

ルール表現の拡張による遺伝的プログラミングを用いた 医療データの学習

新美 礼彦 田崎 栄一郎

桐蔭横浜大学工学部制御システム工学科

Genetic Programming using Extended Rule Expressions for Medical Data

Ayahiko Niimi Eiichiro Tazaki

Department of Control and Systems Engineering, Toin University of Yokohama

Abstract: It is easy for an unexpected decision tree to be generated in the decision tree construction with genetic programming because the probability operation is contained. In the description of the decision tree by normal genetic programming, the division conditions by the attribute are connected with AND operator, and the tree evaluates effectiveness as a rule. However, if the description of the function node is modified, a more flexible rule is sure to be able to be generated. In this paper, we show that the description of a more flexible decision tree is possible by the addition of the OR function and NOT function to the function node group.

Keywords: Genetic Programming, Logical Functions, Knowledge Expression, Data Mining, Medical Database

1. はじめに

遺伝的プログラミングを医療データへのデータマイニングに用いると、進化計算による確率的な操作により意外な知識を発見することが期待できる。遺伝的プログラミングでは、染色体表現に構造表現を用いることにより、使用できる知識表現が決定木からルールまで幅広く適用可能である。しかし、適応度関数により個体を評価する都合上、決定木のように知識全体をカバーできるような形式が主に利用されてきた。一般的な遺伝的プログラミングによる決定木の記述では、属性による分割条件をANDで接続して、ルールとして評価していく。しかし、遺伝的プログラミングでは、遺伝子表現に置き換えられ適応度関数が定義できれば実装可能である。これは、他の知識表現も遺伝的プログラミングに実装可能なことを示している。本論文では、医療データからのデータマイニングにおいて、ルール表現の拡張を提案する。通常使われる関数ノード群にOR関数やNOT関数を追加することにより、より柔軟な決定木の記述が可能であることを示す。また、追加した関数によるルールの解釈についても考察する。検討した決定木、ルール表現による学習の違いを検討するために、これらの関数ノードと自動関数定義を組み込んだ遺伝的プログラミングによる学習の統合を行った。これを髄膜脳炎データによる小規模な評価用のデータからのルール生成問題に適用し、従来の関数ノード定義による学習法による結果と比較・検討した。

2. 遺伝的プログラミング

遺伝的プログラミング(Genetic Programming:GP)は、生物進化論の考えに基づいた学習法であり、その

アルゴリズムの流れは遺伝的アルゴリズム(Genetic Algorithm:GA)と同様である。¹⁾その特徴は染色体表現がGAと異なり、関数ノードと終端ノードを用い構造表現ができるように拡張してあることである。GPでは、関数ノードと終端ノードを用いてLISPのS式形式で個体を表現する。今回は、分類ルールを表現するため、関数ノードに条件文、終端ノードをそれぞれの属性値とクラス名を用いてルールを表現した。また、本論文では、生成されるルールをコンパクトにするため、自動関数定義(Automatically Defined Function:ADF)を用いた。²⁾

3. ANDとORを用いたGP

すでに論理関数を用いたGPがいくつか提案されている。通常の論理関数を用いたGPでは、ブール概念の形成を行うマルチプレクサー問題を対象としていた。これらの実装と同様にAND,OR,NOTの戻値がTrueかFalseと定義すると、データマイニングや知識発見で使いにくい。そのため本論文では、これらの定義をそのまま利用していない。今回提案する論理関数を用いたGPは、データマイニングや知識発見に応用する目的で実装している。そのため、データベースからの属性値のような、連続値属性を含んだものへの応用が可能ないようにそれぞれの関数を定義した。³⁾NOTの定義では、属性反転の戻値が2値化されている。実装内部ではTrueとFalseを用いることにより(NOT A)属性値がA以外を表現するように定義した。ここでは、ORを含んだルールをGPによって表現しやすくするため、ANDとORによるルール表現を以下のような関数ノードとして定義する。

- ・ (AND A B C D) if (A and B) then C else D.
- ・ (OR A B C D) if (A or B) then C else D.

・ (NOT A) if (A = True) then False else True.

GPにおいて、多様なルールの表現法を実装するのは比較的容易である。ANDやOR、NOTなどの実装は、関数ノードの定義を変更するだけで行うことが可能である。関数ノードの定義を変えるだけなので、GPによる学習の枠組みを変える必要がない。したがって、場合によっては適応度関数やその他のパラメータに関しても、そのままのものが使える可能性がある。今回の変更でも、関数ノードの定義のみ変更でよく、適応度関数やパラメータを変更する必要がない。この定義ではif-elseを定義して使用するのに比べてANDやOR、NOTを含んだルールを表現しやすくなっているので、生成される決定木のサイズの縮小が期待される。しかし、定義する関数ノードが増えることにより組み合わせの増加が起こるため、学習速度に関しては、あまり改善を期待できない。一般的に進化計算では、表現方法の豊かさが必ずしも学習を改善させるとは限らない。重複した表現により組み合わせ爆発を引き起こし、進化計算での探索範囲を広げてしまうからである。決定木の中にすでにORやNOTを表現する部分が含まれているので、ORやNOTを関数ノードとして定義すると表現が重複することになる。そのため、必要最小限のノードを定義することが重要になるが、一般的には事前に決定するのは難しい。

4. 髄膜脳炎データによる検証実験

ルール表現の違いによるGPの学習の違いを検証するために、評価用データを用いた実験を行った。評価用データには、髄膜脳炎データベースを使用した。⁴⁾⁵⁾髄膜脳炎データベースは、データ数が140件と少ない。また、一部の記述属性に連属値を含んでいる。今回の実験では、連属値属性を平均値をもとに2値化して使用した。関数ノードに関しては、ANDのみ、AND+OR、AND+NOT、AND+OR+NOTの組み合わせで実験を行った。GPのパラメータは、すべての組み合わせで同じ設定を用いた。AND単体よりANDとORでルールを学習した方が、精度の高いルールを生成することができた。ANDとORの両方を使用する場合、定義する関数ノードが増えるので、組み合わせの増加が起きる。このため、学習が遅くなり、最良個体獲得までの世代数が長くなってしまったものと思われる。NOTを用いたものはNOTを用いなかったものに比べて、精度が向上した。ANDのみに比べて、AND+ORでは最良個体獲得までの世代数が長くなってしまっている。しかし、AND+OR+NOTではAND+ORに比べて最良個体獲得までの世代数が減少している。AND+ORと比較して考えると、定義ノード数の増加により、AND+OR+NOTでは探索しきれていない可能性がある。

5. おわりに

本論文では、遺伝的プログラミングによるルール表現を検討し、ANDとOR、NOTを用いた表現を遺伝的プログラミングに実装した。また、実装したルール表現の有効性を検証するために、髄膜脳炎データを用いてルール生成を行い、その評価を行った。その結果、従来手法ではルール表現が複雑になって

しまう表現が、提案手法では小規模なルールとして表現できることを確認した。実験結果より提案手法では、一つのルールに対して複数の記述ができる環境で、柔軟な決定木の記述が可能になる。これにより提案手法では、ルールの記述サイズを改善することが期待できる。今後は、他の検証用データを用いた評価を行うとともに、定義ノード数の増加による学習速度の低下を押さえる手法の検討をしていく予定である。また、NAND、NORやXORなどによるルール表現についても利用できるか検討を行い、どのルール表現を使用するかに関する指針についても検討していく予定である。

参考文献

- [1] J. R. Koza, Genetic Programming, MIT Press, 1992.
- [2] J. R. Koza, K. E. Kinner(ed.), et al, Scalable Learning in Genetic Programming Using Automatic Function Definition, Advances in Genetic Programming, pp. 99-117, 1994.
- [3] 新美礼彦, 田崎栄一郎, or演算子を含んだ関数ノード群を持つGPによる拡張決定木の生成, 第15回人工知能学会全国大会論文集, 2pages(in CD-ROM), 2001.
- [4] 新美礼彦, 田崎栄一郎, 決定木と遺伝的プログラミングの組み合わせによる知識構造発見手法, 第38回人工知能基礎論研究会&第45回知識ベースシステム研究会, 人工知能学会, pp.19-24, 1999.
- [5] 新美礼彦, 田崎栄一郎, 相関ルールアルゴリズムと遺伝的プログラミングの組み合わせによる医療データの学習, 第20回医療情報学連合大会論文集, pp. 584-585, 2000.