

## ニューラルネットワークと遺伝的プログラミングによる ルール生成と債券格付分析への応用

01014343 信州大学 池田 欽一 IKEDA Yoshikazu  
01304556 九州大学 \*時永 祥三 TOKINAGA Shozo  
九州大学 呂 建軍 LU Jianjung

### 1 まえがき

ニューラルネットワークによる非線形関数の近似と、言語的な解釈を同時に利用するシステムにおいて、より適切な言語的な表現を得るため、遺伝的プログラミングの方法により最適化する。

### 2 ニューラルネットワークとルール抽出

ニューラルネットワークは、ユニットへの入力信号  $x_i$  に対して重み係数  $w_i$  をかけたものを加え  $u_i$  とし、更に、これに対してしきい値演算をほどこしている。しきい値演算とは、加算された結果  $u_i$  がある値  $h_i$  より大きければ出力を出し、そうでない場合には出力をゼロとする演算である。このような演算結果は、更に、次の段階にあるユニットへの入力として用いられる。ある一定の数値を越えるまでは出力はゼロであり、これ以上になると一定の数値を出力としてもつ性質をもつ(関数をシグモイド(sigmoid)関数とよんでいる)。

階層型ニューラルネットワークにおいては、入力端子への入力に応じて出力端子に望ましい出力が得られるように設計することが目的となる。例えば、企業の財務指標を入力として与えた場合に、その評価値が出力として示されるように設計をするので、評価を自動化するシステムとして用いることができる。

以上のようにして準備された学習データを用いてニューラルネットワークを調整する手順(重みの最適化)は、逆伝播法(back propagation algorithm)として整理されている。出力層  $N$  のユニット  $i$  から戻される学習信号は、出力層からの出力値  $x_i^N$  と教師信号  $d_i^N$  との差を使って行なわれる。なお、この方法についての文献は多いので詳細は省略する。

#### ニューラルネットワーク学習と離散化手法

ニューラルネットワークによる学習結果から言語的なルールを抽出する方法として、よく知られている Neurorule の方法の概要は、次のようにまとめられる [1]。なお、ニューラルネットワークとしては3層構造の階層型ニューラルネットワークであり、中間層が1つのものを用い、カテゴリごとに出力信号をユニットとして分離している。

##### (ステップ1)

ニューラルネットワークに対する入力と出力のペアを用いて、通常の逆伝播法を用いて学習重みの最適化を行う。

##### (ステップ2)

ユニット間の重みを持つ結合(枝)を適宜切断する。具体的には、重みが小さい枝を無くする。しかし、その選択においては、もともとのニューラルネットワークの推定精度の極端な低下が起こらない範囲とする。

##### (ステップ3)

このように形成され、簡単化されたニューラルネットワ

クに対して、中間層の信号レベルを離散化する。

##### (ステップ4)

ネットワークの出力を、このような中間層の離散化された信号レベルを用いて記述する。この場合に、得られる記述をルールとする。

##### (ステップ5)

更に、中間層の離散化された信号レベルを、もともとの入力信号により記述する。なお、この入力信号も2進符号化されており、離散化されていると仮定する。

##### (ステップ6)

以上の操作により得られるルールの2つを併合して、1つのルールを求める手順を繰り返す。これにより、入力変数と出力のカテゴリとの間の対応関係を示すルールの集合を得ることができる。

なお、以上の操作において入力信号は2進符号化されていることが仮定されている。例えば、個人貸付の場合には、貸し付け申請者の収入  $W$  を  $W > 3000, 2000 < W \leq 3000, 1000 < W \leq 2000, W \leq 1000$  のように4レベルに離散化し、これらの3ビットの符号化 ( $B_1, B_2, B_3$ ) を与え、それぞれを  $(1, 1, 1), (0, 1, 1), (0, 0, 1), (0, 0, 0)$  として表現しておく。

最も面倒なのが、第4番目のルールを生成する操作であろう。Quinlan らにより開発された従来の手法を拡張して、多重に用いることを行っており、極めて複雑な方法となっている [2]。従って、この部分が自動化できれば、ニューラルネットワークからの言語的ルールの抽出も、効率化されるであろう。このような理由から、本章では、このルール抽出の部分を GP により実施するような拡張を行う。具体的には、Neurorule におけるステップ4からステップ6までの操作を省略し、直接的に言語的ルールを GP により生成するものである [3]-[5]。このような中間段階を省略することにより、Neurorule 適用における調整などの操作が不要となる。

### 3 GP によるルール生成におけるニューラルネットワークの役割

以下の議論では、Neurorule で示されたルール生成の方法を簡単化するために GP を用いる。具体的には、入力信号を2進数に符号化し、その入力が意味のあるものであるかどうかを検定するまでは、Neurorule における処理と同じことを行なうが、言語的ルールを生成する部分を GP により置き換える。これにより、中間層における2進数化の操作や、この中間層データと入力との関係を計算する面倒な操作が不要となる。これを手順として記述すると、次のようになる。

(1) 入力変数を2進化し、これらのビット信号を入力変数としている。

(2) 重みの小さな枝を切断する。

(3) この結果として入力として採用される命題が求められる。  
 (4) 最後にこれらの命題を用いて GP によりルールを生成する。

抽出された命題の任意の組合せとして論理式を生成し、最も観測データを良く説明する論理式を、遺伝的操作により合成していく。以下では、算術式の例にとりながら、このアルゴリズムを説明する。

#### GP の概要

(ステップ 1): 個体の初期値生成

最初の個体の集合 (プール) を乱数をもとにして発生させる。この場合、すでに述べた *StackCount* を用いて、関数として意味をなすものが得られるまで繰り返す。

(ステップ 2): 個体の適応度の計算

すでに述べた関数近似における予測値と観測値との 2 乗誤差の逆数を、個体  $i$  の適合度  $S_i$  として定義する。個体を適応度の大きい順に並びかえておく。

(ステップ 3): 交差処理

適応度に応じて 2 つの個体を取り出し、交差処理を実施する。交差処理では、乱数を 1 個発生させておいて、一方の個体の切斷箇所とし、ここまでの *StackCount* の値を計算する。このうち、もう一方の個体の *StackCount* を計算しながら、同じ *StackCount* の値となる場所を検出し、これらから任意に 1 個を選択して切斷位置を決める。

(ステップ 4): 突然変異

GP における突然変異として、グローバル突然変異ではある個体 (木構造)  $I_i$  を生成し、目的とする木構造の葉の部分  $I_i$  により置き換えるローカル突然変異は木構造の葉の部分、別の変数に置き換える操作である。

#### 論理式の表現への拡張

ルールは算術式および比較演算子からなる命題と、その論理演算により構成される。条件部は 1 つの論理式として記述されており、複数の命題 (statement) を論理記号で結合したものとなっている。更に、1 つの命題は、2 つの算術式を比較演算子で結合した形となっている。

### 4 応用例

応用例として、最初に、日本の企業における倒産予測の問題をとりあげる。これに関しては、すでに、もともとのニューラルネットワーク (財務指標を数値変数として用いる方法) による推定と、その比較分析の結果を公表しているが、ここでは、やや単純化された例をとりあげる。戦後日本の企業倒産の事例を検索し、この倒産時期における財務指標を求めておく。このデータを作成するにあたり、もともとは 13 個の指標を計算しているが、少数が多いので因子分析を実施して、因子負荷量を見ながら、最終的に次に示す 5 つの指標にまで縮約している。

(1) 使用総資本事業利益率、(2) 固定資産回転率、(3) 売上高経常利益率、(4) 流動比率、(5) 負債比率

同時に、この企業倒産時期に倒産していない非倒産 (あるいは健全企業とよばれる) 企業を求めておき、財務指標を整理し、対応するサンプル (ペアサンプル) としておく。倒産企業と非倒産企業の数を、それぞれ、13 個とする。これらの同数のサンプルからなるデータを用いて、ニューラルネットワークの重みを最適化する (学習プロセス)。次に、それぞれの財務指標を 4 レベルに離散化し、これらの 2 進化された入力信号により学習を再度行う。これにより、ルール生成の準備が完了する。

表 1 には GP により生成された言語的ルールによる倒産

予測の結果を示している。また、この表に対応するように、多変量解析のパッケージを用いた判別分析、ニューラルネットワーク (NN) による解析結果についても、比較のために示す。

表 1 倒産予測の比較 (カッコ内は識別率%)

カテゴリ	GP ルール判別	NN 判別	多変量解析
非倒産	11(84%)	12(92%)	12(92%)
倒産	11(84%)	11(84%)	11(84%)

#### 格付け問題への応用

現在の日本企業の格付け問題を取りあげる。2000 年度における 89 社の決算データ (26 財務指標へ集約) と格付データを用いて、本報告の手法による格付の性能評価を行なう。すなわち、外的基準として格付の値 (A, B, C の 3 つに集約) を与え、これに一致するように GP を用いて学習をすすめ、最後に同じ学習データに対して求められた最も適合度の高いルールを適用する。その結果を外的基準と比較して、一致度を求める。

なお、用いる財務指標に関しては、当初の 26 指標は簡潔なルールを生成するには多過ぎるので、因子分析を行なって、主要な 5 指標だけを用いるように変更している。これら企業を、その格付けの区分ごとに整理している。

表 2 格付分析の結果 (識別率%)

格付	GP 判別	NN 判別	多変量解析
A	22(73%)	24(80%)	23(76%)
B	24(75%)	27(84%)	27(84%)
C	19(70%)	21(77%)	21(77%)

### 5 むすび

ニューラルネットワークにより構成された数値的な判別システムを GP の方法により、言語的解釈の部分を最適化する方法を示した。今後、言語的な解釈を与えることや、この単純化を進める予定である。

### 参考文献

- [1] B.Baesens, R.Setino, C.Mues and J.Vanthienen, "Using neural network rule extraction and decision tables for credit-risk evaluation", *Management Science*, vol.49, no.3, pp.312-329, 2003.
- [2] J.R.Quinlan, *C4.5 programming for Machine Learning*, Morgan Kaufmann, Chambery, France, 1993.
- [3] Y. Ikeda and S.Tokinaga, "Approximation of chaotic dynamics by using smaller number of data based upon the genetic programming", *Trans. IEICE*, vol.E83-A, no.8, pp.1599-1607, 2000.
- [4] Y. Ikeda and S.Tokinaga, "Controlling the chaotic dynamics by using approximated system equations obtained by the genetic programming *Trans. IEICE*, vol.E84-A, no.9, pp.2118-2127, 2001.
- [5] X. Chen and S.Tokinaga, "Multi-Agent-Based modeling of artificial stock markets by using the co-evolutionary GP approach", *JORSJ* to appear, 2004.