

卒業論文

過剰訓練逆転効果のシミュレーション

三ヶ尻陽一

青山学院大学・理工学部・物理学科
羽田野研究室

2002年度

過剰訓練逆転効果のシミュレーション

三ヶ尻陽一

羽田野研究室

2003年1月20日

概要

心理学の世界で未だはっきりした解決をみていない問題に、過剰訓練逆転効果という現象がある。過剰訓練逆転効果とは、過剰に訓練されたものの方が逆転した課題（今まで正しいとされた反応が間違いとされ、間違いとされた反応が正しいとされるような課題）を早く完成させるという現象である。この過剰訓練逆転効果の仕組みをモデルのシミュレーションから理解するのが本研究の目的である。相互結合型ニューラルネットワークのボルツマンマシンをネズミの脳に見立ててシミュレーションを行なった。その結果、過剰訓練逆転効果を再現することに成功し、過剰訓練逆転効果の起こる要因を一つ示すことができた。

目次

1	はじめに	3
2	過剰訓練逆転効果	4
3	ボルツマンマシンの動作	7
3.1	ホップフィールドネットワークの状態更新	9
3.2	ボルツマンマシンの状態更新	10
3.3	ボルツマンマシンの学習手続き	12
3.4	ボルツマンマシンの学習	14
4	シミュレーションの内容	16
5	学習におけるシナプス結合の変化	17
6	まとめ	23
7	謝辞	24
A	プログラムリスト	26

1 はじめに

1953年、心理学者の Reid が、後に過剰訓練逆転効果と呼ばれる現象を発見した。過剰訓練逆転効果とは、過剰に訓練したもののの方が逆転した課題（今まで正しいとされた反応が間違いとされ、間違いとされた反応が正しいとされるような課題）を早く完成させるという現象である [1]。この現象については次章「過剰訓練逆転効果」で詳しく説明する。訓練をさせればさせるほど訓練の内容を忘れにくくなる、というのがこれまでの常識であったが、過剰訓練逆転効果とはこの常識に真っ向から反するものであった。そのため過剰訓練逆転効果について多くの心理学者によって様々な研究がなされたが、定性的な説明は未だされていない。

一方、脳の数理モデルにニューラルネットワークがある。ニューラルネットワークとは神経回路網を模擬したもの、又はそれに工学的な処理を施したものである。ニューラルネットワークは様々な分野で研究され、また応用されている。本研究ではこのニューラルネットワークの一つ、ボルツマンマシンを脳と見立ててシミュレーションを行なう。まず第2節で過剰訓練逆転効果という現象を具体的に説明する。次に第3節でシミュレーションの用いたボルツマンマシンを定義する。特に、ネットワークを平衡化させる動作と、ネットワークに学習させる動作を分けて説明する。第4節で過剰訓練逆転効果が成功したシミュレーションの結果を述べる。最後に第5節で過剰訓練逆転効果の起こる原因を究明をする。

2 過剰訓練逆転効果

過剰訓練逆転効果とは、過剰に訓練したもののほうが、逆転した課題（今まで正しいとされた反応が間違いとされ、間違いとされた反応が正しいとされるような課題）を早く完成させるという現象である。この現象が認められた実験として心理学者 Reid の行なったネズミの実験がある [1]。それを具体的に例示することによって過剰訓練逆転効果を説明する。

入り口が一つに対して出口が二つの Y 字型の容器を用意する（図 1）。その容器の出口の片方に餌を用意する。もう片方には何も用意しない。Y 字の分岐点にはどちらに餌があるかの手がかりを用意する。Reid の実験においては、手がかりは色である。この容器を用いて 45 匹のネズミに、分岐点に白い札があれば餌は右の出口にある、分岐点に黒い札があれば左の出口に餌がある、という内容の学習をさせる（図 1）。ネズミを Y 字型の容器の入り口にに入れる回数を訓練回数とし、餌に辿り着ければ正しい反応、辿り着けなければ誤りの反応とみなす。ネズミが毎回餌に辿り着けるようになったら（10 回中 9 回以上餌に辿り着け、かつ最後の 5 回は誤りなく餌に辿り着けたら）それを学習の完成とする。

45 匹のネズミ全てが学習を完成したら、ネズミを三群にわけ、仮に三群を A 群、B 群、C 群と名付けよう。A 群のネズミはただちに逆転訓練に移す。この実験において逆転訓練とは、これまでと逆の内容、つまり、分岐点に白い札があれば餌は左の出口、分岐点に黒い札があれば餌は右の出口にあるという内容を学習させる訓練である（図 1）。B 群は、学習が完成してからさらに 50 回多く訓練をしてから逆転訓練に移す。学習が完成しているにも関わらず、同じ内容の訓練を行なうことを以下、過剰訓練と呼ぶ。C 群は、学習完成してからさらに 150 回の過剰訓練をさせてから逆転訓練に移す。そして逆転学習完成までに必要とした訓練回数を調べる。逆転学習完成の基準は、学習完成の基準と同じく、10 回中 9 回ネズミが餌に辿りつけ、かつ最後の 5 回は誤りなく餌に辿りつけることである。

この実験の結果、逆転学習完成までに必要とした逆転訓練の回数は、A 群で平均 138.3 回、B 群で平均 129.0 回、C 群で 70.0 回であった。以上の実験では、過剰訓練の量が多くなるほど逆転学習が早く完成している。この現象を過剰訓練逆転効果という。これは、常識に反する結果である。なぜなら、従来は、学習をすればするほど学習内容は忘れにくくなると思われてきたからだ [1]。

過剰訓練逆転効果の原因は未だに明らかにされていない。そこで本研

究では、ボルツマンマシンを用いて過剰訓練逆転効果を再現し、その要因を探った。

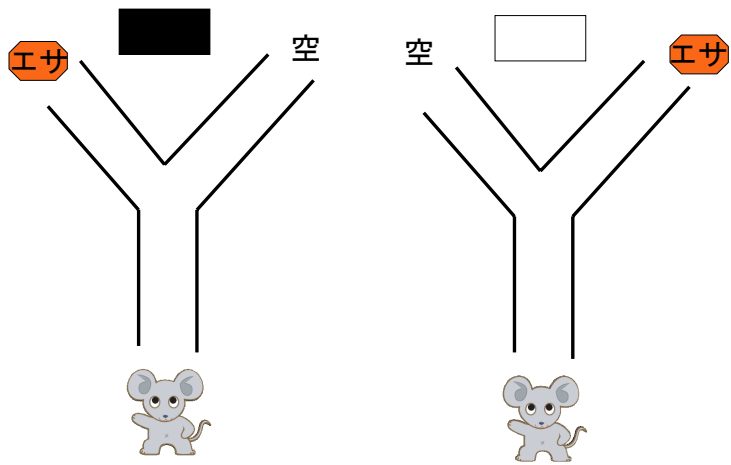
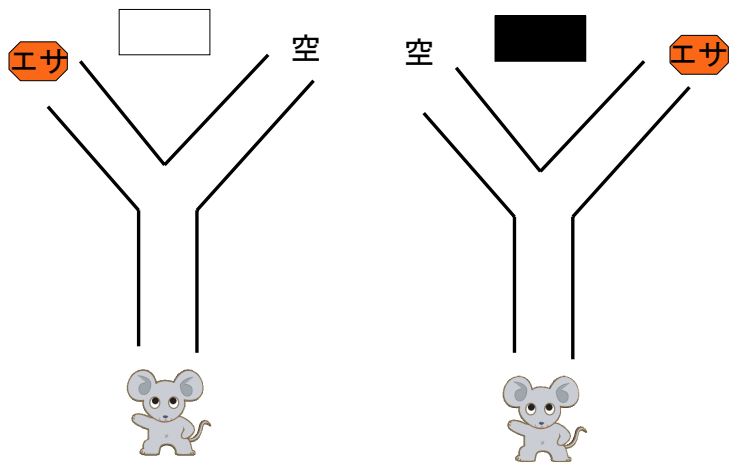


図 1: 上段の左右が Y 字型の容器での学習。下段の左右が Y 字型容器での逆転学習



3 ボルツマンマシンの動作

ここでは本研究に用いたボルツマンマシンの説明をする。ボルツマンマシンとは、脳の数理モデルであるニューラルネットワークの一つである。ニューラルネットワークは神経細胞に相当するいくつかの素子と、それらを結ぶシナプス結合からなる。ニューラルネットワークには平衡状態に達するまでの状態更新動作と、その平衡状態を都合のいいように変えるための学習操作がある。ここではまず、ボルツマンマシンの元になったホップフィールドネットワークについて説明し、次にホップフィールドネットワークとボルツマンマシンの違いに注意しつつ、ニューラルネットワークが平衡状態に達する動作を説明する。最後にボルツマンマシンの学習動作について説明する。

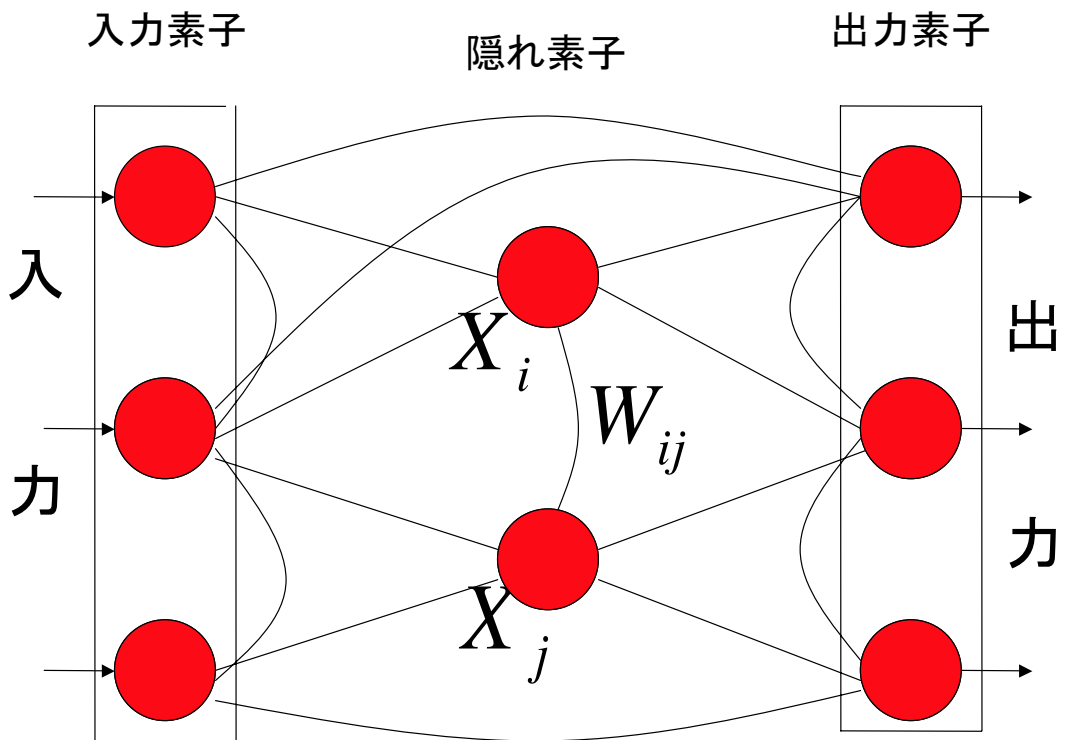


図 2: ホップフィールドモデルやボルツマンマシンの構造 [2]。実際には素子は他の全ての素子と繋がれている。左の3つは入力素子、右の3つは出力素子、中央の2つは情報の処理にのみ用いられる隠れ素子と呼ぶ。

3.1 ホップフィールドネットワークの状態更新

ホップフィールドネットワークは相互結合型ニューラルネットワークである。最も簡単なホップフィールドネットワークにおいては、各素子は、他の全ての素子と相互に結合している (図 2)。素子は 0 か 1 の 2 状態をとり、相互結合している全ての素子の状態によって自らの状態を更新する。以下において i 番目の素子の状態を X_i 、素子 i から素子 j へのシナプス結合の強さを W_{ij} と表記する。ネットワークには、幾つかの入力素子と出力素子がある。入力素子に特定の状態を入力すると、ある状態更新の後に、出力素子が持っている状態を出力するのがニューラルネットワークの機能である。

ここで n 個の素子を持つホップフィールドネットワークの状態更新について簡単に説明する。まず、入力素子に 0 か 1 の値を代入し固定する。それ以外の素子は任意の値を代入する (初期値)。そして以下の動作を繰り返す。

1. 入力素子以外の素子から一様分布乱数で一個の素子 k をランダムに選ぶ。
2. 以下の式に従って周囲の素子から素子 k への入力 S_k を計算する：

$$S_k = \sum_{i=0}^n W_{ik} X_i \quad (1)$$

3. 素子 k への入力 S_k が素子 k の閾値 θ_k を越えたら素子 k の値を 1 にし、越えなければ素子 k の値を 0 にする。

以上のルールで状態更新を繰り返すと、ネットワークの「エネルギー」に相当する評価関数

$$E = - \sum_{i,j=0}^n W_{ij} X_i X_j + \sum_{i=0}^n \theta_i X_i \quad (2)$$

を下げることを示せる。十分に状態更新を行なうと最終的に、入力素子を固定した条件付きのエネルギー最低の状態に落ち着くと期待される。この状態の出力素子の値がホップフィールドネットワークの出力である。

3.2 ボルツマンマシンの状態更新

ホップフィールドネットワークは評価関数 E を下げる方向にしか状態更新できない。場合によっては評価関数の極小値にはまり、エネルギー最低状態でない出力を返すことがある。極小にはまることを緩和するために、ボルツマンマシンでは各素子の状態を確率的に更新する。

今、周囲の素子から素子 k への入力 S_k であったとする。ボルツマンマシンでは、第 3.1 節ステップ 3 の代わりに、素子 k は

$$P_k(0) = 1 - \text{sigmoid}(S_k - \theta_k) \quad (3)$$

$$P_k(1) = \text{sigmoid}(S_k - \theta_k) \quad (4)$$

で状態更新を行なう。ここで $P_k(0)$ は素子の値が 0 になる確率、また $P_k(1)$ は素子の値が 1 になる確率である。また sigmoid 関数とは、

$$\text{sigmoid}(z) = \frac{1}{1 + \exp(-z/T)} \quad (5)$$

である。 T はネットワークの温度に相当するパラメータ、また θ_k は sigmoid 関数の閾値である。素子への入力と状態更新確率の関係がどのように温度に依存するかを図 3 に示す。

温度が高い時は素子はほとんどランダムに状態更新を行なう。一方、温度が 0 のときはホップフィールドネットワークと同じように振る舞う。温度 T を固定して十分に状態更新を行なうと、その温度に対応した平衡状態に落ち着くことが期待される。平衡状態とは、初期値に依存しなくなり出力が一定値に収束することである。

平衡状態を保ちながら徐々に温度を下げていくことを焼きなましと呼ぶ。焼きなましをすることによって評価関数 E が極小値にはまらずに最小値付近に達しやすくしている。こうして最終的に入力素子固定のエネルギー最小状態を得ると、その時の出力素子の値がボルツマンマシンの出力である。

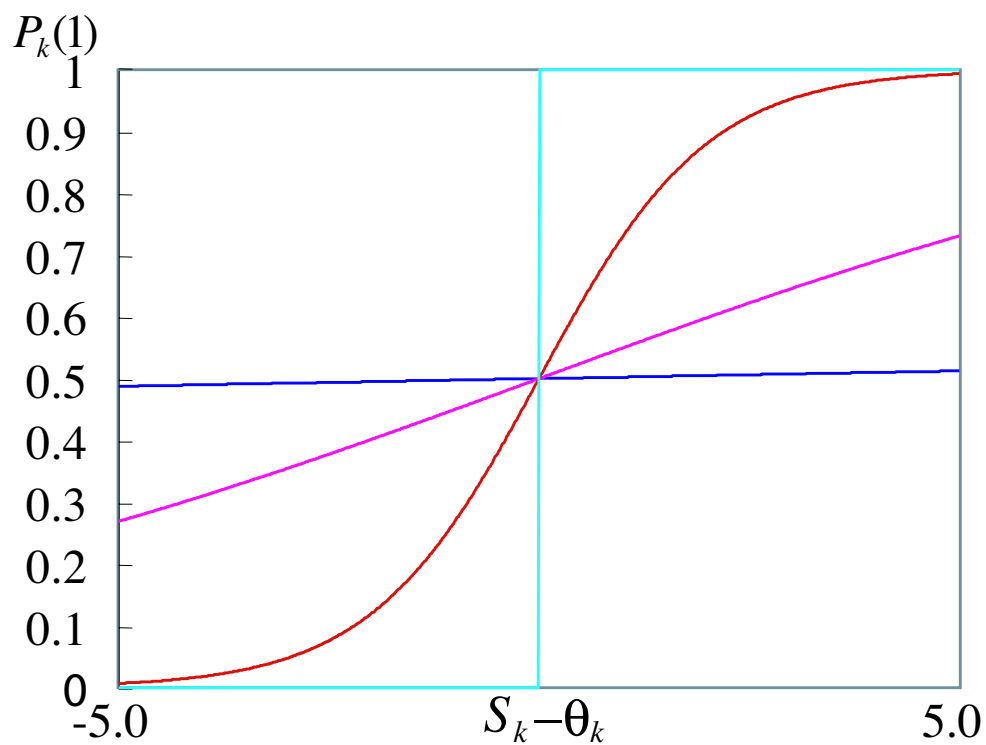


図 3: 周囲素子からの入力と素子の値が 1 になる確率 $P(1)$ の関係。横軸は入力の値から閾値を引いた値、縦軸は素子の値が 1 になる確率を表す。藍色は $T = 100$, ピンクは $T = 5$, 赤は $T = 1$, 水色は $T = 0$ の場合。

3.3 ボルツマンマシンの学習手続き

ボルツマンマシンは、十分な回数、第 3.2 節の状態遷移を繰り返すと、始めに割り当てられた素子値（初期値）に依存しない平衡状態を生成するようになる。ボルツマンマシンは、シナプス結合の強さ W_{ij} と閾値 θ_k の与え方に応じて異なる平衡状態を与える。この平衡状態が、入力素子値に対して望ましい出力値を出せるように、シナプス結合の強さ W_{ij} と閾値 θ_k を変化させるのがボルツマンマシンの学習である [3]。本研究では、シナプス結合の強さ W_{ij} にのみ注目し、閾値は全て 0 とする。

以下シナプス結合の強さ W_{ij} 更新する手続き（学習）を説明する。

1. 入出力パターンの指定

学習させたい入出力のパターン（複数）から一つを選ぶ。

2. 入出力固定モードにおける状態更新

入出力素子の値をステップ 1 で定めた値に固定した状態で、ボルツマンマシンを状態更新させる。入出力以外の素子（隠れ素子）には、初期値を適当に与える。隠れ素子をランダムな順に状態更新する。全素子の状態更新を平均して一巡したとき、1 サイクルの状態更新が行なわれたものとする。十分なサイクル数について状態更新を繰り返し、平衡状態を実現する。平衡状態において、各シナプス結合 W_{ij} の両端の素子 X_i と X_j が同時に 1 となる頻度を調べる。この頻度を p_{ij}^{fix} と記す。例えば、100 サイクルの状態更新の内、素子 i と素子 j が同時に 1 となる回数が 80 であったとすると $p_{ij}^{\text{fix}} = 0.8$ である。

3. 入力素子値固定、出力素子値自由モードにおける状態更新

今度は、入力素子はステップ 1 で定めた値に固定するが出力素子値は固定しない。出力素子と隠れ素子の初期値は適当に与える。入力素子以外の素子は素子間のシナプス結合によって自由に変わる状況でボルツマンマシンを状態更新させる。十分なサイクル数、状態更新を繰り返し、平衡に到達する。平衡状態でシナプス結合 W_{ij} の両端の素子 X_i と X_j が同時に 1 になる頻度を調べる。この頻度を p_{ij}^{free} と記す。

4. ステップ 1、2、3 を繰り返す。

5. シナプス結合の強さの修正

次式により各シナプス結合 W_{ij} の修正を行なう：

$$W_{ij}^{\text{new}} = W_{ij}^{\text{old}} + \frac{\epsilon}{T}(p_{ij}^{\text{fix}} - p_{ij}^{\text{free}}) \quad (6)$$

ここで ϵ は十分小さな正の実数である。今回のシミュレーションにおいては 0.00125 とした。

以上の 5 つのステップを多数繰り返し、結線の重みを少しずつ修正していく [3]。これがボルツマンマシンの学習である。

3.4 ボルツマンマシンの学習

前節で紹介した手続きでボルツマンマシンが学習できる理由を示す。望ましい入出力パターンを全て実現できる状態（つまり入出力固定の時の平衡状態）のボルツマン分布を P^+ と書き、一方入力素子固定、出力素子自由のネットワークの平衡状態のボルツマン分布を $P^-(\{X_i\})$ と書く：

$$P^-(\{X_i\}) = \frac{\exp(-E/T)}{\sum_{X_i=0,1} \exp(-E/T)} \quad (7)$$

この両者の確率分布の「距離」を表現するものとして次のカルバック情報量を定義する：

$$G = \sum_{X_i=0,1} P^+ \ln[P^+/P^-] \quad (8)$$

この情報量は常に非負であり $P^+ = P^-(\{X_i\})$ のときのみ 0 になる。ここでネットワークのシナプス結合強度 W_{ij} を調整して $P^-(\{X_i\})$ を P^+ に近づけるように試みる。このとき式 (8) は結合強度 $W\{W_{ij}\}$ の汎関数である。従って

$$G = G[\{W_{ij}\}] = \sum_{X_i=0,1} P^+ \ln[P^+/P^-] \quad (9)$$

と表現できる。結局ボルツマンマシンによる学習とは、カルバック情報量の結合強度 W_{ij} に関する最小化問題として定式化できる。そこで、 W_{ij} を変化させた時に G がどの程度減少するかを調べる。情報量 $G[W_{ij}]$ の W_{ij} に対する感度は

$$\frac{\partial G}{\partial W_{ij}} = - \sum_{X_i=0,1} P^+ \frac{\partial}{\partial W_{ij}} \ln P^- \quad (10)$$

$$= -\frac{1}{T} \sum_{X_i=0,1} P^+ X_i X_j + \frac{1}{T} \sum_{X_i=0,1} P^- X_i X_j \quad (11)$$

となる。なお式変形において

$$\sum_{X_i=0,1} P^\pm = 1 \quad (12)$$

を用いた。さらに簡単のために

$$p_{ij}^+ = \sum_{X_i=0,1} P^+ X_i X_j \quad (13)$$

$$p_{ij}^- = \sum_{X_i=0,1} P^- X_i X_j \quad (14)$$

という表記を用いると、結局

$$\frac{\partial G}{\partial W_{ij}} = -\frac{1}{T}(p_{ij}^+ - p_{ij}^-) \quad (15)$$

となる。よって W_{ij} を

$$W_{ij} \rightarrow W_{ij} - \epsilon \frac{\partial G}{\partial W_{ij}} \quad (16)$$

と変化させると、 G を最小化する方向へ向かう。

ところで式 (13) によると p_{ij}^+ は入出力素子を固定したときシナプス結合 W_{ij} の両端にある素子 X_i と X_j が同時に 1 になっている頻度を表す。同様に式 (14) によると p_{ij}^- は入力素子固定出力素子自由にした時シナプス結合 W_{ij} の両端にある素子 X_i と X_j が同時に 1 になっている頻度を表す [2]。これより式 (6) でボルツマンマシンが学習できる (G を 0 に近づけられる) ことがわかる。

4 シミュレーションの内容

今回用いたボルツマンマシンは入力素子7個、出力素子3個、隠れ素子60個から成る。ボルツマンマシンに表1の左側に示す8つのランダムな入出力パターンを同じ回数で学習させる。結合の強さ W_{ij} を更新するごとに、8つの内容について100回ずつテストを行なう。テストとはボルツマンマシンの入力素子値を学習内容の値に固定し、望ましい出力が出ているか確認する作業である。全ての内容についてテストに三回連続で9割より高い正解率を出した時を学習の完成とする。学習が完成したボルツマンマシンを3つ用意し、それぞれA、B、Cと名付ける。ボルツマンマシンAはすぐに表1右側に従って逆転学習に移し、ボルツマンマシンBは100回の過剰学習の後、逆転学習に移し、ボルツマンマシンCは200回の過剰学習の後、逆転学習に移した。逆転学習完成までに要した学習回数を調べた。

逆転学習完成に必要なとした学習回数は、ボルツマンマシンAでは113回、ボルツマンマシンBでは80回、ボルツマンマシンCでは48回であった。このシミュレーションの結果より、過剰に学習させたものの方が逆転学習を早く完成していることがわかる。つまり過剰訓練逆転効果が再現できた。

表 1: まず左側の表に従って学習させる。逆転学習は右側に従う。

入力素子 (7 個)	出力素子 (3 個)	入力素子 (7 個)	出力素子 (3 個)
1011110	101	1011110	010
0100101	010	0100101	101
0111110	011	0111110	011
1001011	100	1001011	100
0000000	000	0000000	000
0011011	001	0011011	001
1111110	111	1111110	111
1101100	110	1101100	110

5 学習におけるシナプス結合の変化

過剰訓練逆転効果の仕組みを探るため、学習完成後における各素子間のシナプス結合の強さを色分けしたグラフ図 4 を示す。結合 W_{ij} を行列表示して色分けした。三角形の下 7 行が入力素子と他の素子の結合の強さを表し、三角形の右 3 列が出力素子と他の素子との結合に強さを表す。

ボルツマンマシン A は、学習完成後、ただちに逆転訓練を開始した。その逆転訓練を完成したときのシナプス結合の強さの様子が図 5 である。

一方、ボルツマンマシン C には、図 4 の状況（学習完成）からさらに 200 回の過剰学習をさせた。この時の各素子間の結合の強さを示すのが図 6 である。この状況から逆転学習を開始し、それが完成したときの状況が図 7 である。

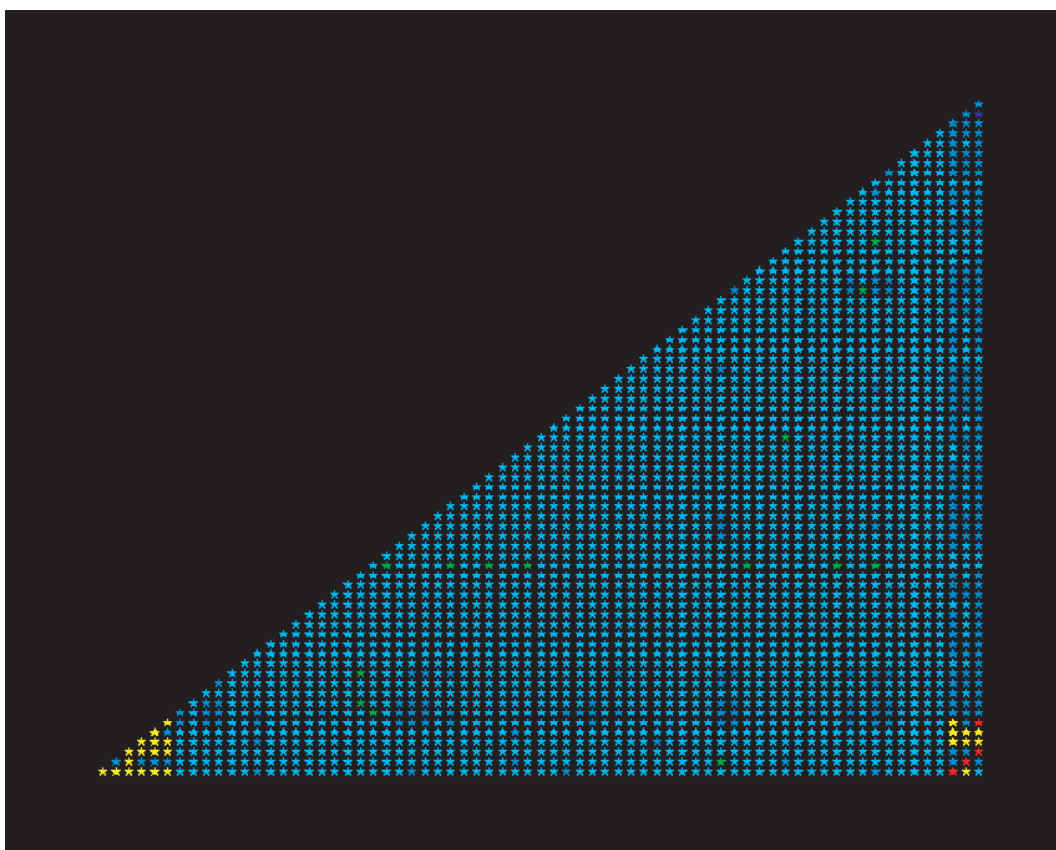


図 4: 学習完成後の各素子間の結合の強さ。青は $W_{i,j} \leq -0.1$ 、青 + シアンは $-0.1 \leq W_{i,j} \leq -0.01$ 、シアンは $-0.01 \leq W_{i,j} \leq -0.001$ 、緑は $0.001 \leq W_{i,j} \leq 0.001$ 、黄は $0.001 \leq W_{i,j} \leq 0.01$ 、赤 + 黄は $0.01 \leq W_{i,j} \leq 0.01$ 、赤は $0.1 \leq W_{i,j}$ 。

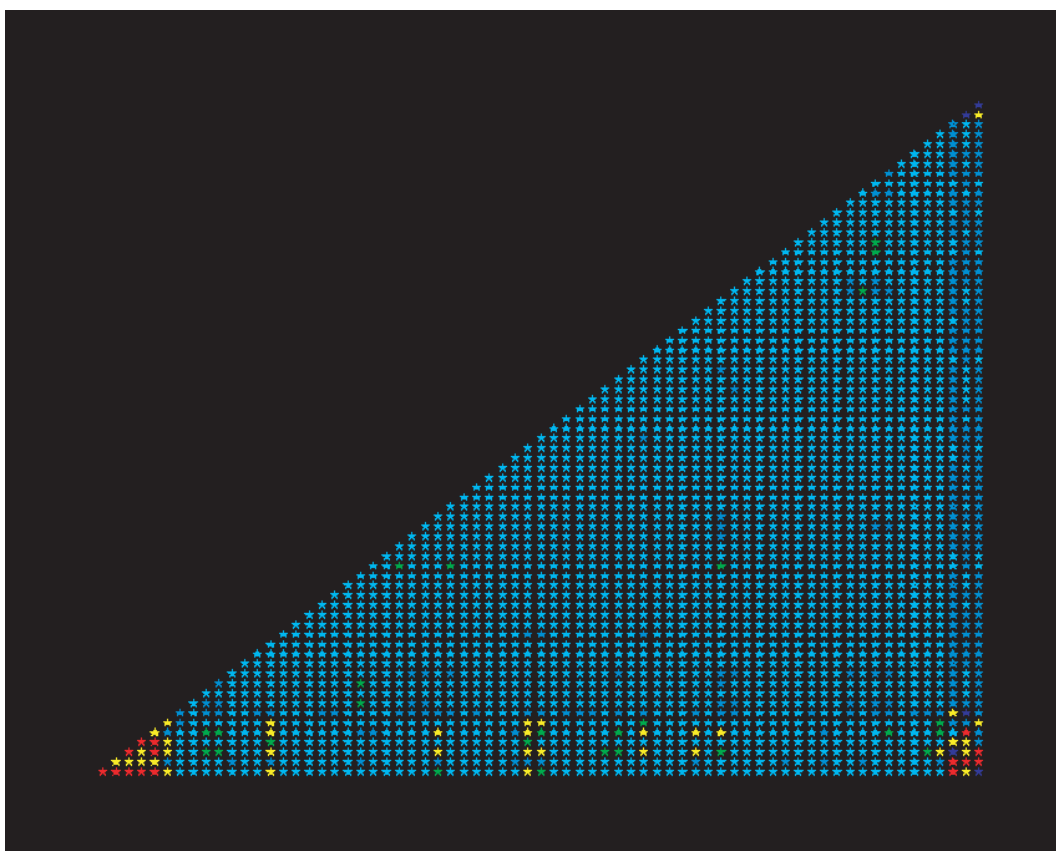


図 5: ボルツマンマシン A の逆転学習完成後の各素子間の結合の強さ。青は $W_{ij} \leq -0.1$ 、青+シアンは $-0.1 \leq W_{ij} \leq -0.01$ 、シアンは $-0.01 \leq W_{ij} \leq -0.001$ 、緑は $0.001 \leq W_{ij} \leq 0.001$ 、黄は $0.001 \leq W_{ij} \leq 0.01$ 、赤+黄は $0.01 \leq W_{ij} \leq 0.01$ 、赤は $0.1 \leq W_{ij}$ 。

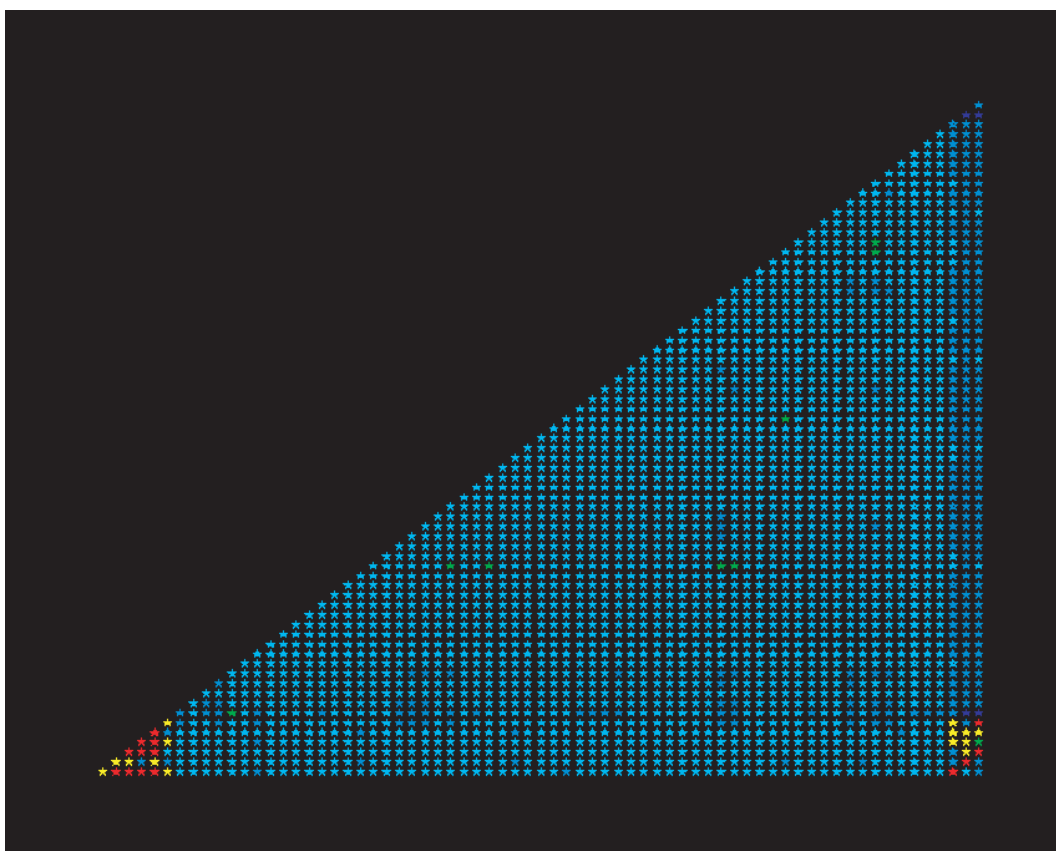


図 6: ボルツマンマシン C の過剰学習直後の各素子間の結合の強さ。青は $W_{ij} \leq -0.1$ 、青+シアンは $-0.1 \leq W_{ij} \leq -0.01$ 、シアンは $-0.01 \leq W_{ij} \leq -0.001$ 、緑は $0.001 \leq W_{ij} \leq 0.001$ 、黄は $0.001 \leq W_{ij} \leq 0.01$ 、赤+黄は $0.01 \leq W_{ij} \leq 0.01$ 、赤は $0.1 \leq W_{ij}$ 。

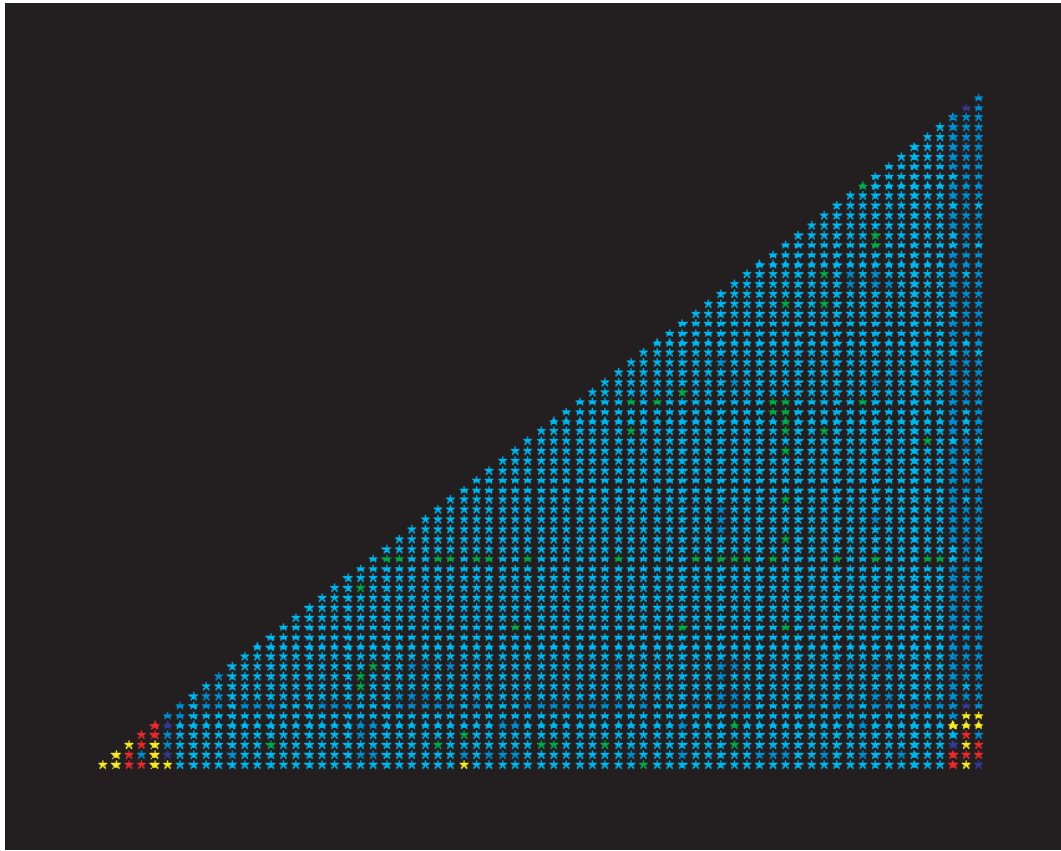


図 7: ボルツマンマシン C の逆転学習完成後の各素子間の結合の強さ。青は $W_{ij} \leq -0.1$ 、青+シアンは $-0.1 \leq W_{ij} \leq -0.01$ 、シアンは $-0.01 \leq W_{ij} \leq -0.001$ 、緑は $0.001 \leq W_{ij} \leq 0.001$ 、黄は $0.001 \leq W_{ij} \leq 0.01$ 、赤+黄は $0.01 \leq W_{ij} \leq 0.01$ 、赤は $0.1 \leq W_{ij}$ 。

通常の学習の直後 (図 3 及び図 5) では、入力素子と他の素子の結合は主に負である。ボルツマンマシン A では、それら負の結合を正に変えることによって逆転学習を完成させている (図 4)。

一方、ボルツマンマシン B では、一部の負の結合をむしろ強くすることによって逆転学習を完成させている。また、ボルツマンマシン A と比べると全般的に変化が小さい。この小さい変化のため、逆転学習が早く完成していると思われる。

6 まとめ

脳の数理モデルであるボルツマンマシンをシミュレーションして、過剰訓練逆転効果を再現した。通常の学習の後、すぐに逆転学習させたものよりも、過剰学習の後に逆転学習をさせたものの方がネットワークにもたらず変化が少なく逆転学習を完成させていることが分かった。

なぜ過剰学習から逆転学習にする過程の方が変化が少なく逆転学習を完成できるかについて、ヒントを示すことができた。しかしその本質は未解明である。評価関数 E の凹凸が深く関わっていると思われるので評価関数に関する解析を行なう必要がある。

最後に、本研究において、ボルツマンマシンをネズミの脳に見立ててシミュレーションを行なったが、ボルツマンマシンはかなり工学的な処理がなされているので厳密には脳のモデルではない。実際の脳との比較が待たれる。

7 謝辞

羽田野先生には、研究の方向を見失いかけた時や、的外れなことをしようとしている時、何をすればいいかわからず立ち止まっている時など、的確なアドバイスで研究を導いて頂き大変お世話になりました。この場をかりて感謝の意を表したいと思います。また、先輩方、同級生の方々にもあつく御礼申し上げます。どうもありがとうございました。

参考文献

- [1] 本吉良治 『講座心理学第6巻』(東京大学出版会、1969年)
- [2] 馬場則夫、小島史男、小澤誠一 『ニューラルネットの基礎と応用』(共立出版株式会社、1994年)
- [3] 熊沢逸夫 『学習とニューラルネットワーク』(森北出版株式会社、1998年)

A プログラムリスト

過剰学習から逆転学習への過程における各素子間結合を描くプログラム

```
#include<stdio.h>
#include<stdlib.h>
#include<math.h>
#include<time.h>
#define SIMPLE_SPRNG
#include"sprng.h"
#include"cpgplot.h"
#define SEED 53575463
#define sosi_input 7
#define sosi_output 3
#define sosi_hidden 60
#define sosisuu (sosi_input+sosi_output+sosi_hidden)
#define NUM_UPDATE 400
#define TEMPERATURE 10
#define T_END 0.01
#define EPSILON 0.00125
float sgn(float y,float T);
int main(void)
{
    static int x[sosisuu],a[sosisuu][sosisuu],wp[sosisuu][sosisuu],
wn[sosisuu][sosisuu],ikichi[sosisuu];
    int over,update,learn,i,j,l,k,m,o,issuu,random,random_number;
    int kannsei,n1,n2,n3,n4,n5,n6,n7,n8,q1,q2,q3,q4,q5,q6,q7,q8;
    float w[sosisuu][sosisuu],y,T,epsilon,r,s;
    float mark_x[7][sosisuu*sosisuu],mark_y[7][sosisuu*sosisuu];
    int color,number[7];
    epsilon=EPSILON;
    init_sprng(DEFAULT_RNG_TYPE,SEED,SPRNG_DEFAULT);
    for(i=0;i<sosisuu;i++)
    {
        a[i][i]=0;
        w[i][i]=0.0;
```

```

    }
    for(i=0;i<sosisuu;i++)
    {
        for(j=0;j<i;j++)
    {
        a[i][j]=1;
        a[j][i]=a[i][j];
        w[i][j]=0.0;
        w[j][i]=w[i][j];
    }
    }
    for(i=0;i<sosisuu;i++)
    {
        ikichi[i]=0;
        for(j=0;j<sosisuu;j++)
    {
        ikichi[i]=ikichi[i]+w[i][j];
    }
    }
    n1=0;n2=0;n3=0;n4=0;n5=0;n6=0;n7=0;n8=0;
    kannsei=0;
    for(learn=0;kannsei<3;learn++)
    {
        for(i=0;i<sosisuu;i++)
    {
        for(j=0;j<i;j++)
        {
            wp[i][j]=0;
            wp[j][i]=0;
            wn[i][j]=0;
            wn[j][i]=0;
        }
    }
    }
    for(o=0;o<80;o++)
    {

```

```

random=(int)(o/10);
if(random==0)
{
    x[0]=1;
    x[1]=0;
    x[2]=1;
    x[3]=1;
    x[4]=1;
    x[5]=1;
    x[6]=0;
    x[sosisuu-3]=1;
    x[sosisuu-2]=0;
    x[sosisuu-1]=1;
}
else if(random==1)
{
    x[0]=0;
    x[1]=1;
    x[2]=0;
    x[3]=0;
    x[4]=1;
    x[5]=0;
    x[6]=1;
    x[sosisuu-3]=0;
    x[sosisuu-2]=1;
    x[sosisuu-1]=0;
}
else if(random==2)
{
    x[0]=0;
    x[1]=1;
    x[2]=1;
    x[3]=1;
    x[4]=1;
    x[5]=1;

```

```

        x[6]=0;
        x[sosisuu-3]=0;
        x[sosisuu-2]=1;
        x[sosisuu-1]=1;
    }
else if(random==3)
    {
        x[0]=1;
        x[1]=0;
        x[2]=0;
        x[3]=1;
        x[4]=0;
        x[5]=1;
        x[6]=1;
        x[sosisuu-3]=1;
        x[sosisuu-2]=0;
        x[sosisuu-1]=0;
    }
else if(random==4)
    {
        x[0]=0;
        x[1]=0;
        x[2]=0;
        x[3]=0;
        x[4]=0;
        x[5]=0;
        x[6]=0;
        x[sosisuu-3]=0;
        x[sosisuu-2]=0;
        x[sosisuu-1]=0;
    }
else if(random==5)
    {
        x[0]=0;
        x[1]=0;

```

```

    x[2]=1;
    x[3]=1;
    x[4]=0;
    x[5]=1;
    x[6]=1;
    x[sosisuu-3]=0;
    x[sosisuu-2]=0;
    x[sosisuu-1]=1;
}
else if(random==6)
{
    x[0]=1;
    x[1]=1;
    x[2]=1;
    x[3]=1;
    x[4]=1;
    x[5]=1;
    x[6]=0;
    x[sosisuu-3]=1;
    x[sosisuu-2]=1;
    x[sosisuu-1]=1;
}
else if(random==7)
{
    x[0]=1;
    x[1]=1;
    x[2]=0;
    x[3]=1;
    x[4]=1;
    x[5]=0;
    x[6]=0;
    x[sosisuu-3]=1;
    x[sosisuu-2]=1;
    x[sosisuu-1]=0;
}

```

```

for(l=sosi_input;l<sosi_input+sosi_hidden;l++)
{
    random_number=(int)(2.0*sprng());
    if(random_number==0)
{
    x[l]=1;
}
    else
{
    x[l]=0;
}
}
for(T=TEMPERATURE;T>T_END;T=T*0.8)
{
    for(issuu=0;issuu<(sosi_hidden)*10/T;issuu++)
{
    k=(int)(sprng()*sosi_hidden);
    i=k+sosi_input;
    y=0.0;
    for(j=0;j<sosisuu;j++)
    {
        y=y+w[i][j]*x[j];
    }
    s=sgn(y,T);
    r=sprng();
    if(r<s)
    {
        x[i]=1;
    }
    else
    {
        x[i]=0;
    }
}
}

```



```

for(update=0;update<NUM_UPDATE;update++)
    {
        for(issuu=0;issuu<2*sosi_hidden;issuu++)
    {
k=(int)(sprng()*sosi_hidden);
i=k+sosi_input;
y=0.0;
for(j=0;j<sosisuu;j++)
    {
        y=y+w[i][j]*x[j];
    }
s=sgn(y,T);
r=sprng();
if(r<s)
    {
        x[i]=1;
    }
else
    {
        x[i]=0;
    }
}
        for(i=0;i<sosisuu;i++)
    {
        for(j=0;j<i;j++)
            {
                if(x[i]==1 && x[j]==1 && a[i][j]!=0.0)
            {
                wp[i][j]=wp[i][j]+1;
                wp[j][i]=wp[j][i];
            }
        }
    }
}
}

```

```

        for(o=0;o<80;o++)
{
random=(int)(sprng()*8.0);
if(random==0)
    {
        x[0]=1;
        x[1]=0;
        x[2]=1;
        x[3]=1;
        x[4]=1;
        x[5]=1;
        x[6]=0;
    }
else if(random==1)
    {
        x[0]=0;
        x[1]=1;
        x[2]=0;
        x[3]=0;
        x[4]=1;
        x[5]=0;
        x[6]=1;
    }
else if(random==2)
    {
        x[0]=0;
        x[1]=1;
        x[2]=1;
        x[3]=1;
        x[4]=1;
        x[5]=1;
        x[6]=0;
    }
else if(random==3)

```

```
{
    x[0]=1;
    x[1]=0;
    x[2]=0;
    x[3]=1;
    x[4]=0;
    x[5]=1;
    x[6]=1;
}
else if(random==4)
{
    x[0]=0;
    x[1]=0;
    x[2]=0;
    x[3]=0;
    x[4]=0;
    x[5]=0;
    x[6]=0;
}
else if(random==5)
{
    x[0]=0;
    x[1]=0;
    x[2]=1;
    x[3]=1;
    x[4]=0;
    x[5]=1;
    x[6]=1;
}
else if(random==6)
{
    x[0]=1;
    x[1]=1;
    x[2]=1;
    x[3]=1;
```

```

        x[4]=1;
        x[5]=1;
        x[6]=0;
    }
else if(random==7)
    {
        x[0]=1;
        x[1]=1;
        x[2]=0;
        x[3]=1;
        x[4]=1;
        x[5]=0;
        x[6]=0;
    }
for(l=sosi_input;l<sosisuu;l++)
    {
        random_number=(int)(sprng()*2.0);
        if(random_number==0)
{
    x[1]=1;
}
        else
{
    x[1]=0;
}
    }
for(T=TEMPERATURE;T>T_END;T=T*0.8)
    {
        for(issuu=0;issuu<(sosi_hidden+sosi_output)*10/T;issuu++)
{
    k=(int)(sprng()*(sosi_hidden+sosi_output));
    i=k+sosi_input;
    y=0.0;
    for(j=0;j<sosisuu;j++)
        {

```

```

        y=y+w[i][j]*x[j];
    }
s=sgn(y,T);
r=sprng();
if(r<s)
    {
        x[i]=1;
    }
else
    {
        x[i]=0;
    }
}
}
for(update=0;update<NUM_UPDATE;update++)
    {
        for(issuu=0;issuu<2*(sosi_hidden+sosi_output);issuu++)
        {
            k=(int)(sprng()*(sosi_hidden+sosi_output));
            i=k+sosi_input;
            y=0.0;
            for(j=0;j<sosisuu;j++)
                {
                    y=y+w[i][j]*x[j];
                }

            s=sgn(y,T);
            r=sprng();
            if(r<s)
                {
                    x[i]=1;
                }
            else
                {
                    x[i]=0;
                }
        }
    }

```

```

    }
}
    for(i=0;i<sosisuu;i++)
{
    for(j=0;j<i;j++)
    {
        if(x[i]==1 && x[j]==1 && a[i][j]!=0.0)
    {
        wn[i][j]=wn[i][j]+1;
        wn[j][i]=wn[j][i];
    }
    }
}
    for(i=0;i<sosisuu;i++)
{
    for(j=0;j<i;j++)
    {
        w[i][j]=w[i][j]+(epsilon/T_END*(wp[i][j]-wn[i][j]))/NUM_UPDATE/o;
        w[j][i]=w[i][j];
    }
}
n1=0;
n2=0;
n3=0;
n4=0;
n5=0;
n6=0;
n7=0;
n8=0;
for(m=0;m<800;m++)
{
    if(m<100)
{

```

```
x[0]=1;
  x[1]=0;
  x[2]=1;
  x[3]=1;
  x[4]=1;
  x[5]=1;
  x[6]=0;
}
    else if(m<200)
{
  x[0]=0;
  x[1]=1;
  x[2]=0;
  x[3]=0;
  x[4]=1;
  x[5]=0;
  x[6]=1;
}
    else if(m<300)
{
  x[0]=0;
  x[1]=1;
  x[2]=1;
  x[3]=1;
  x[4]=1;
  x[5]=1;
  x[6]=0;
}
    else if(m<400)
{
  x[0]=1;
  x[1]=0;
  x[2]=0;
  x[3]=1;
  x[4]=0;
```

```
        x[5]=1;
        x[6]=1;
    }
        else if(m<500)
    {
        x[0]=0;
        x[1]=0;
        x[2]=0;
        x[3]=0;
        x[4]=0;
        x[5]=0;
        x[6]=0;
    }
        else if(m<600)
    {
        x[0]=0;
        x[1]=0;
        x[2]=1;
        x[3]=1;
        x[4]=0;
        x[5]=1;
        x[6]=1;
    }
        else if(m<700)
    {
        x[0]=1;
        x[1]=1;
        x[2]=1;
        x[3]=1;
        x[4]=1;
        x[5]=1;
        x[6]=0;
    }
        else if(m<800)
    {
```



```

x[0]=1;
x[1]=1;
x[2]=0;
x[3]=1;
x[4]=1;
x[5]=0;
x[6]=0;
}
    for(l=sosi_input;l<sosisuu;l++)
{
    random_number=(int)(sprng()*2.0);
    if(random_number==0)
        {
            x[l]=1;
        }
    else
        {
            x[l]=0;
        }
}
    for(T=TEMPERATURE;T>T_END;T=T*0.8)
{
    for(issuu=0;issuu<sosi_hidden+sosi_output;issuu++)
        {
            k=(int)(sprng()*(sosi_hidden+sosi_output));
            i=sosi_input+k;
            y=0.0;
            for(j=0;j<sosisuu;j++)
        {
            y=y+w[i][j]*x[j];
        }

            iki=ikichi[i];
            s=sgn(y,T,iki);
            r=sprng();
            if(r<s)

```

```

{
    x[i]=1;
}
    else
{
    x[i]=0;
}
}
    if(m<100)
{
    if(x[sosisuu-3]==1&&x[sosisuu-2]==0 &&x[sosisuu-1]==1)
    {
        n1=n1+1;
    }
}
    else if(m<200)
{
    if(x[sosisuu-3]==0&&x[sosisuu-2]==1 &&x[sosisuu-1]==0)
    {
        n2=n2+1;
    }
}
    else if(m<300)
{
    if(x[sosisuu-3]==0&&x[sosisuu-2]==1 &&x[sosisuu-1]==1)
    {
        n3=n3+1;
    }
}
    else if(m<400)
{
    if(x[sosisuu-3]==1&&x[sosisuu-2]==0 &&x[sosisuu-1]==0)
    {
        n4=n4+1;
    }
}

```

```

    }
}
    else if(m<500)
{
    if(x[sosisuu-3]==0&&x[sosisuu-2]==0 &&x[sosisuu-1]==0)
    {
        n5=n5+1;
    }
}
    else if(m<600)
{
    if(x[sosisuu-3]==0&&x[sosisuu-2]==0 &&x[sosisuu-1]==1)
    {
        n6=n6+1;
    }
}
    else if(m<700)
{
    if(x[sosisuu-3]==1&&x[sosisuu-2]==1 &&x[sosisuu-1]==1)
    {
        n7=n7+1;
    }
}
    else if(m<800)
{
    if(x[sosisuu-3]==1&&x[sosisuu-2]==1 &&x[sosisuu-1]==0)
    {
        n8=n8+1;
    }
}
}
    if(n1>90&&n2>90&&n3>90&&n4>90&&n5>90&&n6>90&&n7>90&&n8>90)
    {
        kannsei=kannsei+1;
    }
}

```

```

else
  {
    kannsei=0;
  }
}
for(learn=0;learn<200;learn++)
  {
    for(i=0;i<sosisuu;i++)
{
  for(j=0;j<i;j++)
    {
      wp[i][j]=0;
      wp[j][i]=0;
      wn[i][j]=0;
      wn[j][i]=0;
    }
}
    for(o=0;o<80;o++)
{
  random=(int)(o/10);
  if(random==0)
    {
      x[0]=1;
      x[1]=0;
      x[2]=1;
      x[3]=1;
      x[4]=1;
      x[5]=1;
      x[6]=0;
      x[sosisuu-3]=1;
      x[sosisuu-2]=0;
      x[sosisuu-1]=1;
    }
  else if(random==1)
    {

```

```

    x[0]=0;
    x[1]=1;
    x[2]=0;
    x[3]=0;
    x[4]=1;
    x[5]=0;
    x[6]=1;
    x[sosisuu-3]=0;
    x[sosisuu-2]=1;
    x[sosisuu-1]=0;
}
else if(random==2)
{
    x[0]=0;
    x[1]=1;
    x[2]=1;
    x[3]=1;
    x[4]=1;
    x[5]=1;
    x[6]=0;
    x[sosisuu-3]=0;
    x[sosisuu-2]=1;
    x[sosisuu-1]=1;
}
else if(random==3)
{
    x[0]=1;
    x[1]=0;
    x[2]=0;
    x[3]=1;
    x[4]=0;
    x[5]=1;
    x[6]=1;
    x[sosisuu-3]=1;
    x[sosisuu-2]=0;
}

```

```

        x[sosisuu-1]=0;
    }
else if(random==4)
    {
        x[0]=0;
        x[1]=0;
        x[2]=0;
        x[3]=0;
        x[4]=0;
        x[5]=0;
        x[6]=0;
        x[sosisuu-3]=0;
        x[sosisuu-2]=0;
        x[sosisuu-1]=0;
    }
else if(random==5)
    {
        x[0]=0;
        x[1]=0;
        x[2]=1;
        x[3]=1;
        x[4]=0;
        x[5]=1;
        x[6]=1;
        x[sosisuu-3]=0;
        x[sosisuu-2]=0;
        x[sosisuu-1]=1;
    }
else if(random==6)
    {
        x[0]=1;
        x[1]=1;
        x[2]=1;
        x[3]=1;
        x[4]=1;

```

```

        x[5]=1;
        x[6]=0;
        x[sosisuu-3]=1;
        x[sosisuu-2]=1;
        x[sosisuu-1]=1;
    }
else if(random==7)
    {
        x[0]=1;
        x[1]=1;
        x[2]=0;
        x[3]=1;
        x[4]=1;
        x[5]=0;
        x[6]=0;
        x[sosisuu-3]=1;
        x[sosisuu-2]=1;
        x[sosisuu-1]=0;
    }
for(l=sosi_input;l<sosi_input+sosi_hidden;l++)
    {
        random_number=(int)(2.0*sprng());
        if(random_number==0)
    {
        x[l]=1;
    }
        else
    {
        x[l]=0;
    }
    }
for(T=TEMPERATURE;T>T_END;T=T*0.8)
    {
        for(issuu=0;issuu<(sosi_hidden)*10/T;issuu++)
    {

```

```

k=(int)(sprng()*sosi_hidden);
i=k+sosi_input;
y=0.0;
for(j=0;j<sosisuu;j++)
    {
        y=y+w[i][j]*x[j];
    }
s=sgn(y,T);
r=sprng();
if(r<s)
    {
        x[i]=1;
    }
else
    {
        x[i]=0;
    }
}
}
for(update=0;update<NUM_UPDATE;update++)
    {
        for(issuu=0;issuu<2*sosi_hidden;issuu++)
{
k=(int)(sprng()*sosi_hidden);
i=k+sosi_input;
y=0.0;
for(j=0;j<sosisuu;j++)
    {
        y=y+w[i][j]*x[j];
    }
s=sgn(y,T);
r=sprng();
if(r<s)
    {
        x[i]=1;

```



```

    }
else
    {
        x[i]=0;
    }
}
    for(i=0;i<sosisuu;i++)
{
    for(j=0;j<i;j++)
    {
        if(x[i]==1 && x[j]==1 && a[i][j]!=0.0)
        {
            wp[i][j]=wp[i][j]+1;
            wp[j][i]=wp[j][i];
        }
    }
}

    for(o=0;o<80;o++)
{
    random=(int)(sprng()*8.0);
    if(random==0)
    {
        x[0]=1;
        x[1]=0;
        x[2]=1;
        x[3]=1;
        x[4]=1;
        x[5]=1;
        x[6]=0;
    }
    else if(random==1)
    {

```

```
        x[0]=0;
        x[1]=1;
        x[2]=0;
        x[3]=0;
        x[4]=1;
        x[5]=0;
        x[6]=1;
    }
else if(random==2)
    {
        x[0]=0;
        x[1]=1;
        x[2]=1;
        x[3]=1;
        x[4]=1;
        x[5]=1;
        x[6]=0;
    }
else if(random==3)
    {
        x[0]=1;
        x[1]=0;
        x[2]=0;
        x[3]=1;
        x[4]=0;
        x[5]=1;
        x[6]=1;
    }
else if(random==4)
    {
        x[0]=0;
        x[1]=0;
        x[2]=0;
        x[3]=0;
        x[4]=0;
```

```

        x[5]=0;
        x[6]=0;
    }
else if(random==5)
    {
        x[0]=0;
        x[1]=0;
        x[2]=1;
        x[3]=1;
        x[4]=0;
        x[5]=1;
        x[6]=1;
    }
else if(random==6)
    {
        x[0]=1;
        x[1]=1;
        x[2]=1;
        x[3]=1;
        x[4]=1;
        x[5]=1;
        x[6]=0;
    }
else if(random==7)
    {
        x[0]=1;
        x[1]=1;
        x[2]=0;
        x[3]=1;
        x[4]=1;
        x[5]=0;
        x[6]=0;
    }
for(l=sosi_input;l<sosisuu;l++)
    {

```

```

        random_number=(int)(sprng()*2.0);
        if(random_number==0)
{
    x[l]=1;
}
        else
{
    x[l]=0;
}
    }
    for(T=TEMPERATURE;T>T_END;T=T*0.8)
    {
        for(issuu=0;issuu<(sosi_hidden+sosi_output)*10/T;issuu++)
{
    k=(int)(sprng()*(sosi_hidden+sosi_output));
    i=k+sosi_input;
    y=0.0;
    for(j=0;j<sosisuu;j++)
    {
        y=y+w[i][j]*x[j];
    }
    s=sgn(y,T);
    r=sprng();
    if(r<s)
    {
        x[i]=1;
    }
    else
    {
        x[i]=0;
    }
}
    }
    for(update=0;update<NUM_UPDATE;update++)
    {

```

```

        for(isshuu=0;isshuu<2*(sosi_hidden+sosi_output);isshuu++)
    {
        k=(int)(sprng()*(sosi_hidden+sosi_output));
        i=k+sosi_input;
        y=0.0;
        for(j=0;j<sosisuu;j++)
            {
                y=y+w[i][j]*x[j];
            }

        s=sgn(y,T);
        r=sprng();
        if(r<s)
            {
                x[i]=1;
            }
        else
            {
                x[i]=0;
            }
    }

        for(i=0;i<sosisuu;i++)
    {
        for(j=0;j<i;j++)
            {
                if(x[i]==1 && x[j]==1 && a[i][j]!=0.0)
            {
                wn[i][j]=wn[i][j]+1;
                wn[j][i]=wn[j][i];
            }
            }
    }

        for(i=0;i<sosisuu;i++)

```

```

{
  for(j=0;j<i;j++)
    {
      w[i][j]=w[i][j]+(epsilon/T_END*(wp[i][j]-wn[i][j]))/NUM_UPDATE/o;
      w[j][i]=w[i][j];
    }
}

cpgopen("over_turn_step#.gif/GIF");
q1=0,q2=0,q3=0,q4=0,q5=0;q6=0;q7=0;q8=0;
kannsei=0;
for(learn=0;kannsei<3;learn++)
  {
    for(i=0;i<sisuu;i++)
  {
    for(j=0;j<i;j++)
      {
        wp[i][j]=0;
        wp[j][i]=0;
        wn[i][j]=0;
        wn[j][i]=0;
      }
  }
    for(o=0;o<80;o++)
  {
    random=(int)(o/10);
    if(random==0)
      {
        x[0]=1;
        x[1]=0;
        x[2]=1;
        x[3]=1;
        x[4]=1;
        x[5]=1;
      }
  }
}

```

```

        x[6]=0;
        x[sosisuu-3]=0;
        x[sosisuu-2]=1;
        x[sosisuu-1]=0;
    }
else if(random==1)
    {
        x[0]=0;
        x[1]=1;
        x[2]=0;
        x[3]=0;
        x[4]=1;
        x[5]=0;
        x[6]=1;
        x[sosisuu-3]=1;
        x[sosisuu-2]=0;
        x[sosisuu-1]=1;
    }
else if(random==2)
    {
        x[0]=0;
        x[1]=1;
        x[2]=1;
        x[3]=1;
        x[4]=1;
        x[5]=1;
        x[6]=0;
        x[sosisuu-3]=0;
        x[sosisuu-2]=1;
        x[sosisuu-1]=1;
    }
else if(random==3)
    {
        x[0]=1;
        x[1]=0;

```

```

        x[2]=0;
        x[3]=1;
        x[4]=0;
        x[5]=1;
        x[6]=1;
        x[sosisuu-3]=1;
        x[sosisuu-2]=0;
        x[sosisuu-1]=0;
    }
else if(random==4)
    {
        x[0]=0;
        x[1]=0;
        x[2]=0;
        x[3]=0;
        x[4]=0;
        x[5]=0;
        x[6]=0;
        x[sosisuu-3]=0;
        x[sosisuu-2]=0;
        x[sosisuu-1]=0;
    }
else if(random==5)
    {
        x[0]=0;
        x[1]=0;
        x[2]=1;
        x[3]=1;
        x[4]=0;
        x[5]=1;
        x[6]=1;
        x[sosisuu-3]=0;
        x[sosisuu-2]=0;
        x[sosisuu-1]=1;
    }

```



```

else if(random==6)
{
    x[0]=1;
    x[1]=1;
    x[2]=1;
    x[3]=1;
    x[4]=1;
    x[5]=1;
    x[6]=0;
    x[sosisuu-3]=1;
    x[sosisuu-2]=1;
    x[sosisuu-1]=1;
}
else if(random==7)
{
    x[0]=1;
    x[1]=1;
    x[2]=0;
    x[3]=1;
    x[4]=1;
    x[5]=0;
    x[6]=0;
    x[sosisuu-3]=1;
    x[sosisuu-2]=1;
    x[sosisuu-1]=0;
}
for(l=sosi_input;l<sosi_input+sosi_hidden;l++)
{
    random_number=(int)(2.0*sprng());
    if(random_number==0)
{
    x[1]=1;
}
    else
{

```

```

    x[l]=0;
}
    }
for(T=TEMPERATURE;T>T_END;T=T*0.8)
    {
        for(issuu=0;issuu<(sosi_hidden)*10/T;issuu++)
{
    k=(int)(sprng()*sosi_hidden);
    i=k+sosi_input;
    y=0.0;
    for(j=0;j<sosisuu;j++)
        {
            y=y+w[i][j]*x[j];
        }
    s=sgn(y,T);
    r=sprng();
    if(r<s)
        {
            x[i]=1;
        }
    else
        {
            x[i]=0;
        }
}
    }
for(update=0;update<NUM_UPDATE;update++)
    {
        for(issuu=0;issuu<2*sosi_hidden;issuu++)
{
    k=(int)(sprng()*sosi_hidden);
    i=k+sosi_input;
    y=0.0;
    for(j=0;j<sosisuu;j++)
        {

```

```

        y=y+w[i][j]*x[j];
    }
s=sgn(y,T);
r=sprng();
if(r<s)
    {
        x[i]=1;
    }
else
    {
        x[i]=0;
    }
}
    for(i=0;i<sosisuu;i++)
{
    for(j=0;j<i;j++)
        {
            if(x[i]==1 && x[j]==1 && a[i][j]!=0.0)
        {
            wp[i][j]=wp[i][j]+1;
            wp[j][i]=wp[j][i];
        }
        }
}
}
}
    for(o=0;o<80;o++)
{
    random=(int)(sprng()*8.0);
    if(random==0)
        {
            x[0]=1;
            x[1]=0;
            x[2]=1;
            x[3]=1;

```

```
        x[4]=1;
        x[5]=1;
        x[6]=0;
    }
else if(random==1)
    {
        x[0]=0;
        x[1]=1;
        x[2]=0;
        x[3]=0;
        x[4]=1;
        x[5]=0;
        x[6]=1;
    }
else if(random==2)
    {
        x[0]=0;
        x[1]=1;
        x[2]=1;
        x[3]=1;
        x[4]=1;
        x[5]=1;
        x[6]=0;
    }
else if(random==3)
    {
        x[0]=1;
        x[1]=0;
        x[2]=0;
        x[3]=1;
        x[4]=0;
        x[5]=1;
        x[6]=1;
    }
else if(random==4)
```

```
{
    x[0]=0;
    x[1]=0;
    x[2]=0;
    x[3]=0;
    x[4]=0;
    x[5]=0;
    x[6]=0;
}
else if(random==5)
{
    x[0]=0;
    x[1]=0;
    x[2]=1;
    x[3]=1;
    x[4]=0;
    x[5]=1;
    x[6]=1;
}
else if(random==6)
{
    x[0]=1;
    x[1]=1;
    x[2]=1;
    x[3]=1;
    x[4]=1;
    x[5]=1;
    x[6]=0;
}
else if(random==7)
{
    x[0]=1;
    x[1]=1;
    x[2]=0;
    x[3]=1;
```

```

        x[4]=1;
        x[5]=0;
        x[6]=0;
    }
    for(l=sosi_input;l<sosisuu;l++)
    {
        random_number=(int)(sprng()*2.0);
        if(random_number==0)
    {
        x[l]=1;
    }
        else
    {
        x[l]=0;
    }
    }
    for(T=TEMPERATURE;T>T_END;T=T*0.8)
    {
        for(issuu=0;issuu<(sosi_hidden+sosi_output)*10/T;issuu++)
    {
        k=(int)(sprng()*(sosi_hidden+sosi_output));
        i=k+sosi_input;
        y=0.0;
        for(j=0;j<sosisuu;j++)
        {
            y=y+w[i][j]*x[j];
        }
        s=sgn(y,T);
        r=sprng();
        if(r<s)
        {
            x[i]=1;
        }
    }
    else
    {

```

```

        x[i]=0;
    }
}
    }
for(update=0;update<NUM_UPDATE;update++)
    {
        for(issuu=0;issuu<2*(sosi_hidden+sosi_output);issuu++)
    {
        k=(int)(sprng()*(sosi_hidden+sosi_output));
        i=k+sosi_input;
        y=0.0;
        for(j=0;j<sosisuu;j++)
            {
                y=y+w[i][j]*x[j];
            }

        s=sgn(y,T);
        r=sprng();
        if(r<s)
            {
                x[i]=1;
            }
        else
            {
                x[i]=0;
            }
    }
        for(i=0;i<sosisuu;i++)
    {
        for(j=0;j<i;j++)
            {
                if(x[i]==1 && x[j]==1 && a[i][j]!=0.0)
    {
        wn[i][j]=wn[i][j]+1;
        wn[j][i]=wn[j][i];
    }
            }
        }
    }

```

```

}
    }
}
    }
}
    for(i=0;i<sosisuu;i++)
{
    for(j=0;j<i;j++)
    {
        w[i][j]=w[i][j]+(epsilon/T_END*(wp[i][j]-wn[i][j]))/NUM_UPDATE/o;
        w[j][i]=w[i][j];
    }
}
    if(learn%3==0)
{
    for(color=0;color<=6;color++)
    {
        number[color]=0;
    }
    for(i=0;i<=sosisuu-1;i++)
    {
        for(j=0;j<i;j++)
        {
            if(w[i][j]<-0.1)
            {
                color=0;
            }
            else if(w[i][j]<-0.01)
            {
                color=1;
            }
            else if(w[i][j]<-0.001)
            {
                color=2;
            }
        }
    }
}

```



```

else if(w[i][j]<0.001)
    {
        color=3;
    }
else if(w[i][j]<00.1)
    {
        color=4;
    }
else if(w[i][j]<0.1)
    {
        color=5;
    }
else if(w[i][j]>=0.1)
    {
        color=6;
    }
mark_x[color][number[color]]=(float)i;
mark_y[color][number[color]]=(float)j;
number[color]++;
}
}
cpgsci(0);
cpgenv(0.0,(float)(sosisuu-1),0.0,(float)(sosisuu-1),0.0,0.0);
cpglab("", "", "cold -,hot +");
cpgsci(4);
cpgpt(number[0],mark_x[0],mark_y[0],18);
cpgsci(11);
cpgpt(number[1],mark_x[1],mark_y[1],18);
cpgsci(5);
cpgpt(number[2],mark_x[2],mark_y[2],18);
cpgsci(3);
cpgpt(number[3],mark_x[3],mark_y[3],18);
cpgsci(7);
cpgpt(number[4],mark_x[4],mark_y[4],18);
cpgsci(8);

```

```

    cpgpt(number[5],mark_x[5],mark_y[5],18);
    cpgsci(2);
    cpgpt(number[6],mark_x[6],mark_y[6],18);
    cpgpage();
}
    n1=0;
    n2=0;
    n3=0;
    n4=0;
    n5=0;
    n6=0;
    n7=0;
    n8=0;
    for(m=0;m<800;m++)
        {
            if(m<100)
{
    x[0]=1;
    x[1]=0;
    x[2]=1;
    x[3]=1;
    x[4]=1;
    x[5]=1;
    x[6]=0;
}
            else if(m<200)
{
    x[0]=0;
    x[1]=1;
    x[2]=0;
    x[3]=0;
    x[4]=1;
    x[5]=0;
    x[6]=1;
}
}

```

```
        else if(m<300)
    {
        x[0]=0;
        x[1]=1;
        x[2]=1;
        x[3]=1;
        x[4]=1;
        x[5]=1;
        x[6]=0;
    }
        else if(m<400)
    {
        x[0]=1;
        x[1]=0;
        x[2]=0;
        x[3]=1;
        x[4]=0;
        x[5]=1;
        x[6]=1;
    }
        else if(m<500)
    {
        x[0]=0;
        x[1]=0;
        x[2]=0;
        x[3]=0;
        x[4]=0;
        x[5]=0;
        x[6]=0;
    }
        else if(m<600)
    {
        x[0]=0;
        x[1]=0;
        x[2]=1;
```

```

    x[3]=1;
    x[4]=0;
    x[5]=1;
    x[6]=1;
}
    else if(m<700)
{
    x[0]=1;
    x[1]=1;
    x[2]=1;
    x[3]=1;
    x[4]=1;
    x[5]=1;
    x[6]=0;
}
    else if(m<800)
{
    x[0]=1;
    x[1]=1;
    x[2]=0;
    x[3]=1;
    x[4]=1;
    x[5]=0;
    x[6]=0;
}
    for(l=sosi_input;l<sosisuu;l++)
{
    random_number=(int)(sprng()*2.0);
    if(random_number==0)
    {
        x[l]=1;
    }
    else
    {
        x[l]=0;
    }
}

```

```

    }
}
    for(T=TEMPERATURE;T>T_END;T=T*0.8)
{
    for(issuu=0;issuu<sosi_hidden+sosi_output;issuu++)
    {
        k=(int)(sprng()*(sosi_hidden+sosi_output));
        i=sosi_input+k;
        y=0.0;
        for(j=0;j<sosisuu;j++)
    {
        y=y+w[i][j]*x[j];
    }
        iki=ikichi[i];
        s=sgn(y,T,iki);
        r=sprng();
        if(r<s)
    {
        x[i]=1;
    }
        else
    {
        x[i]=0;
    }
    }
}
    if(m<100)
{
    if(x[sosisuu-3]==1&&x[sosisuu-2]==0 &&x[sosisuu-1]==1)
    {
        q1=q1+1;
    }
}
    else if(m<200)
{

```

```

if(x[sosisuu-3]==0&&x[sosisuu-2]==1 &&x[sosisuu-1]==0)
{
    q2=q2+1;
}
else if(m<300)
{
if(x[sosisuu-3]==0&&x[sosisuu-2]==1 &&x[sosisuu-1]==1)
{
    q3=q3+1;
}
else if(m<400)
{
if(x[sosisuu-3]==1&&x[sosisuu-2]==0 &&x[sosisuu-1]==0)
{
    q4=q4+1;
}
else if(m<500)
{
if(x[sosisuu-3]==0&&x[sosisuu-2]==0 &&x[sosisuu-1]==0)
{
    q5=q5+1;
}
else if(m<600)
{
if(x[sosisuu-3]==0&&x[sosisuu-2]==0 &&x[sosisuu-1]==1)
{
    q6=q6+1;
}
else if(m<700)
{

```

```

if(x[sosisuu-3]==1&&x[sosisuu-2]==1 &&x[sosisuu-1]==1)
{
    q7=q7+1;
}
else if(m<800)
{
if(x[sosisuu-3]==1&&x[sosisuu-2]==1 &&x[sosisuu-1]==0)
{
    q8=q8+1;
}
}
if(q1>90&&q2>90&&q3>90&&q4>90&&q5>90&&q6>90&&q7>90&&q8>90)
{
    kannsei=kannsei+1;
}
else
{
    kannsei=0;
}
}
cpgclos();
return 0;
}
float sgn(float y,float T)
{
    float ans;
    ans=1/(1+exp((-y)/T));
    return ans;
}

```