ニューラルネットワークによる連立一次方程式の解法

2010SE097 河合佑太
指導教員：小藤俊幸

1 はじめに

ニューラルネットワーク [1] という, 脳機能に見られるいくつかの特性を計算機上のシミュレーションによって表現することを目指した数学モデルに興味を持った. その中でも連立一次方程式に着目して本論文の研究内容にする.

2 ニューラルネットワークの基礎

ニューラルネットワーク [2] には, 階層型ニューラルネットワークと相互結合型ニューラルネットワークがある.

階層型ニューラルネットワークは, それぞれの層の役割がはっきりしていること, また層内でのユニット間での結合がないという特徴がある.

それに対して相互結合型ニューラルネットワークは, 階層も明確な役割分担もなく, ただすべてのユニットが平等であり, また一つ一つのユニットが他の全てユニットから値を得て, 他の全てのユニットへ値を渡している.

今回の研究では相互結合型ニューラルネットワークを扱う.

3 相互結合型ニューラルネットワーク

相互結合型ニューラルネットワークは相互結合型という通り, 各ユニットが他の全てのユニットと相互に結合している. よって, そこには階層という考え方や入力部や出力部のような役割分担もない. また, 全てのユニットは平等に他のユニットの最新状態を知り, それをもとに自己の動作を決めることができる.

前述の通り, 相互結合型ニューラルネットワークの一つ一つのユニットは, 他の全てのユニットの出力をもらっており, 自分の出力を他の全てのユニットに渡している. そして, このような動きが非同期に各ユニットで自律的に行われている. 相互結合型ネットワークでは, 絶えずこのような状態変化が繰り返されている. しかし, いつか落ち着いてある一つの定常状態になるか, いくつかの状態を繰り返すループになる.

またここでいう状態とは, 各ユニットの出力値を並べたベクトルを指す. どのような状態に落ち着くかは, ユニット間の結合荷重と最初にネットワークに与えた状態によって決まる. そしてこの結合荷重を決定する方法は, 達成する目的によってあらかじめ与えることができる.

しかし, ネットワークは結合したからといってうまく動くわけでない. 例えば, 階層型ニューラルネットワークはバックプロパゲーションという学習方法が存在し, ネットワークの動きを希望通りにすることができる. つまり, 階層型ニューラルネットワークが後天的に機能を獲得するのに対し, 相互結合型ニューラルネットワークは, あらかじめ決められた結合荷重により連想能力が決定されており, 先天的機能を持っている. また, 相互結合型ニューラルネットワークは, 学習により任意の状態を記憶することもできる. だから相互結合型ニューラルネットワークは, その状態の持つエネルギーを最小化することで最適化問題を解ける.

4 連立一次方程式の解法

まず, 相互結合型ニューラルネットワークを使って最適化問題を解くには, ユニットの状態に問題の何を表現させるかという点とエネルギー関数の役割を知る必要がある. まずエネルギー関数について考える.

$$E = -\frac{1}{2}\sum_{ij} W_{ij}X_iX_j - \sum_i h_iX_i \tag{1}$$

この式で X_i はユニット i の出力値であり,W_{ij} はユニット j からユニット i への結合荷重である. また W_{ij} は,i,j について対称であり $W_{ij} = W_{ji}$ となる.h_i はユニット i のしきい値である.

そしてネットワークが状態変化を繰り返すと (1) のエネルギー E が減少し, 極小値に収束する. また, 初期状態の選び方で極小値が最小値と一致する.
また (1) はユニットの状態 X_i の二次式になっている. よって, 相互結合型ニューラルネットワークで最適化問題を解くには, 問題の特徴を表現する量の二乗を最小にすることで求めることができる.

では連立一次方程式の解法を考えてみる. 簡単にするため未知数を二つにする.

$$\begin{cases} a_{11}x_1 + a_{12}x_2 = b_1 \\ a_{21}x_1 + a_{22}x_2 = b_2 \end{cases} \tag{2}$$

これをベクトルを使って表現する.

$$Ax = B \tag{3}$$

また $x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$, $B = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}$, $A = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}$ とする.

(3) において,$Ax - B$ となるように B を移項すると, その成分が 0 となる場合, つまり (4) が最小となる場合,x は求める解となる.

$$\phi = |Ax - B|^2 \tag{4}$$

(4) の右辺はベクトルのノルムの二乗であり, これは (1) が求めている二次式である.ϕ は目的関数である. 次にユニットの状態を使って, 数値を表現する.

x が x_1 と x_2 を成分とするベクトルなので (5) で表せる.

$$x_1 = X_{11} + X_{12} + X_{13} + X_{14} + X_{15}$$
$$x_2 = X_{21} + X_{22} + X_{23} + X_{24} + X_{25} \tag{5}$$

これをまとめると,

$$x_j = \sum_m X_{jm}(j = 1, 2; m = 1, 2, 3, 4, 5) \quad (6)$$

となる.

次に (1) のユニットの状態を表す変数 X_i は, 添字が i のみだが,(6) のユニットの表現は,X_{jm} であり添字が二つなので, 次の式をエネルギー関数とする.

$$E = -\frac{1}{2}\sum_{jm}\sum_{j'm'} W_{jm,j'm'}X_{jm}Xj'm' - \sum_{jm} h_{jm}X_{jm} \quad (7)$$

次に, 目的関数 ϕ をユニットの状態を使って表現する.

$$\begin{aligned}\phi &= |Ax - B|^2 \\ &= \sum_i b_i^2 - 2\sum_i\sum_{jm} b_i a_{ij} X_{jm} \\ &\quad + \sum_i(\sum_{jm} a_{ij}X_{jm})^2 \end{aligned} \quad (8)$$

ここで,(8) と (7) を比べてみる.(8) の第一項は, 定数項なので無視する.(8) の第二項はユニットの状態についての一次項なので,(7) と係数比較法で一致させ,(7) でのしきい値 h_{jm} を決めることができる.

$$h_{jm} = 2\sum_i b_i a_{ij} \quad (9)$$

次に二次の項についても同様に決める.(8) の第三項は

$$\begin{aligned}&\sum_i(\sum_{jm} a_{ij}X_{jm})^2 \\ &= \sum_i\sum_{jm}\sum_{j'm'} a_{ij}a_{ij'}X_{jm}X_{j'm'}\end{aligned} \quad (10)$$

となり,(7) の第一項と比較すると, 結合荷重 $W_{jmj'm'}$ を決められる.

$$W_{jm,j'm'} = -2\sum_i a_{ij}a_{ij'} \quad (11)$$

$a_{ij}, a_{ij'}, b_i$ は全て,(2) を与えることによって決まる定数なので,(9)(11) によりユニット jm のしきい値とユニット $j'm'$ から, ユニット jm への結合荷重が計算できる. つまりネットワークの状態変化を繰り返したエネルギーが最小になり, 目的関数も最小になるため, 解を出したことになる. 次に状態変化は次の式を使って行う.

$$u_{jm} = \sum_{j'm'} W_{jm,j'm'}X_{j'm'} + h_{jm} \quad (12)$$

$$X_{jm} = \frac{1}{2}(1 - \tanh(\frac{u_{jm}}{0.5})) \quad (13)$$

5 結果

$$\begin{cases} 0.5x_1 + x_2 - x_3 = -1 \\ x_1 - 3x_2 + x_3 = 2 \\ 2_1 + 3x_2 - 2x_3 = 1 \end{cases} \quad (14)$$

以下が (14) の式を解くために実際に付録のプログラムを作成し, 実行したものである. 引数の二つは 1.0 とし, ユニットの初期状態は, その初期値として全てのユニットの出力値に 0 を与えた. ある一つのユニット X_{jm} の状態を変化させるために,(12) に (9)(11) の値を代入し求められた u_{jm} を (13) に代入する. これを全てのユニットに適用させたのが一サイクルである. 各サイクルの X1 と X2 と X3 が方程式の変数で, それぞれに五個のユニットが割り当てられているので, その値を並べてある. 値が 0.5 以上のユニットの数が, その変数の表す答えとなる. 今回は十サイクル目でネットワークのエネルギーが 0 となり,X1=2.0,X2=1.0,X3=3.0 の解を導いている.

Units Cycle : 1

vector	1	2	3	4	5
X1	1.00	0.00	0.00	0.00	0.00
X2	0.00	0.00	0.00	0.00	0.00
X3	1.00	0.00	0.00	0.00	0.00

Energy = 1.250000

Units Cycle : 2

vector	1	2	3	4	5
X1	1.00	1.00	0.00	0.00	0.00
X2	0.00	0.00	0.00	0.00	0.00
X3	1.00	1.00	0.00	0.00	0.00

Energy = 5.000000

中略

Units Cycle : 9

vector	1	2	3	4	5
X1	0.00	1.00	0.47	0.00	0.00
X2	1.00	0.00	0.00	0.00	0.00
X3	0.00	0.00	0.00	1.00	1.00

Energy = 4.250000

Units Cycle : 10

vector	1	2	3	4	5
X1	0.00	0.00	1.00	1.00	0.00
X2	0.00	1.00	0.00	0.00	0.00
X3	1.00	0.54	0.00	0.00	1.00

Energy = 0.000000

The answer is

X1 = 2.0 X2 = 1.0 X3 = 3.0

6 おわりに

今回の研究により, ニューラルネットワークを用いて連立一次方程式が解けることが分かった.

しかし, ユニットによる数の表現法によりどのような連立一次方程式を解けるわけではないので, 多変数でさらに数の表現範囲を広めることや, 解が整数に限られているので, 実数解も求められるようにすることなどが必要だと感じた.

参考文献

[1] http://ja.wikipedia.org/wiki/ニューラルネットワーク

[2] c++と Java でつくるニューラルネットワーク,2008 著者:平野廣美