

1. はじめに

ニューラルネットの一種であるボルツマンマシンという回路は、二次元の対称性学習問題、統計的パターン認識、スピーチの認識など多くの問題に応用されている。今回の研究では、文字(アルファベット)の認識をボルツマンマシンで行う事が目的である。

2. ニューラルネット

ニューラルネットとは、140 億個もの神経細胞により構成されている人間の脳の特徴的な機能に着目してモデル化した回路である。1 つのユニットは、他の複数のユニットから送られてきた信号に重み付けをしたものの総和を閾値にかけ、その結果を関数(通常シグモイド関数やステップ関数)にかけて他のユニットへと出力する。

3. ボルツマンマシン

今回の研究に用いるボルツマンマシンはそのニューラルネットの一種であるが、以下のような特徴を持つ。

- 相互結合である
- 対称結合である ($w_{ij} = w_{ji}$)
- エネルギーを定義できる
- 動作が確率的である(ただし、温度パラメータ T が $T \rightarrow 0$ の極限の場合は、動作は決定的である)
- 学習能力を持つ

任意のユニット 2 つの値が同時に 1 である時、その 2 つのユニット間は活性化しているといい、エネルギーとはネットワーク内で活性化しているユニット間の重みの合計である。ボルツマンマシンのユニットの出力 S_i 、およびボルツマンマシンのエネルギー E は以下の通りである。

$$S_i = \begin{cases} 1 & \text{if 確率 } g(h_i) \\ -1 & \text{if 確率 } 1 - g(h_i) \end{cases}$$

$$h_i = \sum_j w_{ij} S_j, \quad g(h_i) = \frac{1}{1 + \exp\left(-\frac{2h_i}{T}\right)}$$

$$E = \sum_{i < j} w_{ij} S_i S_j$$

ボルツマンマシンを確率的に動作(これをボルツマンマシン動作という)させていくと、十分に時間が経過した頃には回路は平衡状態に達する。この平衡状態において各パターンが出現する確率分布をボルツマン分布という。エネルギー $E(k)$ を持つパターン k が出現する確率 $q_k(T)$ は以下の通りである。 N_0 は確率の総和を 1 にする為の正規化の定数である。

$$q_k(T) = \frac{1}{N_0} \exp\left(\frac{E(k)}{T}\right)$$

4. ボルツマンマシン学習

ボルツマンマシンにおいて、状態 $\{S_i\}$ の出現確率(つまりボルツマン分布)が与えられた確率分布(これを理想分布という)に等しくなるように重みを変えていく事を学習という。重みの更新は以下の規則に基づく。

$$w_{ij}^{new} = w_{ij}^{old} + \varepsilon (p_{ij}^{fix} - p_{ij}^{free})$$

は学習幅、 p_{ij}^{fix} 、 p_{ij}^{free} はそれぞれ入出力を固定した時、固定しなかった(自由にした)時のユニット i と j が同時に 1 となる期待値である。

p_{ij}^{fix} 、 p_{ij}^{free} はユニット i と j の値が同時に 1 となる統計をとるか又は現在の確率分布より決定する事が出来る。

学習の進み具合を測る尺度をカルバックの情報量という。カルバックの情報量の最小化は学習の収束を意味する。カルバックの情報量は、理想分布 Q とボルツマン分布 q を用いて、以下のように定義できる。

$$K = \sum Q \log \frac{Q}{q}$$

5. 学習アルゴリズム

手順 1...ネットワークの入出力ユニットを学習パターンの1つにクランプする.

手順 2...ボルツマンマシン動作させネットワークを平衡状態にする.

手順 3...平衡状態においてユニット i と j が同時に 1 を出力する回数を調べる. 又は現在の確率分布より p_{ij}^{fix} を求める.

手順 4...ネットワークの入出力ユニットを固定しないでボルツマンマシン動作させネットワークを平衡状態にする.

手順 5...平衡状態においてユニット i と j が同時に 1 を出力する回数を調べる. 又は現在の確率分布より p_{ij}^{free} を求める.

手順 6... $w_{ij}^{new} = w_{ij}^{old} - \varepsilon(p_{ij}^{fix} - p_{ij}^{free})$ に従って重みを更新する.

手順 7... $(p_{ij}^{fix} - p_{ij}^{free})$ が設定値以下になれば学習が収束したと判断して終了. そうでなければ手順 1~6 を繰り返す.

6. 実験

25 のユニットを用いて 5×5 マスでアルファベットを描き,それを学習パターンとする.学習後に,任意の学習パターンにゴミを混ぜ,そのゴミ付きデータが何のアルファベットであるかを調べる.

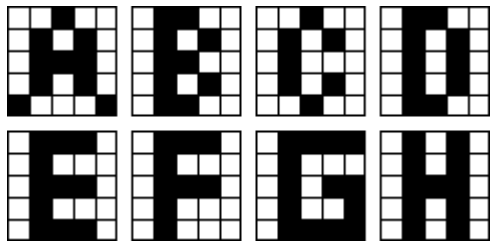


図 1 アルファベットデータの一例

アルファベット 26 文字を学習パターンとして学習を行った.上記のアルゴリズムの手順 1~6 を 25 万回繰り返して得られた重みを用い,学習パターンのいずれかにゴミを与えたデー

タを入力して,その入力からゴミを与える前のデータと識別するかを調べる.

7. 実験結果

データにゴミを与え,それを入力した回路を平衡状態へ持っていった後に回路がデータを正しく識別しているかを判定する.これを 100 回繰り返し,正しく識別した回数を調べた.その結果は以下のようになった.

表 1 入力データを正しく識別した回数

入力	識別回数	入力	識別回数
A	46	B	13
C	27	D	5
E	1	F	7
G	39	H	3
I	40	J	83
K	71	L	8
M	13	N	6
O	63	P	12
Q	94	R	2
S	5	T	63
U	6	V	82
W	83	X	48
Y	81	Z	74

誤認識されるデータも少なからずあり,M と N では総認識回数そのものが少なかった.

8. 考察

O や Q 等の 5×5 マスのデータとして形が特徴的なものは正しく識別する回数が多かった.だが,この実験とは別に学習パターンを 4 つに絞り 70 万回繰り返したものでも実験を行ったものでは認識率は 95%以上だったので,今回の実験のものは性能は高くないといえる.

9. 参考文献

J.Hertz,A.Krogh,R.G.Palmer :

INTRODUCTION TO

THE THEORY OF NEURAK

COMPUTATION(Springer)(1991,P163-187)

伊藤大介:ボルツマンマシン学習の高速化(山梨大学修士論文,2004)