\ 0 & (\alpha \rightarrow -\infty) \end{cases} \quad (34)$$

であるから、最近接のスピンが up, down 同数であれば、 s_i の次の状態は1/2でランダムに決まり、up スピンが多ければ s_i も up, down スピンが多ければ s_i も down スピンに移りやすいことが分かる。また、最近接スピンの状態によらず温度が高くなれば s_i の状態はよりランダムに決まることも分かる。

次にこの熱浴法を用いてスピン配位のシミュレーションをし、磁化の温度依存性を数値計算した結果を図14に示す。イジング模型における熱浴法の実装は mathutil.h 内の IsingHeatBathMethod クラスであり、これを用いて磁化の温度依存性を計算したものが isingspinconfig.h 内にある。

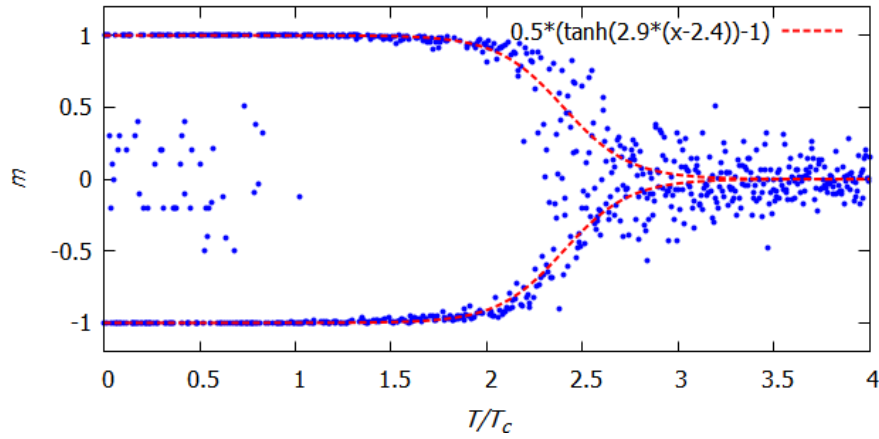


図 14: 熱浴法によるスピン配位のシミュレーションから求めた磁化の温度依存性

条件は図 13 のメトロポリス法と同様で、サイズは 20×20 、熱浴法による更新数 10^6 とした。結果はメトロポリス法によるものと同じである。熱浴法はメトロポリス法と違い、系全体でなく局所的なエネルギーを計算するため計算時間はかなり短くなった。メトロポリス法はエネルギーを計算する際に全スピン数の $O(NM)$ の走査が必要であるが、熱浴法は隣接スピン数の $O(1)$ だけで済む。また、熱浴法は隣接スピンのみから、対象のスピンの遷移確率が決まるため、とびとびの点同士は同時に更新可能であるから並列化して計算できる。

図 13 と図 14 は各温度で異なるランダムな初期値を与えて異なるランダムな更新した。つまり、無秩序な $m \sim 0$ を初期状態とした。 $T/T_c > 2$ で $m = 0$ を中心と分布しているが、これが更新によって収束した結果であるのか、更新数が足りずにまだ収束していない状態なのかを確認したい。そのために、各温度で初期状態を全てのスピンが揃った状態で開始して同じシミュレーションしてみる。また、各温度ごとに乱数のシード値を初期化する。つまり、全ての温度で同じ状態で初期化し、同じように更新されるようにする。そのときの結果を図 15 に示す。プログラムは `isingspinconfig.h` にある。

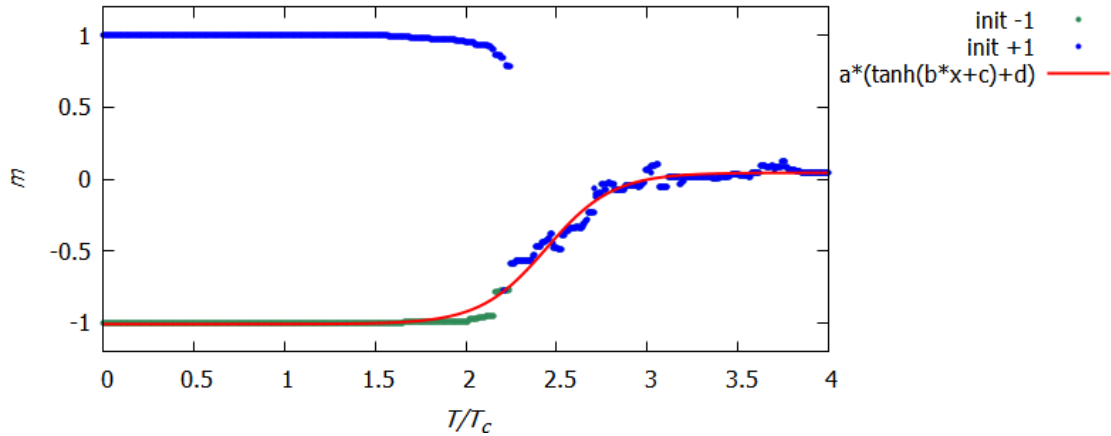


図 15: 熱浴法によるスピン配位のシミュレーションから求めた磁化の温度依存性 (シード値固定)

配位のサイズは 20×20 で更新数は 10^6 回である。青の点がスピンの初期状態を $+1$ にした場合で、緑の点がスピンの初期状態を -1 にした場合である。赤線は初期状態を -1 にした場合のデータを $m = a(\tanh(b(T/T_c) + c)) + d$

c) + d) によってフィッティングしたものである．フィッティングのパラメータは以下ようになった．

$$a = 0.526 \pm 0.002$$

$$b = 2.73 \pm 0.004$$

$$c = -(6.7 \pm 0.1)$$

$$d = -(0.916 \pm 0.006)$$

図より初期値を $m = \pm 1$ にした場合でも $T/T_c > 2$ で $m \sim 0$ であるため，これは更新数が足りずに $m \sim 0$ となっているわけではないことが分かった．初期状態を $+1$ とした青点をみると $T/T_c = 2.25$ から m がいきなりマイナスになっている．図からは分かりにくいですが，それより前にもいくつかマイナスになる点もあった．

3 ニューラルネットワークによる相検出

上でみた熱浴法によるスピン配位を用いて相転移温度を学習し，学習されたパラメータを用いて相検出ができるか確かめる．

まず，データとして，熱浴法で 2 次元正方格子のイジングモデルでスピン配位をシミュレートする．データはランダムな温度で熱浴法によって 10^6 回更新したものをを用いる．データ数は訓練データとテストデータそれぞれ 10000 用意した．転移温度 $T^* = 2.269$ 前後それぞれ 5000 データとした．ラベルは転移温度前後の 2 値分類である．格子サイズは 20×20 とした．データの作成は時間がかかるため，ファイルで保存してあとから読み込めるようにした．プログラムはファイルの `train-isingmodel.h` 内の `createIsingModelDataSet()` 関数にある．

次に上で作成したデータをニューラルネットワークで学習させる．レイヤは以下の構成とした．

$$\text{Affine}(10) \rightarrow \text{BatchNorm} \rightarrow \text{TanhExp} \rightarrow \text{DropOut} \quad (35)$$

$$\rightarrow \text{Affine}(2) \rightarrow \text{BatchNorm} \rightarrow \text{Softmax} \quad (36)$$

隠れ層は 2 層で活性化関数は `tanhExp` を用いた．途中でバッチ正則化とドロップアウトを適用させた．バッチサイズが訓練データ 10000 に対して，20 とした．ハイパラメータをいくつか変えたときの学習の様子を図 16 に示す．

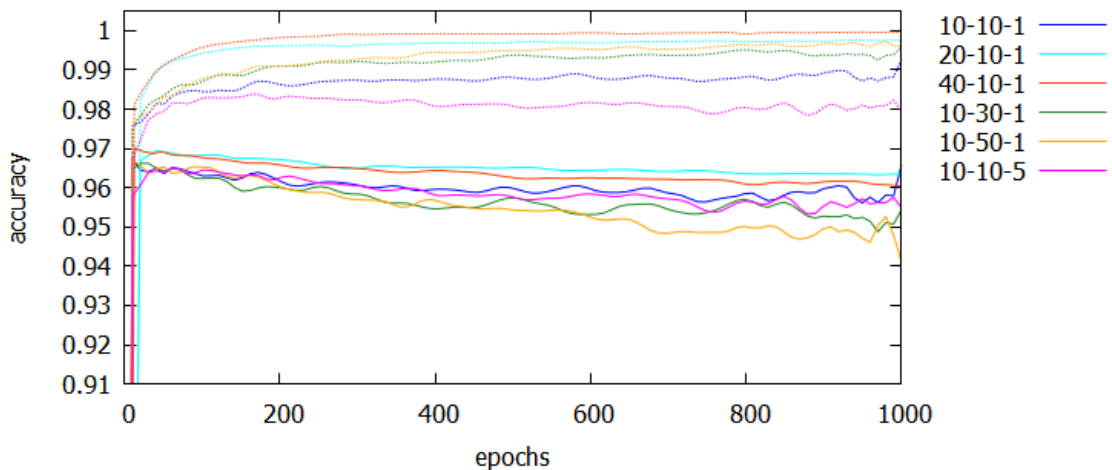


図 16: 2 次元正方格子イジング模型のスピン配位の学習の様子

横軸はエポック数で縦軸が精度である．点線が訓練データの精度で実線がテストデータの精度である．右側の凡例は (バッチ数, Affine レイヤのノード数, 層数) に対応する．層数やノード数，バッチ数を増減させたが，(36) の結果に落ち着いた．テストデータは最高でも 97% 前後となった．テストデータの精度はエ

ボックス数が 70 ほどで最高になり，そこから過学習によって少しずつ下がった。

最後に (36) のレイヤーモデルのネットワークで各温度でシミュレートしたスピン配位を入力として，磁化の値を推論させる．学習データは 2 次元正方格子であったが，同じく 2 次元正方格子のスピン配位の相を推論させる．そのときの結果を図 17 に示す．プログラムはファイル train-isingmodel.h の predictMagnetization() にある．

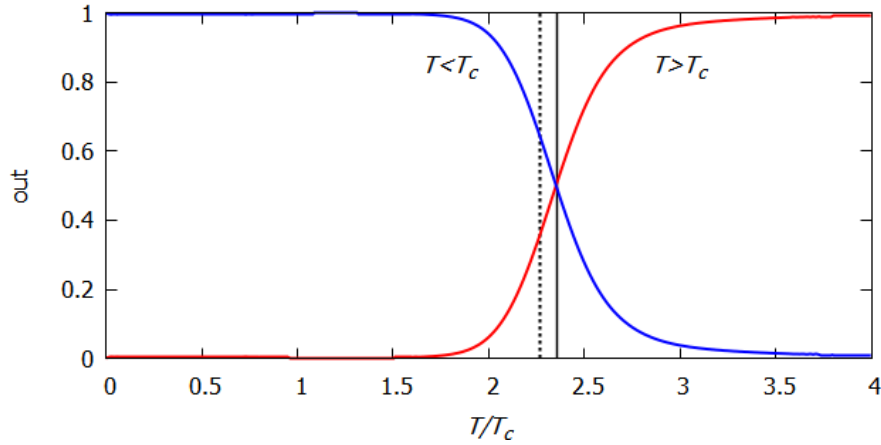


図 17: ニューラルネットワークの 2 次元正方格子イジングモデルの各温度の出力層の値 (2 次元正方格子イジングモデルで学習)

点線が Onsager による厳密解 $T^* = 2.269$ であり，実線が今回ニューラルネットワークの学習によって推論された転移温度 $T = 2.354$ である．縦軸は出力層の値であり，ニューラルネットワークが低温相であると推定した確率である．データは 20 回推論させた平均をベジェ曲線で結んだものである．相対誤差は 3.7% ほどである．おおよそ一致しており，相転移が起きたことは検出できている．厳密解は格子がサイズが無限のときであり，有限のときには厳密値からずれる．学習モデルと推論させるモデルは同じであるからこの結果も納得できる．

[2], [4] によれば，三角格子イジングモデルについても，同じネットワークで相転移温度を検出できるとあった．そこで，詳しいことは書かれていなかったが，正方格子のときと同様にして相検出できるか確かめてみる．熱浴法によって正方格子と同様にサイト (i, j) に対して最近接からのエネルギー寄与は次のように考えられる．

$$\text{偶数行目: } (i-1, j-1) + (i-1, j) + (i, j-1) + (i, j+1) + (i+1, j-1) + (i+1, j)$$

$$\text{奇数行目: } (i-1, 0) + (i-1, j+1) + (i, j-1) + (i, j+1) + (i+1, 0) + (i+1, j+1)$$

正方格子のときと同様に $N \times M$ の格子を考え，偶数行目を右に格子間隔の半分だけずらしたようなものを考える．したがって，正方格子の場合と同様のデータ型で扱える．正方格子の場合から変更する部分は上の熱浴からの寄与だけとなる．熱浴法のプログラムは同様に isingheatbathmethod.h にあり，クラステンプレート引数を変えることによって，近接エネルギーを計算する関数 neighborSpin() が変わるようにしてある．三角格子のスピン配位から相を推論させたときの結果を図 18 に示す．

図 18 より三角格子の場合も相転移が検出されていることが分かる．正方格子の場合よりも高温で相転移している．三角格子の相転移温度は $T^* \sim 3.64$ であるが [2]，このニューラルネットワークによって得られた転移温度の推論は $T = 3.74$ である．相対誤差は 2.7% ほどであり，これもおおよそ一致している．正方格子で学習したニューラルネットワークから三角格子の相検出をすることができた．

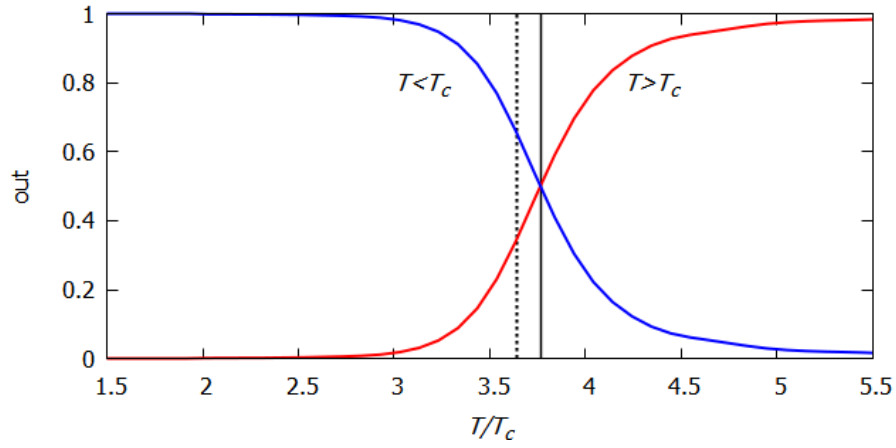


図 18: ニューラルネットワークの2次元正方格子イジングモデルの各温度の出力層の値 (2次元正方格子イジングモデルで学習)

同様に菱形格子についても相検出を行った．サイト (i, j) に対して近接エネルギーは次のようにした．

$$\text{偶数行目: } (i-1, j) + (i-1, j+1) + (i+1, j) + (i+1, j+1)$$

$$\text{奇数行目: } (i-1, j-1) + (i-1, j) + (i+1, j-1) + (i+1, j)$$

このときのニューラルネットワークの出力層の結果を図 19 に示す．

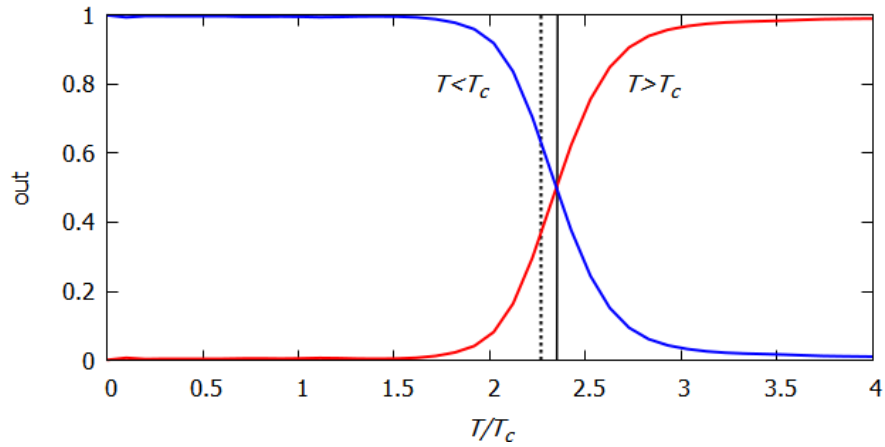


図 19: ニューラルネットワークの2次元菱形格子のイジングモデルの各温度の出力層の値 (2次元正方格子イジングモデルで学習)

相転移温度の結果は，正方格子のときと同じであった．同様に六角格子についても行った．

相変化が $T = 0.209$ と $T = 1.6225$ 付近で見られた．2箇所に見られたのは実装ミスなのか実際そうなのかは分からなかった．

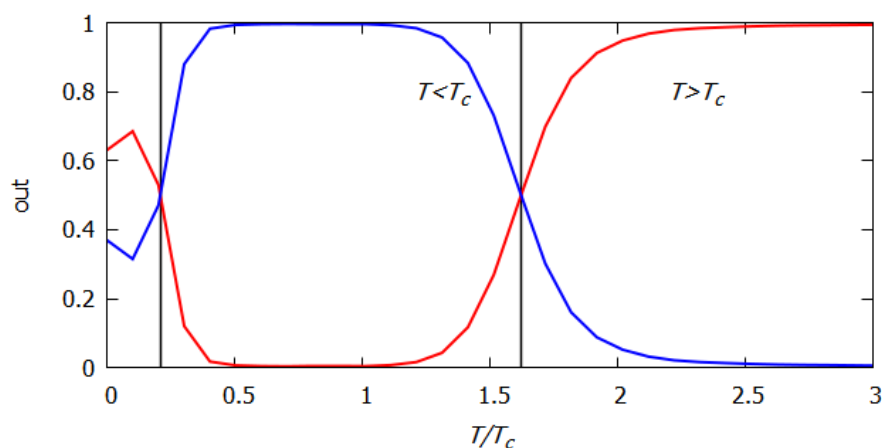


図 20: ニューラルネットワークの 2 次元六角格子イジングモデルの各温度の出力層の (2 次元正方格子イジングモデルで学習)

以上より、ニューラルネットワークによって異なる格子でも相検出を行うことができた。今回は行わなかったが、相転移温度の格子間隔や格子定数の依存性なども気になった。今回は簡単な普通のニューラルネットワークを用いたが、[4]では畳み込みニューラルネットワークを用いて推論されている。CNN を用いればより 3 次元イジングモデルやより複雑なモデルでも精度良く転移温度が得られるのではないかとと思われる。イジングモデルは格子の種類や次元、磁場などの条件によって厳密に解かれていないモデルに対しても、このモンテカルロ法とニューラルネットワークによって解ける可能性があるのではないかと考えられる。

4 参考文献

- [1] 長岡洋介. 「岩波基礎物理学シリーズ 7 統計力学」. 岩波書店, 2003.
- [2] 田中章詞, 富谷昭夫, 橋本幸士. 「ディープラーニングと物理学」. 講談社, 2021.
- [3] 斎藤康毅. 「ゼロから作る Deep Learning — Python で学ぶディープラーニングの理論と実装」. ティムオライリー, 2022.
- [4] A.Tanaka And A.Tomiya. "Detection of phase trainsition via convolutional neural networks." Journal of the Physical Society of Japan