

# ニューラルネット 中間レポート

学籍番号 035743A : 比嘉 雅樹

## 1 サンプルプログラムを用い Ex-or の学習を行う

### 1.1 逐次修正法の特徴を指摘せよ

入出力パターンが与えられるごとに総合荷重を修正する方法。入力パターンをもとに各ユニットの誤差を計算し、その誤差をもとに、修正量を計算し結合荷重を修正する。

### 1.2 逐次修正法 + 慣性項の特徴を指摘せよ

慣性項を用いる事で逐次修正法の荷重計算の際に1ステップ前の修正量を考慮する事ができる。これにより、始めは結合荷重を大きく修正し、誤差の変化に応じて徐々に修正量を減らしていく事で収束性を向上させる。

### 1.3 一括修正法

逐次修正法が入力毎に結合荷重の修正を行うのに対し、全ての入出力パターンが与えられた後で一括して修正する方法。修正は、これまでに蓄積された結合荷重の修正量をもとに行われる。

## 2 隠れ層の unit 数を増やし、その影響を調べよ

### 2.1 逐次修正法

- HIDDEN = 2 : iteration = 10000, error = 0.03426
- HIDDEN = 5 : iteration = 1287, error = 0.00001
- HIDDEN = 10 : iteration = 1011, error = 0.00001

実行結果を見てみると、隠れ層の unit 数を増やすと学習速度が速くなり、誤差も少なくなっている事が分かる。

## 2.2 逐次修正法 + 慣性項

- HIDDEN = 2 : iteration = 614, error = 0.00009
- HIDDEN = 5 : iteration = 89, error = 0.00010
- HIDDEN = 10 : iteration = 72, error = 0.00010

実行結果を見ると、隠れ層の unit 数を減らすと学習速度は速くなるが、誤差はほとんど変化していない事が分かる。

## 2.3 一括修正法

- HIDDEN = 2 : iteration = 30000, error = 0.00016
- HIDDEN = 5 : iteration = 3186, error = 0.00000
- HIDDEN = 10 : iteration = 3084, error = 0.00000

実行結果を見ると、隠れ層の unit 数を増やすと学習速度が速くなる。また、誤差が無くなった。

## 3 オリジナルの問題を設計し、問題設計の意図、入出力データ、学習に用いるニューラルネットの構造について議論せよ。

- 問題設計と入出力データ と  $\times$  を判別する。  
3x3 の配列を用意し、図のようにして と  $\times$  を判別させる。

			$\times$		
1	1	1	1	0	1
1	0	1	0	1	0
1	1	1	1	0	1

- 問題設計の意図  
ただ入力された値を EX-or するだけでなく、配列を利用して形を判別する事ができないか考えた。

- 実行結果

変更したプログラムを「逐次修正法 + 慣性項」を用いて実行した。

以下に実行結果を示す。

```
ctg[0] : i[0] = 0.9 i[1] = 0.9 i[2] = 0.9 i[3] = 0.9 i[4] = 0.1  
i[5] = 0.9 i[6] = 0.9 i[7] = 0.9 i[8] = 0.9 o[0] = 0.90174, t{0} = 0.9  
ctg[1] : i[0] = 0.9 i[1] = 0.1 i[2] = 0.9 i[3] = 0.1 i[4] = 0.9  
i[5] = 0.1 i[6] = 0.9 i[7] = 0.1 i[8] = 0.9 o[0] = 0.91527, t{0} = 0.9  
iteration = 114, error = 0.00008
```

実行結果を見ると、きちんと学習できていることが分かる。