

クラスタ例からの学習 ——分類対象集合全体の属性の利用

神 嶋 敏 弘[†] 元 吉 文 男[†]

クラスタ例からの学習とは、ID3に代表される例からの学習にクラスタリングの要素を加えたものである。例からの学習では、分類対象とそれが分類されるべきクラスの組である学習事例から未知の分類対象をクラスに分類する規則を学習する。一方、クラスタ例からの学習では、分類対象集合とその集合の適切な分割の具体例の組を学習事例とし、未知の集合を適切に分割するための規則を学習する。従来のクラスタ例からの学習では、一対の分類対象の特徴という非常に局所的な要素のみに基づいて分割を試みていた。しかし、大域的な分類対象全体の属性も考慮して分割を推定すべき場合も多いので、このような特徴を考慮できるよう改良を加えた。この改良によって、より適切な分割が推定できることを実験的に示す。

Learning from Cluster Examples ——Considering Attributes Assigned to Entire Object Sets

TOSHIHIRO KAMISHIMA[†] and FUMIO MOTOYOSHI[†]

A *Learning from cluster examples* is a combined task of a popular learning from examples task, such as Quinlan's ID3, and a clustering. A goal of the learning from examples task is finding rules to identify proper class of given unseen objects, learning from a set of examples that are pairs of an object and class the object to belong. On the other hand, a goal of the learning from cluster examples task is finding rules to partition unseen object sets, learning from a set of examples that are pairs of an object set and an instance of its appropriate partition. The former learning method tried to derive appropriate partitions only grounded on attributes of object pairs. However, such attributes often doesn't have enough facility to express information for object sets to be appropriately partitioned. Therefore, we present a method that can consider global attributes of an entire object set for partitioning. We experimentally show that our new method can acquire the rules that derive more appropriate partition than the former one.

1. はじめに

クラスタ例からの学習⁷⁾とは、例からの学習にクラスタリングの要素を加えたものである。これは、未知の(学習事例集合には含まれない)分類対象集合が与えられたとき、その集合を適切に分割するための規則を獲得する学習問題である。

文献7)は、この学習に対する具体的な解法を提案し、実験を行った。しかし、この方法ではまだ十分に適切な分割が推定されていない問題があった。この問題の原因の1つが、次のような分類対象集合の表現方法の制限にあると考えた。文献7)では、例からの学習が個々の分類対象をその対象に関連付けた属性値ベ

クトルによって表現すると同様に、集合の様々な部分に関連付けた複数の属性値ベクトルによって個々の分類対象集合を表現していた。ところが、これらの属性値ベクトルは、分類対象集合の要素とその対といった局所的な部分に関連付けられていたため、これら局所的な特徴だけを手がかりに分割を推定せざるをえず、クラスタの数などの大域的な情報を参照して分割を推定することができなかった。そこで、本論文では、分類対象全体に関連付けた属性ベクトルを導入し、このベクトルを利用することで大域的な情報を考慮して未知の分類対象集合を分割できるよう改良する。さらに、この改良によって、より適切な分割の推定が可能であることを実験的に示す。

以下、2章ではクラスタ例からの学習について、3章では新たな学習方法について、4章では実験の方法と結果について、そして、5章ではまとめを述べる。

[†] 電子技術総合研究所
Electrotechnical Lab.

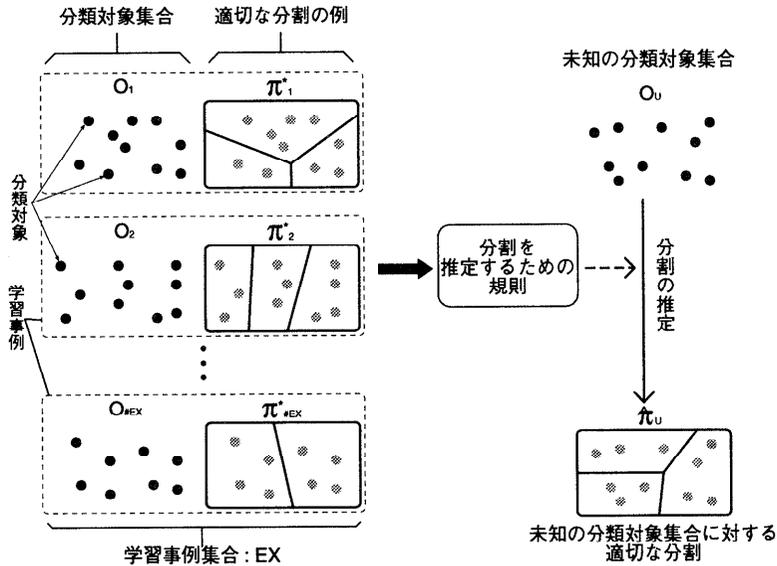


図1 クラスタ例からの学習の概要

Fig. 1 An review of learning from cluster examples.

2. クラスタ例からの学習

2.1 クラスタ例からの学習とは

ここでは、クラスタ例からの学習⁷⁾の概要と形式的定義について述べる。

クラスタリングとは、分類対象 o^i の集合 O が与えられたときに、内的なまとまり (internal cohesion) と外的な分離 (external isolation) が達成されるようなクラスタと呼ぶ部分集合に、分類対象集合を分割することである¹⁾。ただし、内的なまとまりや外的な分離といった規準、もしくは、規則は分類対象集合が与えられる以前に定める必要がある。クラスタ例からの学習とは、このクラスタリングの要素を、Quinlan の ID3⁸⁾ に代表される例からの学習に加えたもので、未知の分類対象集合が与えられたとき、この集合に対する適切な分割を推定するための規則を、与えられた学習事例集合 EX から獲得する問題である。ただし、学習事例は、例からの学習の場合のように、分類対象とそれが分類されるべきクラスの具体例の組ではなく、分類対象集合 O_I とこの集合に対する真に適切な分割の具体例 π_I^* の組である。この学習は、適切な分割そのものを示すことは容易だが、その分割を得るための規則は容易には獲得できない状況 (例を 2.2 節に示す) で有用である。

図 1 はクラスタ例からの学習の概略を示したもので、左側は学習段階、右側は学習結果を利用した分割

の推定段階を表す。学習段階では、学習事例集合 EX から分割を獲得するための規則を学習し、推定段階では、学習した規則を用いて、与えられた未知の分類対象集合 O_U に対する推定分割 $\hat{\pi}_U$ を求める。

以下、このクラスタ例からの学習を形式的に定義する。学習事例集合 EX は $\#EX$ 個の事例からなる集合 $\{(O_1, \pi_1^*), (O_2, \pi_2^*), \dots, (O_{\#EX}, \pi_{\#EX}^*)\}$ であり、各学習事例は分類対象集合 O_I と、これに対する真に適切な分割の具体例 π_I^* の組である。分類対象集合 O は $\#O$ 個の分類対象 o^i からなる集合であり、分割 π は、 $\#\pi$ 個の O の部分集合の組である。これらの部分集合はクラスタと呼ばれ、互いに素で、 O の要素はいずれかのクラスタの要素である。また、 O に対する真に適切な分割を π^* で表し、推定分割を $\hat{\pi}$ で表す。

ID3 に代表される例からの学習では、一般に、属性ベクトルによって分類対象が表現される。属性とは、分類対象の大きさや位置といったある観点からみた特徴を表現するためのもので、分類対象の特徴を事前に定めておいた定義域中の属性値によって表す。属性ベクトルは、異なる観点からみた複数の属性で構成されるベクトルである。クラスタ例からの学習では、以下の 3 種類の属性ベクトルを複数用いて分類対象集合を表現する。

1 種類目は、各分類対象の特徴を表現するための属性ベクトルで、これを分類対象属性ベクトルと呼ぶ。1 つの分類対象集合につき、その集合の要素数 $\#O$ に

等しい個数のベクトルが関連付けられる。この属性は関連付けられた分類対象の、位置や形状といった特徴を表現するために利用される。

2種類目は、分類対象対の特徴を表現するための属性ベクトルであり、これを分類対象対属性ベクトルと呼ぶ。分類対象対とは O 中の分類対象 o^i と o^j の組であり、この分類対象対を p^{ij} で表す。ただし、対となる分類対象の順序は考慮しない、すなわち、 p^{ij} と p^{ji} は同じものとして扱う。 O の要素から作ることでできるすべての分類対象対の集合を P とし、その要素数 $\#P$ は、 P の定義により、 $\#O(\#O - 1)/2$ 個になる。この属性は、1つの分類対象集合につき $\#P$ 個のベクトルが関連付けられ、関連付けられた分類対象対の間の距離といった特徴を表現するために利用される。

3種類目は、本論文で新しく導入する、分類対象集合全体と分割全体の特徴を表現するための属性ベクトルであり、これを分類対象集合全体の属性ベクトルと呼ぶ。分類対象集合全体の特徴に加えて分割の特徴もあわせて表現するため、分類対象集合が与えられた時点ではなく、分類対象集合に対して分割を定めた時点で、集合と分割の両方に依存して属性値が定まる。この属性は、分類対象集合1つにつき1個の属性値ベクトルが関連付けられ、クラスタの数といった分類対象集合全体の大域的な特徴を表すために利用される。

分類対象集合を π に分割したとき、分類対象 o^i 、分類対象対 p^{ij} 、および、分類対象集合全体に関連付けられた属性値ベクトルをそれぞれ $A(o^i)$ 、 $A(p^{ij})$ 、および、 $A(\pi)$ と表記する。また、これらのベクトルの s 番目の属性値をそれぞれ $a^s(o^i)$ 、 $a^s(p^{ij})$ 、および、 $a^s(\pi)$ で、ベクトルの要素数を $\#A(O)$ 、 $\#A(P)$ 、および、 $\#A(\pi)$ で表記する。本論文では、アルゴリズムの制限により、分類対象属性と分類対象対属性の定義域は、各属性は ID3 と同様の形式で、その定義域が順序のある離散値である連続値属性の場合と、定義域が順序のない離散値である離散値属性の場合があり、分類対象集合全体の属性の定義域は $[0, 1]$ の範囲の実数とした。最後に、以上で定義した記号を表 1 に整理しておく。

2.2 クラスタ例から学習の必要性

クラスタ例の学習が必要とされる場合の一例として、画像のセグメンテーションの問題について述べる。セグメンテーションとは、図 2 の (a) の原画像を、(b) のように意味があると考えられる部分ごとに切り分ける（点線で示した部分で分離する）操作である。この図の場合、論理回路図の画像が与えられ、それを図面

表 1 記号定義一覧
Table 1 The list of notations.

EX	事例集合
$\#EX$	事例集合中の事例数
(O_I, π_I^*)	事例
O	分類対象集合
O_U	未知の分類対象集合
$\#O$	集合 O 中の分類対象数
o^i	分類対象
$A(o^i)$	分類対象属性値ベクトル
$a^s(o^i)$	$A(o^i)$ の s 番目の属性値
$\#A(O)$	$A(o^i)$ の属性値の数
P	分類対象対の集合
$\#P$	集合 P 中の分類対象対の数
p^{ij}	分類対象 o^i と o^j との対
$A(p^{ij})$	分類対象対属性値ベクトル
$a^s(p^{ij})$	$A(p^{ij})$ の s 番目の属性値
$\#A(P)$	$A(p^{ij})$ の属性値の数
π	O の分割
π^*	O の真に適切な分割
$\hat{\pi}$	O の推定した適切な分割
$\#\pi$	O の分割のクラスタ数
$A(\pi)$	分類対象集合全体の属性値ベクトル
$a^s(\pi)$	$A(\pi)$ の s 番目の属性値
$\#A(\pi)$	$A(\pi)$ の属性値の数

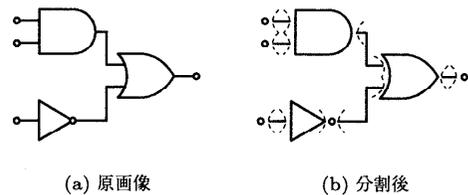


図 2 画像のセグメンテーションの例
Fig. 2 An example of image segmentation.

の部品ごとに切り分けている。このセグメンテーションは画像認識の過程でよく利用され、その結果の良否は認識結果に大きく影響する。

セグメンテーションは、クラスタリングの技法を利用して行われることが多いが、次のような問題がある。クラスタリングを適用するためには、分類対象が同じクラスタに分類されるかどうかを判定するための、分割の基準をあらかじめ明示しておく必要があるが、これは一般には困難である。なぜならば、原画像は、画素の位置や線の長さといった観測値で表されているのに対し、分割の目標は図面の部品といった抽象的な概念であり、両者は大きく隔たっていて、直観的に人間が対応を見いだすことが難しいからである。それに対し、適切な分割の具体例を与えることは容易である。よって、この具体例をもとにクラスタ例からの学習によってセグメンテーションを行うことができれば有用であるとする。

3. 学習と分割推定の方法

3.1 従来の学習と分割推定の方法とその問題点

文献 7) では、最初に、未知の分類対象集合 O_U の分類対象対集合 P_U のすべての要素について、学習段階で獲得した規則（具体的には確率分布関数）を用いて非類似度を計算し、これらの非類似度をもとに代表的なクラスタリング手法である最小距離法、最大距離法、および、群平均法を適用して分割を推定した。ここでいう非類似度とは、分類対象対の 2 つの要素がどれだけ似ていないか、すなわち、同じクラスに分類されにくいさを表す量である。分類対象対 p^{ij} の非類似度は、分類対象属性値ベクトル $A(o^i)$ と $A(o^j)$ 、および、分類対象対属性値ベクトル $A(p^{ij})$ と、学習段階で獲得しておいた確率分布関数 $f_1(\cdot)$ とを用いて $1 - f_1(p^{ij})$ で定義される。この確率密度関数 $f_1(\cdot)$ は次式で定義される。

$$f_1(p^{ij}) \stackrel{def}{=} \Pr[\text{in}(p^{ij}, \pi^*) = 1 | A_C(p^{ij})]$$

ただし、 $\Pr[\cdot]$ は確率または確率密度を表し、 $\text{in}(p, \pi)$ は、分類対象対 p が分割 π の同じクラスに属するとき 1、それ以外の場合 0 とする関数である。また、 $A_C(p^{ij})$ は属性 $A(p^{ij})$ 、 $A(o^i)$ 、および、 $A(o^j)$ を次の手順で合成した属性である。

- (1) $A(p^{ij})$ のすべての要素を $A_C(p^{ij})$ に追加
- (2) $A(o^i)$ と $A(o^j)$ の同じ位置の属性値を取りだし、次の方法で $A_C(p^{ij})$ に順に追加
 - 連続値属性の場合、2 つの属性値を比較して、小さい方から順に 2 つの値を $A_C(p^{ij})$ に追加
 - 離散値属性の場合、2 つの値 $a(o^i)$ と $a(o^j)$ を組み合わせ、新たな属性値 " $a(o^i) - a(o^j)$ " を作成し、 $A_C(p^{ij})$ に追加。ただし、" $a(o^i) - a(o^j)$ " と " $a(o^j) - a(o^i)$ " は同じものと見なす。たとえば、 \mathbf{Y} と \mathbf{N} の 2 つの属性値を定義域とする属性の場合、組み合わせた属性の定義域は $\mathbf{Y} - \mathbf{Y}$ 、 $\mathbf{N} - \mathbf{N}$ 、および、 $\mathbf{Y} - \mathbf{N}$ となる

確率分布関数 $f_1(\cdot)$ は、学習事例集合から以下の手順によって獲得する。最初に、与えられた事例集合 EX のすべての事例 (O, π^*) を、次の手続きによって変換することにより、新たな学習事例集合 ex_1 を生成する。分類対象対集合 P のすべての要素 p に対して関数 $\text{in}(p, \pi^*)$ の値と合成属性ベクトル $A_C(p)$ を求め、これらの組を新たな事例として ex_1 に加える。この ex_1 から、文献 5) のものを改良した付録 A.2 のアルゴリズムによって $f_1(\cdot)$ を学習する。

この分割の推定方法には次のような問題があった。

1 つは、分類対象の間の非類似度のみをもとに分割を推定したことによる問題。もう 1 つは、最小距離法、最大距離法、および、群平均法といったクラスタリング手法を用いることによって生じる問題である。前者の問題は、2.2 節の図 2 を例に説明すると、図全体に注目すると、この図には回路図という意味がある。しかし、この図の局所的な部分に注目すると単なる線という意味しかない。よって、適切な分割を導くためには、分類対象集合全体の大域的な特徴を考慮する必要があると考えるが、上記の方法では分類対象の間の非類似度という局所的な視点だけで分割を推定する。文献 7) では、大域的な特徴を考慮するために、属性ベクトルに対する工夫で取り組んだが、より直接的にこのような特徴を利用できるべきである。後者の問題としては、最小距離法の場合、この方法では個々の分類対象が 1 つのクラスとなった初期分割のクラスを一つずつ併合することで適当なクラスを求めようとするのだが、併合すべきクラス対の対として、一方のクラス対の任意の分類対象ともう一方の任意の分類対象との間の非類似度が最小であるものが選ばれる。そのため、たとえ他の分類対象対の非類似度が適切に求められていても、たった 1 つでも非類似度が誤って小さく推定されている分類対象対があると、望ましくないクラス対が併合されてしまうため、ロバスト性に欠けている問題がある。また、最大距離法の場合にも最小距離法と同様のロバスト性の問題があり、群平均法を用いた場合では、分割のクラス数を推定できない問題がある。

3.2 分割の推定方法の改良

本論文では、前節の問題を解決する分割の推定方法について述べる。

まず、分割を推定する分類対象集合 O についてある分割 π を定めた場合を考える。このとき、 π が真に適切な分割であるという事象 $\pi = \pi^*$ を用い、この事象と属性値ベクトルとの結合確率を、あらゆる可能な分割の中で最大にするものを、最も適切な推定分割にするという方針を本論文では採用した。これは、確率モデルの母数推定で、観測されたデータと母数の結合確率を最大にする最尤推定と同様の方針である。この結合確率は次式で表される。

$$\Pr[\pi = \pi^*, A(\pi), A(O), A(P)] \quad (1)$$

ただし、 $A(O)$ と $A(P)$ は、それぞれ O と P の要素に関連付けられた属性値ベクトルの集合を表す。この結合確率は、全部で $\#O + \#P + 1$ 個と非常に多くの属性値ベクトルを含み、容易には計算できない。そこで、この式を以下の 3 つの項の積に分解し、それぞ

れを計算することを試みる.

$$\Pr[A(\pi)|\pi=\pi^*, A(O), A(P)] \quad (2)$$

$$\Pr[\pi=\pi^*|A(O), A(P)] \quad (3)$$

$$\Pr[A(O), A(P)] \quad (4)$$

まず, 式 (4) は O のいかなる分割に対しても一定なので, 式 (1) の最大化に関して無視できる.

次に, 式 (2) の計算は非常に複雑で困難なので, 事象 $\pi=\pi^*$ が発生した条件のもとで $A(O)$ または $A(P)$ と $A(\pi)$ とは独立である, すなわち, 次式が成立するという仮定を導入する.

$$\Pr[A(\pi), A(O), A(P)|\pi=\pi^*] =$$

$$\Pr[A(\pi)|\pi=\pi^*] \times \Pr[A(O), A(P)|\pi=\pi^*]$$

この仮定はいつも成立するわけではないが, この仮定の下で実験では良い結果が得られた. これは, $A(O)$ や $A(P)$ は局所的な視点から見ているのに対し, $A(\pi)$ は大域的な視点から見たものなので, これらの間の相関は一般には小さいからであると考え. また, 分類対象属性や分類対象対属性と分類対象集合全体の属性との間に明らかな相関がある場合には, 一方の属性が表現する特徴をもう一方の属性でも表現できることを示している. 一方の属性は必然的に不要となるため事実上問題が生じないともえる. この仮定のもとで, 式 (2) は次の確率密度に等しくなる.

$$\Pr[A(\pi)|\pi=\pi^*] \quad (5)$$

この確率密度は, 3.3 節で述べる方法によって学習段階で獲得しておいた確率密度関数 $f_2(A(\pi))$ を利用して計算される.

最後に, 式 (3) は, Dempster&Shafer の確率結合則²⁾ (以後, D&S 則と略す. 概略は付録 A.1 に記した) を利用し, すべての分類対象対 p について前節の確率分布 $f_1(p)$ を求め, これらを統合することで求める. 以下, D&S 則を利用して式 (3) を具体的に計算する方法について述べる. 分割集合 Π を入力とし, 事象の集合を出力する関数 ev を次式で定義する.

$$\text{ev}(\Pi) = \{\pi=\pi^* : \forall \pi \in \Pi\}$$

分類対象集合 O の可能なあらゆる分割を含む集合を Π_{AU} とする. 次に, 3.1 節の確率分布関数 $f_1(p)$ の解釈について考察する. $f_1(p)$ は分類対象対 p とこの対を構成する 2 つ分類対象に関連付けられた 3 つの属性値ベクトルが与えられた場合に, p が真に適切な分割の同じクラスタの要素となるという事象が生じる確率である. これは言い換えると, 分割集合 $\Pi(p) = \{\pi : \text{in}(p, \pi) = 1, \forall \pi \in \Pi_{AU}\}$ を用いて, 3 つの属性値ベクトルを証拠とした基本確率 $\Pr[\text{ev}(\Pi(p))|A_C(p)]$ となる. 一方, $1 - f_1(p)$ は,

$\bar{\Pi}(p) = \Pi_{AU} - \Pi(p)$ とすると, 同じ証拠に基づく基本確率 $\Pr[\text{ev}(\bar{\Pi}(p))|A_C(p)]$ に相当する. この証拠に基づく基本確率は, これら 2 つ以外の事象集合については 0 となる. すべての分類対象対について, このような基本確率の組を考えることができるので, 全部で $\#P$ 個の基本確率の組ができることになる. これら $\#P$ 個の組を統合して, 事象 $\pi=\pi^*$ が発生する確率を求めれば, この確率は $A(O)$ と $A(P)$ を証拠とする, すなわち, $A(O)$ と $A(P)$ が与えられた場合の確率である式 (3) が得られることになる. 具体的に, 分類対象集合を π に分割した場合の統合確率は次の方法で求められる. すべての p について, $\text{ev}(\Pi(p))$ か $\text{ev}(\bar{\Pi}(p))$ のうち $\pi=\pi^*$ を含む方を選択すると, これらの $\#P$ 個の事象集合の交わりはちょうど $\pi=\pi^*$ だけを含む事象集合となる. よって, 統合確率の分子は, $\pi=\pi^*$ が $\text{ev}(\Pi(p))$ の要素であるときは基本確率 $\Pr[\text{ev}(\Pi(p))|A_C(p)] = f_1(p)$ を, そうでない場合は $\Pr[\text{ev}(\bar{\Pi}(p))|A_C(p)] = 1 - f_1(p)$ を選び, これらの積である次式で表せる.

$$\prod_{p \in P^+} f_1(p) \times \prod_{p \in P^-} (1 - f_1(p)) \quad (6)$$

ただし, $\text{in}(p, \pi) = 1$ を満たす分類対象対で構成される P の部分集合を P^+ , この補集合を P^- で表す. 一方, 分母は分割 π の選択にかかわらず不変なので式 (3) の最大化に関しては無視できるので, 式 (3) の定数倍が式 (6) により得られる.

以上の議論から, 式 (5) と式 (6) の積を最大にすれば, 式 (1) を最大化できることが分かる.

最初に定めた方針に基づき, 式 (1) を最大にするような分割を探索することにより未知の分類対象集合に対する適切な分割を推定できる. しかし, 可能な分割の総数は非常に多く, 最適な分割を実際には求めることができないので, 図 3 の方法で準最適な分割を探索した. これは, 最初に各クラスタに分類対象が 1 個ずつだけ含まれる初期分割を作成し, 式 (1) が大きくなるように, 漸次この分割を変形する手法である. また, 手続き Evaluation は, 評価対象の分割と現在の分割を比較して, より適切な分割を選択する手続きだが, 式 (5) が 0 となる場合に対する例外的な処理を含んでいる.

最後に, 本論文の方法の利点についてまとめておく. 文献 7) では分類対象対の非類似度だけに依存していたが, それに対して, 分類対象集合全体の属性も考慮して分割を推定している. さらに, 最小距離法や最大距離法のように, たった 1 個の分類対象対の非類似度に依存してクラスタが併合されるかどうか決定され

```

t := 0, π0 := {C = {o}, ∀ o ∈ O}
if (Pr[A(π0)|π0=π*] > 0) then {
  f := true, E0 := Pr[A(π0)|π0=π*]Pr[π0=π*|A(O), A(P)]
} else {
  f := false, E0 := Pr[π0=π*|A(O), A(P)]
}
start:
  t := t + 1, Et := Et-1
  forall (CA ∈ πt-1, CB ∈ πt-1, CA ≠ CB) {
    π' := πt-1 - CA - CB + {CA ∪ CB}, call Evaluation(π')
  }
  forall (CA ∈ πt-1, CB ∈ πt-1, CA ≠ CB) {
    forall (o ∈ CA) {
      π' := πt-1 - CA - CB + {CA - {o}} + {CB ∪ {o}}, call Evaluation(π')
    }
  }
  if (f = false ∨ Et ≠ Et-1) then goto start
output πt-1

```

手続き Evaluation (π') の定義

```

if (f = false) then {
  if (Pr[A(π')|π'=π*] > 0) then {
    f := true, πt := π', Et := Pr[A(π')|π'=π*]Pr[π'=π*|A(O), A(P)]
  } else if (Pr[π'=π*|A(O), A(P)] > Et-1) then {
    πt := π', Et := Pr[π'=π*|A(O), A(P)]
  }
} else if (Pr[A(π')|π'=π*]Pr[π'=π*|A(O), A(P)] > Et-1) then {
  πt := π', Et := Pr[A(π')|π'=π*]Pr[π'=π*|A(O), A(P)]
}

```

図 3 分割探索アルゴリズム

Fig. 3 Our algorithm for searching an appropriate partition.

ないため、ロバスト性に優れ、クラスタ数を事前に与えなければならぬ群平均法のような問題もない。このように以前の方法の問題を本論文の方法は解決している。

3.3 確率密度学習アルゴリズム

前節では学習段階で獲得しておいた確率密度関数 $f_2(\cdot)$ を利用して式 (5) を求めると述べたが、ここでは、この関数の獲得方法について述べる。

この確率密度関数 $f_2(A(\pi)) = \text{Pr}[A(\pi)|\pi=\pi^*]$ は、ID3 で採用されている決定木に類似した次の回帰木を用いて記述される。回帰木は図 4 のような二分木で、長方形で表した非終端ノードと、長丸で表した終端ノードがある。非終端ノードにはしきい値、しきい値と比較すべき属性値、および、左右 2 本の枝があり、属性値がしきい値未満であれば左のノードを、そうでなければ右のノードをたどる。終端ノードには出力すべき確率密度がある。また、最も上のノードをルートノードと呼び、属性値としきい値の比較はこのノードから開始される。

属性値ベクトル (0.3, 0.5) を図 4 に与えた場合を例にとり、確率密度が出力される過程を示す。最初に、

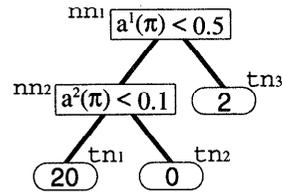


図 4 回帰木の例

Fig. 4 An example of a regression tree.

ルートノード nn₁ で第 1 の属性値 0.3 としきい値 0.5 とが比較される。属性値はしきい値よりも小さいので、左の枝をたどりノード nn₂ に行く。nn₂ も非終端ノードなので、そのしきい値 0.1 と第 2 の属性値 0.5 とが比較される。今度は、属性値の方が大きいので右の枝をたどり tn₂ へ行く。このノードは終端ノードなので、このノードの値 0 が確率密度として出力される。

以下、この回帰木の学習方法について述べる。最初に、EX の各事例に対して $A(\pi_i)$ を求め、新たな学習事例集合 $ex_2 = \{A(\pi_1), A(\pi_2), \dots, A(\pi_{\#EX})\}$ を作成する。この学習事例集合から回帰木を求めるために、確率モデルの選択基準である Rissanen の MDL

```

 $ex_2 := \{A(\pi_1), A(\pi_2), \dots, A(\pi_{\#EX})\}$ : 学習事例
RT := (TN, NN): ルートが終端ノードである回帰木
  TN := { $tn_1$ }: 終端ノードの集合 ( $tn_1$  はルートノード)
  NN := {}: 非終端ノードの集合
for s from 1 to #A( $\pi$ ) {
   $\delta^s := 6 \times \langle (a^s(\pi_1), a^s(\pi_2), \dots, a^s(\pi_{\#EX})$  の標準偏差) / #EX
}
start:
RTBEST := RT
foreach  $tn'$  in TN {
  for s from 1 to #A( $\pi$ ) {
    領域  $R(tn')$  で s 番目の属性値に関する下限を  $l$ , 上限を  $u$  とする.
    for d from 1 to  $\infty$  {
       $q := (1/2)^d$ 
      if ( $q < \delta^s$ ) then goto checkend
      for t from 1 to  $2^{(d-1)}$  {
         $\theta := q(2t - 1)$ 
        if ( $l \leq \theta < u$ ) then {
          RT' := (TN', NN')
          NN' := NN  $\cup$  nnU (nnU は  $\theta$  をしきい値とする非終端ノードで,  $tn'$  があった位置に作成)
          TN' := {TN -  $tn'$ }  $\cup$  { $tn_U^l, tn_U^r$ } ( $tn_U^l$  と  $tn_U^r$  はそれぞれ nnU の左と右の枝のノード)
          if ( $\ell(EX_2, RT') < \ell(EX_2, RT)$ ) then RT := RT'
        }
      }
    }
  }
}
checkend:
}
}
if (RTBEST  $\neq$  RT) goto start
end:
output RTBEST

```

図 5 回帰木探索アルゴリズム

Fig. 5 Our algorithm for searching a proper regression tree.

原理¹¹⁾を利用した。これは「与えられたデータを、モデル自身の記述も含めて最も短く符号化できるような確率モデルが最良のモデルである」というものである。言い換えれば、回帰木自身を符号化する方法とこの回帰木を利用して学習事例 ex_2 を符号化する方法とを定め、これらの符合長の合計を最小にする回帰木が確率密度を最も適切に表現しているということである。この MDL 原理に基づき回帰木を獲得するには、符号化の方法と、最小の符合長を実現する回帰木の探索アルゴリズムが必要となるが、これらについて順に述べる。

全体の符合長 $\ell(ex_2, RT)$ を、事例集合 EX_2 と回帰木の終端ノードの確率密度の値とを合わせた符合長 $\ell(ex_2|RT)$ と回帰木の残りの部分の符合長 $\ell(RT)$ との和で表す。 $\ell(RT)$ は付録 A.3 に示し、ここでは $\ell(ex_2|RT)$ について述べる。

$\ell(ex_2|RT)$ は文献 14) より

$$-\log \mathcal{L}(EX_2|RT) + \frac{1}{2} \#TN (\log e + \log \#EX)$$

となる。ただし、 $\mathcal{L}(ex_2|RT)$ は回帰木の表す確率密度分布に基づく ex_2 の尤度であり、本論文の対数の底

はすべて 2 である。また、 TN は RT の終端ノードの集合で、 $\#TN$ はその要素数である。同様に、 NN は RT の非終端ノードの集合で、 $\#NN$ はその要素数と定めておく。この、尤度 $\mathcal{L}(ex_2|RT)$ は次式で表される。

$$\mathcal{L}(ex_2|RT) = \prod_{tn_x \in TN} \Pr[tn_x]^{\#tn_x}$$

ただし、 $\#tn_x$ は ex_2 の事例のうち tn_x にたどりつくもの数である。また、 $\Pr[tn_x]$ は終端ノード tn_x での確率密度であり、次式で表される。

$$\Pr[tn_x] = \frac{\#tn_x}{\#EX \times V(R(tn_x))}$$

ただし、ある終端ノード tn_x について、もし属性値が領域 $R(tn_x)$ の値であれば終端ノード tn_x にたどりつくものとし、また、この領域の体積を $V(R(tn_x))$ と表記した。たとえば、図 4 のノード tn_2 の場合、入力属性値が $a^1(\pi) < 0.5$, $a^2(\pi) \geq 0.1$ であればこのノードにたどりつくので、 $R(tn_2)$ は $(0 \leq a^1(\pi) < 0.5) \wedge (0.1 \leq a^2(\pi) \leq 1)$ の領域となり、 $V(R(tn_2))$ は 0.45 となる。

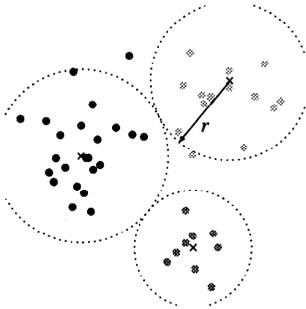


図 6 ドット・パターン例
Fig. 6 An example of a dot-pattern.

記述長 $\ell(EX_2, RT)$ を最小にする回帰木を求めるアルゴリズムを図 5 に示す。このアルゴリズムは、ID3 など決定木を学習するアルゴリズムで一般的な手法である、divide-and-conquer を利用したものである。すなわち、終端ノード 1 つだけでできた初期状態の木を生成する。そして、記述長を短くするような木を、終端ノードを再帰的に分割していくことによって探索し、これ以上上の終端ノードを分割しても記述長が短くならなくまでこの分割操作を続ける。

4. 実験

4.1 実験対象

文献 7) のドットパターン 3 種類と論理回路図画像の合計 4 種類の事例集合に対して分割を推定する。

ドットパターンとは、図 6 のように平面にドットが分布している分類対象集合で、クラスタリング手法の実験的検証によく利用される。各分類対象集合は 50 個のドットから構成されており、3~5 個のクラスタに分割されている。各クラスタ内ではドットは円形のガウス分布に従って分布しており、そのガウス分布の標準偏差が異なる 3 種類の事例集合を作成した。これらを重複、接触、および、分離と呼び、それぞれ標準偏差が $r/2.0$, $r/2.5$, そして、 $r/3.0$ の事例集合である。ただし、 r は各クラスタごとに与えたパラメータで、図 6 の点線のように、各クラスタを中心にこのパラメータを半径とする領域が、他のクラスタのこの領域と接触し、かつ、重ならない条件のもとでランダムに定めた。分割の推定には、文献 6) の 4 種類の分類対象属性と 8 種類の分類対象対属性、そして、次の 3 種類の分類対象集合全体の属性を利用した。

- 〈クラスタ数〉/〈分類対象数〉
- 〈クラスタ内要素数の標準偏差〉/〈分類対象数〉^{*}
- 〈クラスタ内要素数の最大値〉/〈分類対象数〉

^{*} [0,1] の範囲を超える場合は 0 または 1 に正規化。

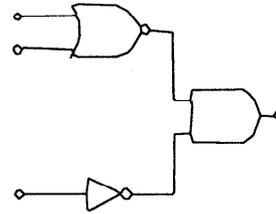


図 7 論理回路図画像例
Fig. 7 An example of a logic diagram image.

論理回路図画像とは、図 7 のような、AND, OR, バッファ、端子、および、接続線の 5 種類の部品で構成された図面の画像である。これは、手書きの図面をスキャナでビットマップ画像とし、それを細線化・ベクトル化といった処理によって、直線によって構成されるベクトルデータと呼ばれる画像に変換したものである。これらの画像について、1 つの部品を表す直線の集合が各クラスタとなるような分割を推定した。これは、各分類対象がガウス分布のような単純な分布に従って発生しているわけではないので、ドットパターンよりも実用的な実験である。分割の推定には、文献 4) の 8 種類の分類対象属性と 7 種類の分類対象対属性、そして、ドットパターンと同じ 3 種類の分類対象集合全体の属性を利用した。この事例集合の平均クラスタ数は 16.7 個、平均分類対象数は 102.9 個である。

4.2 評価方法

ここでは、クラスタ例からの学習によって、どれだけ真に適切な分割に近い分割が推定されたかを評価する方法について述べる。与えられた事例集合 EX の最初の事例 (O_1, π_1^*) を取り出し、これをテスト用事例、残りを学習用事例とする。学習用事例から獲得された規則を利用して、分類対象集合 O_1 に対する推定分割 $\hat{\pi}_1$ を求める。この推定分割と真に適切な分割 π_1^* との間の RIL (後述) を求める。これら一連の操作を残りの 2 から $\#EX$ 番目までの学習事例に対して行い、RIL の平均値を求める。この平均値によってどれだけ適切な分割が推定できているかを評価した。

RIL (Ratio of Information Loss) とは次のような値である。真の分割が生じる事象を Π^* 、推定分割が生じる事象を $\hat{\Pi}$ と記す。事象 Π^* の事前エントロピー $H(\Pi^*)$ と、推定分割を知ったあとの事後エントロピー $H(\Pi^*|\hat{\Pi})$ を用いて RIL は次式で表される。

$$RIL = \frac{H(\Pi^*|\hat{\Pi})}{H(\Pi^*)}$$

言い換えれば、RIL は真の分割について知るかわりに推

表 2 ドットパターン各事例集合に対する RIL の平均
Table 2 The means of RILs for example sets of dot-patterns.

(a) データ「重複」				
	今回	最小距離	最大距離	群平均
平均	0.364	0.991	0.658	0.462
標準偏差	0.2370	0.0428	0.1755	0.2001
t 値		25.9	9.9	3.2
(b) データ「接触」				
	今回	最小距離	最大距離	群平均
平均	0.160	0.825	0.452	0.282
標準偏差	0.1775	0.3058	0.2304	0.1722
t 値		18.7	10.0	4.9
(c) データ「分離」				
	今回	最小距離	最大距離	群平均
平均	0.068	0.416	0.208	0.204
標準偏差	0.1472	0.3996	0.2295	0.1731
t 値		8.1	5.1	5.9

表 3 論理回路図画像の事例集合に対する RIL の平均
Table 3 The means of RILs for an example set of logic diagram images.

	今回	最小距離	最大距離	群平均
平均	0.430	0.744	0.700	0.409
標準偏差	0.1196	0.1722	0.0925	0.1337
t 値		14.9	17.8	-1.2

定分割について知っていた場合、真の分割を知っている場合に比べてどれだけの割合の情報量の損失があるかを表す。0に近いほど真の分割に近い推定分割が獲得できたといえ、また、推定分割が真の分割と一致するときだけ0になる。具体的に、 $H(\Pi^*)$ と $H(\Pi^*|\hat{\Pi})$ は次式で表される。

$$H(\Pi^*) = \sum_{x=0}^1 \frac{N_x}{\#P} \log \frac{\#P}{N_x}$$

$$H(\Pi^*|\hat{\Pi}) = \sum_{x=0}^1 \sum_{y=0}^1 \frac{M_{xy}}{\#P} \log \frac{M_{0y} + M_{1y}}{M_{xy}}$$

ただし、 N_x は P の要素のうち $\text{in}(p, \pi^*) = x$ を満たすもの数、 M_{xy} は、 $\text{in}(p, \pi^*) = x$ かつ $\text{in}(p, \hat{\pi}) = y$ を満たすもの数である。

4.3 実験結果と考察

文献 7) の方法と本論文の方法を、ドットパターンと論理回路図画像の事例集合に適用した結果を表 2 と表 3 にそれぞれ示す。どの事例集合も 100 個の事例で構成され、表中には前節で述べた RIL の平均値と標準偏差、および、 t 値を記した。

t 値は、2つの観測値集合が異なる平均を持つ母集団からとられた標本と見なされるかどうかを検定するために用いられる。今回の実験の結果の実験回数を

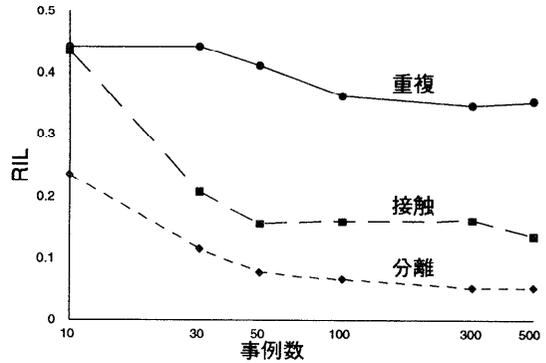


図 8 ドットパターン事例集合に対する学習曲線
Fig. 8 Learning curves for example sets of dot-patterns.

$n_1 (= 100)$ 、平均を μ_1 、そして、標準偏差を σ_1 とする。一方、最小距離法（もしくは最大距離法や群平均法）の実験の結果の実験回数を $n_2 (= 100)$ 、平均を μ_2 、そして、標準偏差を σ_2 とする。このとき次式で定義される t 値は、自由度 $n_1 + n_2 - 2 (= 198)$ の t 分布に従う。

$$t \text{ 値} = \frac{\mu_2 - \mu_1}{\sqrt{\frac{n_1\sigma_1^2 + n_2\sigma_2^2}{n_1 + n_2 - 2} \left(\frac{1}{n_1} + \frac{1}{n_2} \right)}}$$

危険率を 1%とした場合、この t 値が自由度 198 の 99%点である $t_{0.99} = 2.34533$ よりも大きければ、 $\mu_1 < \mu_2$ であるといえる。論理回路図画像での群平均法との比較場合の以外、 t 値はいずれも $t_{0.99}$ よりも大きく、今回の手法の方がより適切な分割を推定できている。論理回路図画像での群平均法では RIL は今回の方法の方が大きい、逆に群平均法の方が優れているとも有意にはいえず、ほぼ同じくらいである。群平均法ではクラスタ数を与えなければ分割の推定ができなかったのに対し、今回はその必要がないぶんだけ優れている。

本手法の性質を詳しく調査するために学習曲線を求めた。図 8 は、ドットパターンの各事例集合について事例数を変化させて描いた学習曲線である。4.2 節の方法によって求めた RIL を、事例数が 10, 30, 50, 100, 300, および、500 個の事例集合について求めた。事例数が 100 になるまでは、いずれの事例集合でも順調に RIL は顕著に減少しており、事例をより多く与えることにより、より適切に分割を推定できている。事例数が 100 を超えると明確な減少傾向は見られないが、これは今回採用した属性だけでは適切な分割を十分に記述できないことが原因と推測され、より適切な分割を推定するためには新たな属性が必要になるであろうと考える。

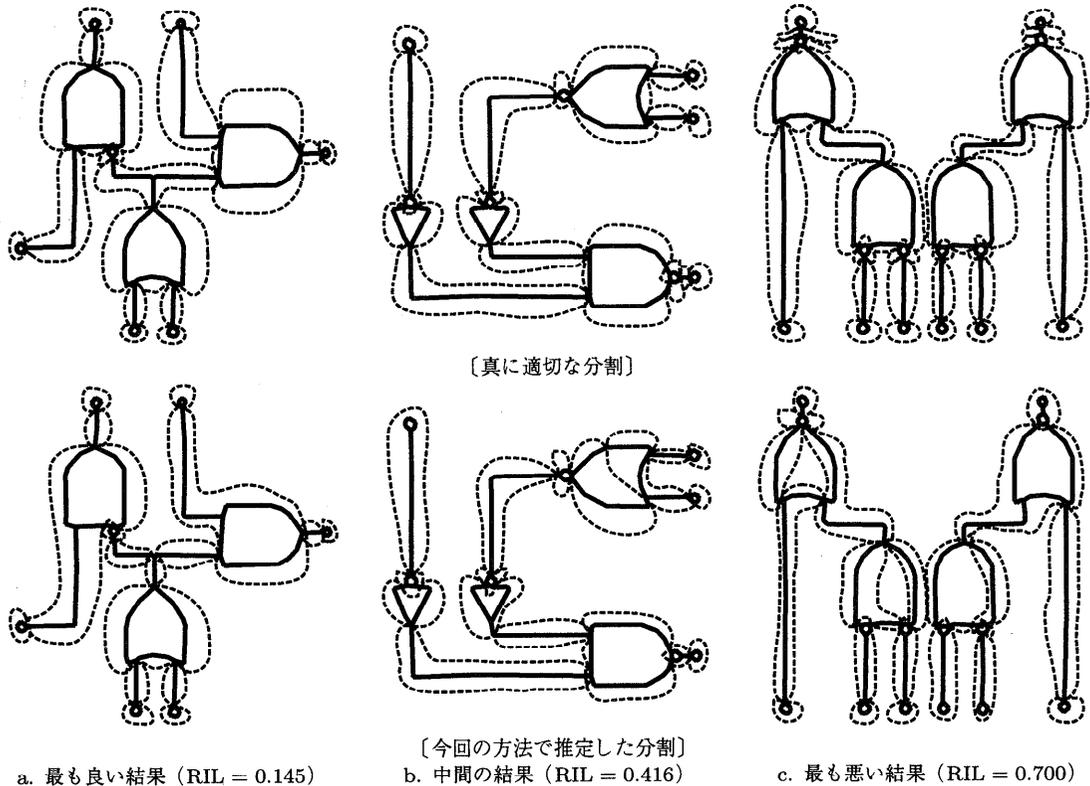


図9 真に適切な分割の例と推定された分割の例

Fig. 9 Examples of true and estimated partitions from an example set of logic diagram images.

論理回路図画像の分割の代表的な推定結果を、同じクラスタに分類された一群の線分を点線で囲うことで、図9に視覚的に示した。上段はその画像に対する真に適切な分割、下段は推定分割であり、a, b, および、cはそれぞれ、真に適切な分割と推定分割の間のRILが最小、中央、最大であった分割である。既存のクラスタリング手法を用いた画像のセグメンテーションには主観的なパラメータ調節などがともなうので、客観的かつ公平に本手法と比較することはできないが、図9に示した推定分割の具体例は、既存のクラスタリング手法によるものとの直観的な比較に役立つものと考えられる。著者は、RILが最大の場合では推定分割は真の分割とややかけ離れており、この分割をもとに図面を認識するのはやや困難ではないかと考えるが、その他の場合ではこの分割結果をもとにした認識は可能であると推測する。

最後に、クラスタリング手法ではなく、クラスタ例からの学習によって分割を推定することの利点について強調したい。従来のクラスタリング手法によっても適切な分割を獲得できる場合もあるが、これらの手法を用いるためには、停止条件、非類似度など事前に定

めなければならない要素が数多く存在する。そのため、ユーザが適切だと考える分割を獲得するには、これらの要素を試行錯誤によって決定しなくてはならないことが多い。言い換えれば、クラスタリング手法によってユーザが適切だと考える分割を獲得することは、コンピュータが自動的に分割を推定したのではなく、クラスタリング手法を道具としてユーザが分割を獲得していたともいえる。一方、クラスタ例からの学習を利用した場合、ユーザは分類対象集合を記述するための属性を定め、分割の具体例を与えれば、試行錯誤の必要もなく分割を推定できる。ただし、分類対象集合を記述するための属性はユーザが分類対象の種類に応じて定めなくてはならない問題は残っている。しかし、様々な種類の属性を適当に作成し、これらを様々に組み合わせることで4.2節の方法でRILを計算し、このRILを大きくするような組合せを採用するようにすれば、半自動的に決定することができ、大きな問題とはならないと考える。本論文の実験で採用した属性もこのような方法によって定めたものである。さらに、4.2節の方法によって求めたRILは、未知の分類対象集合に対して得られる推定分割の適切さの定量的尺度とし

て利用できるが、クラスタリングによって分割を獲得した場合はこのような尺度は得られない。推定した分割を利用する場合に、このような尺度があることも大きな利点である。

5. おわりに

本論文では、クラスタ例からの学習の手法の改良について述べた。今回の手法は分類対象集合全体の属性を考慮して分割を推定することができたため、ロバスト性に優れており、クラスタの数も同時に推定することができた。実験の結果、従来の手法よりも適切な分割を推定できることを示した。

今回の手法で、分類対象集合全体の属性を考慮できたが、クラスタ内の分類対象の分布など、クラスタごとの特徴については考慮できない。今後は、このような特徴も取り扱えるように改良したい。

参考文献

- 1) Everitt, B.S.: *Cluster Analysis*, third edition, Edward Arnold (1993).
- 2) 石塚 満: Dempster&Shafer の確率理論, 信学誌, Vol.66, No.9, pp.900-903 (1983).
- 3) 伊藤秀一: MDL のパターン認識への応用, 人工知能学会誌, Vol.7, No.4, pp.608-614 (1992).
- 4) 神島敏弘: 学習を用いた論理回路図面からの図面部品の抽出, 画像ラボ, Vol.7, No.1, pp.44-47 (1996).
- 5) 神島敏弘, 美濃導彦, 池田克夫: 帰納学習を用いた図面部品の抽出と分類のための規則の形成, 情報処理学会論文誌, Vol.36, No.3, pp.614-626 (1995).
- 6) 神島敏弘, 元吉文男: クラスタ例からの学習—分類対象集合全体に関わる特徴の利用, 人工知能学会研究会資料, Vol.SIG-FAI-9703-10, pp.69-76 (1998).
- 7) 神島敏弘, 新田克己: クラスタ例からの学習, 人工知能学会誌, Vol.12, No.2, pp.276-284 (1997).
- 8) Quinlan, J.R.: *Induction of Decision Trees*, *Machine Learning*, Vol.1, pp.81-106 (1986).
- 9) Quinlan, J.R. and Rivest, R.L.: *Inferring Decision Trees Using the Minimum Description Length Principle*, *Information and Computation*, Vol.80, pp.227-248 (1989).
- 10) Rissanen, J.: *A Universal Prior for Integers and Estimation by Minimum Description Length*, *The Annals of Statistics*, Vol.11, No.2, pp.416-431 (1983).
- 11) Rissanen, J.: *Stochastic Complexity in Statistical Inquiry*, *World Scientific Series in Computer Science*, Vol.15, World Scientific (1989).
- 12) Wallace, C.S. and Patrick, J.D.: *Coding Decision Trees*, *Machine Learning*, Vol.11, pp.7-22 (1993).
- 13) 山西建司: MDL 入門: 計算論的学習理論の立場から, 人工知能学会誌, Vol.7, No.3, pp.435-442 (1992).
- 14) 山西建司, 韓 太舜: MDL 入門: 情報理論の立場から, 人工知能学会誌, Vol.7, No.3, pp.427-434 (1992).

付 録

A.1 Dempster&Shafer の確率結合則

異なる証拠から推論された確率を統合する D&S 則について述べる。まず、事象の集合 E_{All} のべき集合 $P(E_{All}) = \{E : \forall E \subseteq E_{All}\}$ を考える。 $P(E_{All})$ の任意の要素 E について、 E の中のいずれかの事象 e が発生する確率を $\Pr[E]$ で表し、これ E の基本確率と呼ぶ。この基本確率は次の条件を満たす。

$$\Pr[E] \geq 0, \Pr[\emptyset] = 0, \sum_{E \in P(E_{All})} \Pr[E] = 1$$

ここで、証拠 A に基づく基本確率を $\Pr[E|A]$ と表記すると、異なる証拠 A と B に基づいた基本確率を統合した、事象 e の生じる確率は D&S 則により次式で表される。

$$\frac{\sum_{E_A \cap E_B = \{e\}} \Pr[E_A|A] \Pr[E_B|B]}{1 - \sum_{E_A \cap E_B = \emptyset} \Pr[E_A|A] \Pr[E_B|B]} \quad (7)$$

ただし、 E_A や E_B は $P(E_{All})$ の要素で、式中の Σ 記号は事象集合 E_A と E_B の交わりが、 e のみを含む集合 $\{e\}$ や空集合 \emptyset になった場合の和を表す。この D&S 則は、証拠が 3 つ以上の場合にも、事例集合 E_A, E_B, \dots の交わりを考えることで容易に拡張できる。

A.2 確率分布関数 $f_1(\cdot)$ の推定アルゴリズム

このアルゴリズムは、例からの学習とはほぼ同じ問題を取り扱うが、未知の分類対象を正しいクラスに分類することよりも、正しいクラスに分類される確率分布関数を計算することに重点がある点が異なる。

3.1 節の学習事例集合 ex_1 は簡潔に $ex_1 = \{(c_1, A_1), (c_2, A_2), \dots, (c_{\#ex_1}, A_{\#ex_1})\}$ の形で表現できる。 c_I はクラスで 0 または 1 であり、 $A_I = (a^1, a^2, \dots, a^{\#A})$ は ID3 と同様の形式の属性値ベクトルである。この ex_1 から、未知の属性値ベクトル A_U が与えられたときの条件付き確率分布関数 $\Pr[c = 1|A_U]$ を求めることが、本アルゴリズムの目標である。

このアルゴリズムでは、 $\Pr[c = 1|A_U]$ を次の山西の確率的決定リスト¹³⁾ により記述する。これは、 m 個の順序付けされた条件のリスト DL と確率のリス

```

カバーされていない学習事例  $S := ex_1$ , ターム番号  $i := 0$ , リスト  $DL := \langle \rangle$ , 条件付確率  $PR := \langle \rangle$ 
do {
   $i := i + 1$ , タームの更新回数  $j := 0$ ,  $j$  回更新したターム  $T_i^j := \text{true}$ 
  do {
     $j := j + 1$ , 評価値  $G$  を最大にするリテラルを  $L_B$ , その評価値を  $G_B$  とする (注 1)
    if ( $G_B \leq 0$ ) then {
       $T_i^j := T_i^{j-1}$ , goto term_end
    }
     $T_i^j := T_i^{j-1} \wedge L_B$ 
  } until ( $S(T_i^j)$  中の事例のクラスがすべて 0 もしくはすべて 1)
term_end:
  if ( $T_i^j = \text{true}$ ) then goto list_end
   $DL$  に  $T_i^j$  を,  $PR$  に  $\text{Pr}(S(T_i^j))$  を追加 (注 2)
   $S := S - S(T_i^j)$ 
} until ( $S$  中の事例のクラスがすべて 0 もしくはすべて 1)
list_end:
 $DL$  に true を,  $PR$  に  $\text{Pr}(S)$  を追加
(手続き Pruning)
総記述長  $\ell = \ell(ex_1, DL)$ , ターム数  $m := i + 1$ 
while ( $m > 1$ ) {
   $S' := S(T_m) \cup S(T_{m-1})$ ,  $DL' := \langle T_1, \dots, T_{m-2}, \text{true} \rangle$ ,  $PR' := \langle \text{Pr}_1, \dots, \text{Pr}_{m-2}, \text{Pr}(S') \rangle$ ,  $\ell' = \ell(ex_1, DL')$ 
  if ( $\ell \leq \ell'$ ) then goto noprunes
   $DL := DL'$ ,  $PR := PR'$ ,  $m := m - 1$ ,  $\ell = \ell'$ 
}
noprunes:
output  $DL, PR$ 

```

図 10 確率的決定リストの獲得アルゴリズム

Fig. 10 Our algorithm for acquiring the stochastic decision lists.

ト PR の組である。

$$DL = \langle T_1, T_2, \dots, T_{m-1}, \text{true} \rangle$$

$$PR = \langle \text{Pr}_1, \text{Pr}_2, \dots, \text{Pr}_{m-1}, \text{Pr}_m \rangle$$

DL の要素 T_i は true または false の値をとる属性値ベクトル A の関数であり, これをタームと呼ぶ。ただし, DL の最後のタームはつねに true の値をとる。この決定リストの意味は, A を与えたとき, T_1 から T_{i-1} のタームがすべて false で, T_i が true であったとき, 確率分布関数 $\text{Pr}[c = 1|A]$ が値 Pr_i をとるということである。ターム T_i は, n_i 個のリテラル L_j (true か false の値をとる属性値 a^s の関数) の連言である。リテラル L_j は, a^s が連続値属性か, 離散値属性かによって異なる。連続値属性の場合, リテラルは $(a^s < \theta_u)$, $(\theta_l \leq a^s < \theta_u)$, または, $(\theta_l \leq a^s)$ のいずれかの形式で, a^s が不等号を満たすとき true をとる。離散値属性の場合, リテラルは $(a^s = v_1 \vee v_2 \vee \dots \vee v_d)$ の形式で, 集合 $\{v_1, v_2, \dots, v_d\}$ をこの属性の定義域の空でない真部分集合としたとき, a^s がこの集合のいずれかの値と等しいとき true をとる。また, 事例 (c_I, A_I) の属性値ベクトルがターム T_i を true にするとき, この事例は T_i にカバーされるという。

この確率的決定リストを, 学習事例 ex_1 から獲得するアルゴリズムを図 10 に示す。このアルゴリズムも,

3.3 節のアルゴリズムと同様, MDL 原理に基づき学習事例と確率的決定リストの記述長の和を短くするリストを探索するものである。記述長については, 本節の最後でまとめて述べることにし, 図 10 について説明する。このアルゴリズムは, 図中“(手続き Pruning)”と示した箇所前後で大きく 2 つの部分に分けられる。前半では, 後述の評価関数を利用して, 短い記述長を達成するようなタームを順次作成し, そのタームにカバーされた学習事例を取り除く過程を繰り返すことで, 暫定的な確率的決定リストを作成する。後半では, 前半で作成されたリストを, より短い記述長を達成するように修正する。以下, 図の注の箇所について補足する。図の(注 1)の箇所の評価値とこれを最大にするリテラルの探索について述べる。この評価値は, 短い記述長を達成する目的で作成した次の評価関数によって求める。2 つのターム T_1 と T_2 があり, 事例集合 S のうちこれらのタームにカバーされる事例集合をそれぞれ $S(T_1)$ と $S(T_2)$ で表す。ただし, $S(T_1) \supseteq S(T_2)$ であるとし, $S(T_1)$ の事例数を $\#S(T_1)$ で, ターム T_1 と事例集合 $S(T_1)$ の記述長をそれぞれ $\ell(T_1)$ と $\ell(S(T_1))$ で表す。このとき評価関数 $G(T_1, T_2)$ は

$$\frac{\ell(T_1) + \ell(S(T_1))}{\#S(T_1)} > \frac{\ell(T_2) + \ell(S(T_2))}{\#S(T_2)}$$

が成立する場合

$$G(T_1, T_2) = (\ell(T_1) + \ell(S(T_1))) - (\ell(T_2) + \ell(S(T_2)) + \ell(S(T_1) - S(T_2)))$$

であり、そうでない場合 0 をとる。そして、ターム T_i^{j-1} に新なりテラル L を追加したターム $T_i^{j-1} \wedge L$ を考え、評価関数 $G(T_i^{j-1}, T_i^{j-1} \wedge L)$ を最大にするリテラルが L_B となる。ここでは、 $\#A$ 個のすべての属性について、連続値属性の場合はあらゆるしきい値、また、離散値属性の場合はあらゆる属性値の組合せのリテラルについて、 $S(T_i^{j-1}) \supseteq S(T_i^{j-1}) \wedge L$ を満たすものすべてを評価する。図の(注2)の箇所の $\Pr(S(T_i^j))$ は、 $S(T_i^j)$ の記述長に文献 12) の方法を用いたことにより定まり、次式で定義される。

$$\Pr(S(T_i^j)) = \frac{\#S^+(T_i^j) + 1}{\#S(T_i^j) + 2}$$

ただし、 $\#S^+(T_i^j)$ は、 $S(T_i^j)$ の事例のうちクラス c が 1 であるものの数を表す。

最後に、確率的決定リストと事例集合の合計記述長を示す。

$$\ell(ex_1, DL) = \log^*(m) + \sum_{i=0}^m (\ell(T_i) + \ell(S(T_i)))$$

$\log^*(\cdot)$ —Rissanen の自然数の記述長¹⁰⁾

m —確率的決定リスト中のターム数

$\ell(T_i)$ —ターム T_i の記述長

$\ell(S(T_i))$ — T_i にカバーされる事例集合の記述長

● 事例集合 S の記述長：

$$\ell(S) = \log(\#S + 1) + \log(\#S C_{\#S+})$$

$\#S$ — S の事例数

$\#S^+ - c = 1$ を満たす S の事例数

$c C_y$ —Combination

● ターム T の記述長：

$$\ell(T) = \sum_{j=1}^{\#A_{used}} (\log(\#A C_j + 1) + \ell(L_j))$$

$\#A$ —属性値ベクトルの要素数

$\#A_{used}$ — T 中のリテラルで使用されている属性数

$\ell(L_j)$ —リテラル L_j の記述長

● リテラル L の記述長 (連続値属性の場合)：

この場合、3 種類形式があるがそれぞれ記述長は次のとおり

$$\ell(a^s < \theta_u) = \log 3 + \ell(\theta_u)$$

$$\ell(\theta_l \leq a^s < \theta_u) = \log 3 + \ell(\theta_l) + \ell(\theta_u)$$

$$(\theta_l \leq a^s) = \log 3 + \ell(\theta_l)$$

ただし $\ell(\theta_l)$ と $\ell(\theta_u)$ は閾値の記述長で、付録 A.3 の閾値と同じ伊藤の方法³⁾ を用いた。

● リテラル L の記述長 (離散値属性の場合)：

$$\log(d - 1) + \log({}_{d-1}C_{d'})$$

d —定義域に含まれる属性値の数

d' —リテラルの右辺にある属性値の数

A.3 回帰木の記述長 $\ell(RT)$

回帰木の木構造は、木を行き掛け順にたどり、非終端ノードを 0、終端ノードを 1 で表すことによりノードの総数のビットで符号化できる⁹⁾。これに、各非終端ノードでの、比較すべき属性の番号を指定する記述長 $\log \#A(\pi)$ としきい値を符号化するのに必要な記述長 $\ell(\theta(\mathbf{nn}_x))$ を加えて、回帰木の記述長 $\ell(RT)$ は次式で表される。

$$\#NN + \#TN + \sum_{\mathbf{nn}_x \in NN} (\ell(\theta(\mathbf{nn}_x)) + \log \#A(\pi))$$

また、しきい値 $\theta(\mathbf{nn}_x)$ は次の伊藤の方法³⁾ で符号化した。 $\theta(\mathbf{nn}_x)$ は、属性値が $[0, 1]$ の範囲にあるため、必ず $(0, 1)$ の範囲にある。この性質を利用し、0.5 を長さ 1 の符合で、その倍の精度の 0.25 と 0.75 を長さ 3 の符合で符号化する。以下同様に、精度が倍になるときに 2 ずつ長い符合で符号化する。

(平成 10 年 8 月 19 日受付)

(平成 11 年 7 月 1 日採録)



神鷹 敏弘 (正会員)

1968 年生。1992 年京都大学情報工学科卒業。1994 年同大学院修士課程修了。同年電子技術総合研究所入所。機械学習とその応用の研究に従事。人工知能学会会員。



元吉 文男 (正会員)

1976 年東京大学大学院理学系研究科物理学専門課程修士課程修了。同年通産省電子技術総合研究所入所。現在同研究所知能情報部帰納推論ラボリーダ。博士 (工学)。数式処理、自然言語処理、論理型言語の研究に従事。著書「LISP で学ぶ認知心理学」(東大出版会, 1983) (共著), 「数式処理システム」(昭晃堂, 1986) (共著), 訳書「プログラムの構造と実行 (上, 下)」(マグローヒル出版, 1989)。