論　文

Technical Papers

相関ルールにもとづく属性生成手法

Attribute Generation Based on Association Rules

寺邊 正大
Masahiro Terabe
（株）三菱総合研究所
Mitsubishi Research Institute, Inc.
terabe@mri.co.jp, http://www.ar.sanken.osaka-u.ac.jp/teraprjp.html

片井 修
Osamu Katai
京都大学大学院情報学研究科
Graduate School of Informatics, Kyoto University.
katai@prec.kyoto-u.ac.jp, http://www.symlab.sys.i.kyoto-u.ac.jp/members/katai/katai.html

椹木 哲夫
Tetsuo Sawaragi
京都大学大学院工学研究科
Graduate School of Engineering, Kyoto University.
sawaragi@prec.kyoto-u.ac.jp, http://isl.prec.kyoto-u.ac.jp/sawaragi/

鷲尾 隆
Takashi Washio
大阪大学産業科学研究所
Institute of Industrial and Scientific Research, Osaka University.
washio@sanken.osaka-u.ac.jp, http://www.ar.sanken.osaka-u.ac.jp/washprjp.html

元田 浩
Hiroshi Motoda
大阪大学産業科学研究所
Institute of Industrial and Scientific Research, Osaka University.
motoda@sanken.osaka-u.ac.jp, http://www.ar.sanken.osaka-u.ac.jp/motoprjp.html

Keywords: data mining, data pre-processing, association rule, decision tree, attribute generation.

Summary

A decision tree is considered to be appropriate (1) if the tree can classify the new data accurately, and (2) if the size of the tree is small. One of the approaches to generate such a good decision tree is to add the new attributes and their attribute values to extend the description of the training data in the data pre-processing stage. Many works on data pre-processing have been done such as attribute generation and attribute selection methods. However, most of them are based on logic programming so that it takes much time for the pre-processing computation, and some of them need a priori knowledge of the data domain. These are disadvantage for the data mining dealing with large volume data. We propose a novel approach that knowledge on the relevance of attributes is generated as association rules from the training data. The new attributes and attribute values are generated from the association rules among attributes. In this paper, we present the proposing method and investigate its feature. The effectiveness of our approach is demonstrated through some experiments.

1. はじめに

近年の計算機性能の向上により，大規模なデータを扱うデータマイニング技術が注目されている．データマイニングにおける分類ルールの知識表現としては，理解しやすさから決定木がよく採用される．一般に決定木の「良さ」は以下の2つの観点から評価される．

サイズ： 決定木のサイズ，すなわち決定木に含まれるノード数が少ないほど（決定木のサイズが小さいほど）分析者にとって理解しやすいものである．また，決定木導出のために準備された訓練データに対する過学習 (overfitting) を避ける効果が期待できる．

分類性能： 決定木が新たに与えられたデータのクラスをより正確に予測し，分類できるほど有用である．

しかしながら，決定木導出のために準備されたデータ（以下，元データとよぶ）から決定木を導出しても「良い」決定木を得ることは困難である場合が多い．その原因の1つとして，元データの属性記述が冗長であったり不足しているなど，そのままでは決定木の導出に適していない場合があげられる．

このような問題点を解決する方法として，新たに属性およびそれらの値を生成して元データに加えることによりデータの記述を変更する手法が提案されている [Bloedorn 98, Lavrăc 98]．ただし，これらの手法は属性あるいは属性間の相関などに関する先験的な知識を必要としたり，属性生成に論理プログラミングを採用しているものが多

い．このため，データマイニングのように大規模データを対象とする場合には，多くの処理時間を要するなどの課題を含んでいる．

これに対して本論文では，属性間の相関についての知識をデータから属性間相関ルールとして自動的に抽出し，この相関ルールをもとに相関のある属性を統合した新規属性を生成して元データに追加することにより，データの記述を拡張するデータ前処理手法について提案する．提案手法は，データが含意する内容に関して分析者が有する知識が少ない段階から，大規模のデータを扱うデータマイニングの枠組みをデータ前処理手法として採用し，

(1) 属性間相関ルールをデータをから自動的に抽出するために，分析者はデータに関する先験的な知識を必要としない．

(2) Agrawalらが提案したAprioriアルゴリズム[Agrawal 94]を提案手法の一部として採用することにより，大規模データを対象とする場合にも現実的な時間での処理を可能にする．

など，優れた特長を備えている．

本論文は，以下のように構成される．まず，**2章**では相関ルールと大規模データから効率的に相関ルールを抽出することが可能なAprioriアルゴリズムについて概説する．次に，**3章**で提案するデータ前処理手法のアルゴリズムについて説明する．さらに**4章**では，テストデータを用いた実験を通して，提案手法の決定木改良への有効性と大規模データへの適用可能性について確認する．そして，**5章**では実験結果をもとに提案手法についての評価と特長に関する考察を行う．

2. 相関ルール

本章では，相関ルールとこれをデータから効率良く抽出することができるAprioriアルゴリズムについて説明する．

相関ルール分析では，各事例がデータ記述の最小単位としてのアイテムの集合であるトランザクション形式のデータを対象にする．そこでは，事例に含まれるアイテム組に見られる共起(co-occurence)パターンが相関ルールとして抽出される[Berry 97]．相関ルールは以下のように記述される．

$$\boldsymbol{B} \Rightarrow \boldsymbol{H} \tag{1}$$

ここで，

$\boldsymbol{B}$: $Body$(相関ルールの条件部)

$\boldsymbol{H}$: $Head$(相関ルールの結論部)

ここで“$Body$”と“$Head$”はアイテムの集合である．この相関ルールは「もし，事例中に$Body$の全てのアイテムが含まれるならば，その事例は$Head$の全てのアイテムも含むことが多い」という内容を意味する．すなわち，事例データにおけるアイテム集合組の共起パターンを表現している．

大量のデータから相関ルールを探索的に抽出するには，従来手法では多くの計算時間が必要であった．これに対して，Agrawalらが提案したAprioriアルゴリズムは，探索の効率化を図ることにより大量データからでも現実的な時間で相関ルールを抽出することを可能にした．Aprioriアルゴリズムでは，相関ルールの探索中に支持度(support value)と確信度(confidence value)という2つの指標を用いて相関ルールの候補を評価する．相関ルールRの支持度$sup(R)$は，以下の式(2)のように定義される．

$$sup(R : \boldsymbol{B} \Rightarrow \boldsymbol{H}) = \frac{n(\boldsymbol{B} \cup \boldsymbol{H})}{N} \tag{2}$$

ここで，

$n(\boldsymbol{B} \cup \boldsymbol{H})$: $\boldsymbol{B}$と$\boldsymbol{H}$の全アイテムを含む事例数

N : 全データ数

この支持度が高いほど，相関ルールが表現するパターンがデータ中に多く現れる．

一方，相関ルールRの確信度$conf(R)$は，以下の式(3)のように定義される．

$$conf(R : \boldsymbol{B} \Rightarrow \boldsymbol{H}) = \frac{n(\boldsymbol{B} \cup \boldsymbol{H})}{n(\boldsymbol{B})} \tag{3}$$

ここで，

$n(\boldsymbol{B} \cup \boldsymbol{H})$: $\boldsymbol{B}$と$\boldsymbol{H}$の全アイテムを含む事例数

$n(\boldsymbol{B})$: $\boldsymbol{B}$の全アイテムを含む事例数

確信度は，相関ルールの条件部と結論部が同じ事例内で現れる共起性を表現する．すなわち相関ルールの確信度が高いほど，このルールがより信頼性の高い結論を導くことを表現している．

Aprioriアルゴリズムでは，これら支持度と確信度について最小支持度(minimum support value)と最小確信度(minimum confidence value)という閾値を設定し，各最小値を充たさない相関ルール候補をその相関性が低いものとして逐次的に評価対象から除外し，探索空間を縮小することにより，従来の相関ルール分析手法よりも計算時間について大幅に効率化している．

ここで，これら各最小値を小さく設定すれば，評価の対象となる相関ルール候補数が多くなり，分析者にとって有用なルールを抽出できないという損失を避けることができるメリットがある．反面，ルール数が多くなり過ぎ有用でないルールも多数混在し，かつ探索コストも増加するというデメリットがある．

3. 提案手法

3·1 概要

提案手法は，決定木を導出するために準備されている元データに対して，決定木を導出する前段階でデータ前

処理を適用することにより，元データの属性記述を改善したデータを新たに生成するものである．この後，新たに生成されたデータをもとにして決定木を導出することになる．

提案手法では，データをもとに説明属性と分類属性の相関を**属性間相関ルール**として抽出する．この属性間相関ルールの抽出を高速に行うために，Apriori アルゴリズムを適用する．**2 章**で説明したように，Apriori アルゴリズムでは統計的尺度を用いた厳密な相関性の判定を行わず，支持度と確信度というデータ中の共起パターンの出現頻度をもとにした指標に用いているが，提案手法で必要な精度の相関判定が十分可能である．

提案手法の処理の概要を以下に示す．

Step 1: 決定木のデータをトランザクションデータ記述に変換する．

Step 2: Apriori アルゴリズムにより属性間相関ルールを抽出する．

Step 3: 抽出された属性間相関ルールをもとに新規属性候補を生成する．

Step 4: 新規属性候補について決定木に対する有用性の評価を行う．

Step 5: 評価基準を満たした新規属性候補を新規属性として元データに加える．

以下では，各ステップの詳細を説明する．

3·2 データ記述の変換 (Step 1)

まず，決定木アルゴリズムのために準備されている元データを相関ルール分析で扱われるトランザクションデータ形式に記述変換する．

元データは，事例とよばれるデータ単位の集合として記述される．事例は，各説明属性の属性値と 1 つの分類属性の属性値（クラス）の組として以下のように記述される．

$$data_i = \langle v_{i,1}, \ldots, v_{i,j}, \ldots, v_{i,n}, c_i \rangle \quad (4)$$

ここで，

$data_i$: i 番目の事例

$v_{i,j}$: 事例 $data_i$ における属性 a_j の属性値

c_i : 事例 $data_i$ のクラス

一方，トランザクションデータでは，データの事例に対応するものがトランザクションであり，各トランザクションはアイテム組として以下のように記述される．

$$trans_i = \langle item_{i,1}, \ldots, item_{i,j}, \ldots, item_{i,n+1} \rangle \quad (5)$$

ここで，

$trans_i$: i 番目の事例トランザクション

$item_{i,j}$: $trans_i$ の j 番目のアイテム

提案手法では，データ中に頻出する属性·属性値組の共起パターンを相関ルールとして取り上げる．よって，属性 “a_j” と属性値 “$v_{i,j}$”の組 “$\langle a_j, v_{i,j} \rangle$”をアイテムとしたデータへと記述を変換する．

例えば表 1 に示されるような元データを，表 2 に示されるトランザクションデータに変換する．

3·3 属性間相関ルールの抽出 (Step 2)

次に，トランザクション形式に変換されたデータを用いて属性・属性値間に含まれる相関関係を示した属性間相関ルールを抽出する．この相関ルール抽出には，**2 章**で説明した Apriori アルゴリズムを用いる．

決定木の導出に有用である説明属性とは，その属性に関する事例中の属性値から事例の属するクラスをより正確に推定することができるもの，すわなち属性値とクラスの間に強い相関関係が存在するものである．そこで，提案手法では属性間相関ルールのうち，とくに以下の条件を充たすものだけを抽出する．

(1) **条件部は説明属性からなるアイテム組**のみを含む．

(2) **結論部は分類属性からなるアイテム**のみを含む．

これらの条件を充たす属性間相関ルールは,「条件部に含まれる全ての〈説明属性, 属性値〉を含む事例は，結論部に含まれるクラスである場合が多い」という事例中の〈説明属性, 属性値〉集合と〈分類属性, クラス〉との相関関係を表現している．よって，この条件部内の〈説明属性, 属性値〉組の集合を統合して 1 つの新規属性として生成すれば，事例の分類に有効な属性になると期待できる．

3·4 新規属性候補の生成 (Step 3)

3·3 節で説明した方法により抽出された属性間相関ルール $R_i : \{R_i | i = 1\ldots, K\}$ をもとにして，相関ルール中の説明属性を統合した新規属性候補 $AN_j : \{AN_j | j = 1, \ldots, M\}$ を生成する．このアルゴリズムの詳細は，表 3 のようになる．

まず，生成された各属性間相関ルール R_i が新規属性候補の基本単位となる．例えば，以下のような属性間相関ルール R_i,

$$R_i : \textbf{if } \langle a_{i_1}, v_{i_1} \rangle \ldots \langle a_{i_l}, v_{i_l} \rangle \textbf{ then } \langle c, c_i \rangle$$
支持度 : $sup(R_i)$，確信度 : $conf(R_i)$ (6)

は，新規属性候補 AN_j の基本単位として以下のような情報を含む．

$$AN_j = \langle \mathcal{A}_j, \mathcal{V}_j, \mathcal{S}_j, \mathcal{C}_j \rangle \quad (7)$$

ここで，

$\mathcal{A}_j = \langle a_{i_1}, \ldots, a_{i_l} \rangle$:属性

$\mathcal{V}_j = \{V_{j_0}, V_{j_1}\}$:属性値

ただし，

表 1 決定木用データ（例）

	天候	温度	湿度	風	クラス
$data_1$	晴	高	低	普通	開催
⋮	⋮	⋮	⋮	⋮	⋮
$data_i$	雨	低	高	強い	中止
⋮	⋮	⋮	⋮	⋮	⋮
$data_m$	曇	低	高	弱い	開催

表 2 表1のデータ記述を変換したトランザクションデータ

$trans_1$	⟨天候=晴⟩,⟨温度=高⟩,⟨湿度=低⟩,⟨風=普通⟩,⟨クラス=開催⟩
⋮	⋮
$trans_i$	⟨天候=雨⟩,⟨温度=低⟩,⟨湿度=高⟩,⟨風=強い⟩,⟨クラス=中止⟩
⋮	⋮
$trans_m$	⟨天候=曇⟩,⟨温度=低⟩,⟨湿度=高⟩,⟨風=弱い⟩,⟨クラス=開催⟩

$$V_{j_1} = true(\langle a_{i_1}, v_{i_1}\rangle, \ldots, \langle a_{i_l}, v_{i_l}\rangle)$$
$$V_{j_0} = \neg V_{j_1}$$
$$\mathcal{S}_j = \{sup(R_i)\}\text{:}R_i \text{の支持度}$$
$$\mathcal{C}_j = \{conf(R_i)\}\text{:}R_i \text{の確信度}$$

新規属性候補 AN_j は，属性間相関ルール R_i の条件部に含まれるアイテム中の属性部分 $a_{i_1}, \ldots, a_{i_l}$ を統合して $\mathcal{A}_j = \langle a_{i_1}, \ldots, a_{i_l}\rangle$ とする．一方，この属性 $\mathcal{A}_j$ の属性値 $\mathcal{V}_j$ については，統合する属性がとる属性値により定義する．統合する属性の属性値が，属性間相関ルール R_i の条件部に含まれる属性値に全て一致する場合，すなわち，$true(\langle a_{i_1}, v_{i_1}\rangle, \ldots, \langle a_{i_l}, v_{i_l}\rangle)$ であるとき，新規属性の属性値を $V_{j_1} = true(\langle a_{i_1}, v_{i_1}\rangle, \ldots, \langle a_{i_l}, v_{i_l}\rangle)$ とする．一方，統合する属性の属性値が1つでも属性値 V_{j_1} 中の属性値以外の値をとる場合，すなわち，$\neg true(\langle a_{i_1}, v_{i_1}\rangle, \ldots, \langle a_{i_l}, v_{i_l}\rangle)$ のとき，$V_{j_0} = \neg V_{j_1}$ とする．以上のように，属性値は，$\mathcal{V}_j = \{V_{j_0}, V_{j_1}\}$ の2値となる．さらに，新規属性候補を構成するもとになった属性間相関ルールの支持度と確信度については，後に新規属性候補の評価を行う際に必要となるので，付随する情報として併せて記憶しておく．

このように生成される新規属性候補 AN_j について，既に生成されている新規属性候補 $AN_k \in \mathcal{AN}$ に属性が一致する，すなわち $\mathcal{A}_k \equiv \mathcal{A}_j\,(j \neq k)$ なるものが存在する場合には，AN_j を AN_k にまとめて1つの候補にする．属性値に $\mathcal{V}_k$ については，AN_k の属性値 $\mathcal{V}_k$ と AN_j の属性値 $\{V_{j_1}\}$ の和をとって，$\mathcal{V}_k = \{V_{k_0}, \ldots, V_{k_h}\} \cup \{V_{j_1}\} = \{V_{k_0}, \ldots, V_{k_h}, V_{k_{h+1}}\}$ とする．ただし V_{k_0} については，$V_{k_0} = \bigwedge_{i=1}^{h+1} \neg V_{k_i}$ とする．また，それぞれの候補で記憶されている支持度と確信度については，$\mathcal{S}_k \leftarrow \mathcal{S}_k \cup \mathcal{S}_j$，$\mathcal{C}_k \leftarrow \mathcal{C}_k \cup \mathcal{C}_j$ として，そのまま記憶しておく．

具体例として，説明属性 A_1（属性値:$V_1 = \{0,1\}$），A_2（属性値：$V_2 = \{0,1\}$），分類属性 C(クラス:$C = \{0,1\}$) について，以下のような2つの属性間相関ルール R_p, R_q が抽出された場合の，提案手法による新規属性候補の生成過程について以下に示す．

$$R_p : \textbf{if } \langle A_1, 0\rangle, \langle A_2, 0\rangle \textbf{ then } \langle C, 0\rangle$$
$$\text{支持度} : sup(R_p),\ \text{確信度} : conf(R_p) \tag{8}$$
$$R_q : \textbf{if } \langle A_1, 1\rangle, \langle A_2, 1\rangle \textbf{ then } \langle C, 0\rangle$$
$$\text{支持度} : sup(R_q),\ \text{確信度} : conf(R_q) \tag{9}$$

これら属性間相関ルール R_p, R_q から次のような新規属性候補 AN_p, AN_q が生成される．

$$AN_p = \langle \mathcal{A}_p, \mathcal{V}_p, \mathcal{S}_p, \mathcal{C}_p\rangle \tag{10}$$
$$AN_q = \langle \mathcal{A}_q, \mathcal{V}_q, \mathcal{S}_q, \mathcal{C}_q\rangle \tag{11}$$

ここで，

$\mathcal{A}_p = \langle A_1, A_2\rangle$:属性

$\mathcal{V}_p = \{V_{p_0}, V_{p_1}\}$:属性値

ただし，

$V_{p_1} = true(\langle A_1, 0\rangle, \langle A_2, 0\rangle)$,

$V_{p_0} = \neg true(\langle A_1, 0\rangle, \langle A_2, 0\rangle)$

$\mathcal{S}_p = \{sup(R_p)\}$, $\mathcal{C}_p = \{conf(R_p)\}$

$\mathcal{A}_q = \langle A_1, A_2\rangle$:属性

$\mathcal{V}_q = \{V_{q_0}, V_{q_1}\}$:属性値

ただし，

$V_{q_1} = true(\langle A_1, 1\rangle, \langle A_2, 1\rangle)$,

$V_{q_0} = \neg true(\langle A_1, 1\rangle, \langle A_2, 1\rangle)$

$\mathcal{S}_q = \{sup(R_q)\}$, $\mathcal{C}_q = \{conf(R_q)\}$

ここで，新規属性候補 AN_p, AN_q は同じ属性を統合したものなので，これらを以下のようにまとめて1つの候補とする．

$$AN_p = \langle \mathcal{A}_p, \mathcal{V}_p, \mathcal{S}_p, \mathcal{C}_p\rangle \tag{12}$$

ここで，

$\mathcal{A}_p = \langle A_1, A_2\rangle$:属性

$\mathcal{V}_p = \{V_{p_0}, V_{p_1}, V_{p_2}\}$:属性値

表 3 新規属性候補生成アルゴリズム:*NewAttributeCand*

属性間相関ルール集合 $\mathcal{R}$:
$\mathcal{R}=\{R_i|i=1,\ldots,K\}$

属性間相関ルール R_i:
R_i:$B_i \rightarrow H_i$
ここで,
$B_i=\{\langle a_{i_1},v_{i_1}\rangle,\ldots,\langle a_{i_l},v_{i_l}\rangle\}$; /*条件部*/
$H_i=\{\langle c,c_i\rangle\}$; /*結論部*/
sup_i; /*支持度*/
$conf_i$; /*確信度*/

新規属性候補集合 $\mathcal{AN}$:
$\mathcal{AN}=\{AN_j|j=1,\ldots,M\}$

新規属性候補 AN_j:
$AN_j=\langle\mathcal{A}_j,\mathcal{V}_j,\mathcal{S}_j,\mathcal{C}_j\rangle$
ここで,
$\mathcal{A}_j$; /*属性（名）*/
$\mathcal{V}_j$; /*属性値*/
$\mathcal{S}_j$; /*元の相関ルールの支持度（集合）*/
$\mathcal{C}_j$; /*元の相関ルールの確信度（集合）*/

アルゴリズム:
NewAttributeCand($\mathcal{R}$,$\mathcal{AN}$){
M=0;
for($i=1;i\leq K;i++$){
/*属性間相関ルール R_i から*/
/*統合属性候補 AN_j を生成*/
/*$AN_j=\langle\mathcal{A}_j,\mathcal{V}_j,\mathcal{S}_j,\mathcal{C}_j\rangle$*/
j=M+1;
$\mathcal{A}_j=\langle a_{i_1},\ldots,a_{i_l}\rangle$;
$\mathcal{V}_j=\{V_{j_0},V_{j_1}\}$;
ただし,
$V_{j_1}=true(\langle a_{i_1},v_{i_1}\rangle,\ldots,\langle a_{i_l},v_{i_l}\rangle)$;
$V_{j_0}=\neg V_{j_1}$;
$\mathcal{S}_j=\{sup(R_i)\}$;
$\mathcal{C}_j=\{conf(R_i)\}$;
/*既存の統合属性候補と属性部分を比較*/
if($\exists AN_k\in\mathcal{AN}|\mathcal{A}_j==\mathcal{A}_k,k=1,\ldots,M$){
/*属性の一致するものが既存*/
$\mathcal{V}_k=\{V_{k_0},\ldots,V_{k_h}\}\cup\{V_{j_1}\}$
$=\{V_{k_0},\ldots,V_{k_h},V_{k_{h+1}}\}$;
ただし, $V_{k_0}=\bigwedge_{i=1}^{h+1}\neg V_{k_i}$;
$\mathcal{S}_k\leftarrow\mathcal{S}_k\cup\mathcal{S}_j$;
$\mathcal{C}_k\leftarrow\mathcal{C}_k\cup\mathcal{C}_j$;
}
else{
/*属性の一致するものが存在しない*/
M++;
$\mathcal{AN}\leftarrow\mathcal{AN}\cup\{AN_j\}$;
}
}
}

ただし,
$V_{p_1}=true(\langle A_1,0\rangle,\langle A_2,0\rangle)$,
$V_{p_2}=true(\langle A_1,1\rangle,\langle A_2,1\rangle)$,
$V_{p_0}=\bigwedge_{i=1}^{2}\neg V_{p_i}$
$\mathcal{S}_p=\{sup(R_p),sup(R_q)\}$
$\mathcal{C}_p=\{conf(R_p),sup(R_q)\}$

3·5 新規属性候補の評価 (Step 4)

生成された新規属性候補の評価には，ID3 などで用いられている情報量利得基準 [Russell 95] を用いる．具体的には，新規属性候補 AN_j を決定木のルートノードとして採用した際の情報量利得の近似値 $Gain(AN_j)$ を評価値とする．ここで，新規属性候補の評価に，訓練データ数などの訓練データ全体に関する基本的な情報以外には，既に属性間相関ルールを抽出する際に計算されている支持度と確信度の値のみしか使わない．すなわち，この新規属性候補の評価の部分については一般的に決定木アルゴリズムで必要とされるようなデータ数の数え上げなどを新たに行う必要はなく，僅かな計算時間で実行可能なものにしている．具体的には，$Gain(AN_j)$ は新規属性候補を生成するもとになった属性間相関ルールの支持度 $\mathcal{S}_j$ と確信度 $\mathcal{C}_j$ を用いて以下のように計算する．

$$
\begin{aligned}
&Gain(AN_j)\\
&=-\sum_{i=1}^{|C|}\frac{n(c_i)}{N}\log_2\frac{n(c_i)}{N}\\
&-\sum_{k=1}^{|\mathcal{V}_j|-1}\frac{sup_k}{conf_k}\left\{-conf_k\log_2 conf_k-p_k\log_2\frac{p_k}{q}\right\}\\
&-\left\{1.0-\sum_{k=1}^{|\mathcal{V}_j|-1}\frac{sup_k}{conf_k}\right\}\{-r_i\log_2 r_i\} \qquad (13)
\end{aligned}
$$

ただし,

$p_k=1.0-conf_k$

$q=|C|-1$

$$r_i=\frac{N_{j_0}(c_i)}{N-N\sum_{k=1}^{|\mathcal{V}_j|-1}(sup_k/conf_k)}$$

ここで,

N : 訓練データ数

$n(c_i)$: クラスが c_i である事例数

$C=\{c_i|c_i$ はデータ中に現れるクラスの種類 $\}$

$N_{j_0}(c_i)$: 子ノード V_{j_0} に属する事例のうち, クラスが c_i である事例数

$N_{j_0}(c_i)=n(c_i)-N_j(c_i)$

$$N_j(c_i)=\sum_{k=1}^{|\mathcal{V}_j|-1}N_{j_k}(c_i)$$

$$N_{j_k}(c_i) = \begin{cases} N \cdot sup_k & (c_k \equiv c_i) \\ \dfrac{N \cdot sup_k(1.0 - conf_k)}{conf_k(|C| - 1)} & (\text{otherwise}) \end{cases}$$

$sup_k \in \mathcal{S}_j : V_{j_k}$ に対応する相関ルールの支持度

$conf_k \in \mathcal{C}_j : V_{j_k}$ に対応する相関ルールの確信度

ここで右辺の第1項は，決定木におけるルートノードで事例を正確に分類するのに必要な情報量である．次に第2項は，ルートノードを新規属性候補 AN_j で分割した際に属性間相関ルールをもとに生成された属性値 $\{V_{j_k}|k = 1,\ldots,h\}$ に相当する枝に続く各子ノードにおいて正確にクラスを分類するのに必要な情報量の近似値である．また第3項は，属性相関ルールから生成された以外の場合に相当する属性値 V_{j_0} に続く子ノードにおいて事例を正確に分類するのに必要な情報量の近似値である(詳細については**付録 A** を参照)．

もし，この情報量利得が正であれば，新規属性候補 AN_j は分類について有益な情報を含んでいるものとして決定木のノードに採用される可能性があるため，これを新規属性として採用する．

3·6 属性の追加 (Step 5)

3·5 節で説明した新規属性候補の評価値 $Gain(AN_j)$ を用いて評価を行う．この評価値が正である場合には，これを新規属性として元データに付加する．

このように提案するデータ前処理手法では，属性間相関ルールをもとにクラスの決定に対して相関のある属性を1つに統合した属性を新たにデータに加えることにより，データの記述を決定木の導出に適したものへと拡張する．

4. 実　　験

4·1 実験に用いたデータと決定木アルゴリズム

提案したデータ前処理手法の決定木改良に対する有効性について確認するために，評価用データを用いた実験を行った．決定木アルゴリズムには，Quinlan の C4.5[Quinlan 93] を使用した．また，以下の評価では，デフォルトの設定で導出して枝刈りを行った後の決定木を用いている．

実験には，UCI の Machine Learning Repository [Blake 98] に収められているデータを用いた．ただし，提案した手法は離散属性にしか適用できないために，この条件を満たすものを選択している．また，訓練データとテストデータが別に準備されていないものについては，データを10分割して交差検定を行った．実験に用いた各データの特徴を以下の表4にまとめる．

実験では，以下の2点について確認を行った．

- 決定木改良に関する効果：決定木のサイズと分類性能により評価する．
- 大規模データへの適用可能性：訓練データ数が増加した場合のデータ前処理に要する計算時間と導出される決定木のサイズと分類性能について評価する．

4·2 実 験 結 果

§1 提案前処理手法による決定木の改良

まず，決定木の改良に関する提案データ前処理手法の効果を確認するために，評価用データを用いて実験を行った．(1) 元データ，(2) 元データに対して提案手法によりデータ前処理を行ったデータ（以下，前処理データと呼ぶ）のそれぞれを訓練データとして決定木を導出し，それら決定木のサイズと分類精度を比較した．なお，以下の実験で相関ルール抽出に用いた最小支持度，最小確信度は，強い相関を表現する相関ルールだけを抽出するために，それぞれ 0.05，0.95 としている．実験結果を表5に示す．

実験結果から確認されたデータ前処理の効果を以下にまとめる．

- 3つのデータ (car，Monk1，tic-tac-toe) で，前処理データから導出された決定木の方が，サイズ，分類精度ともに改善されている．
- 3つのデータ (Monk2，Monk3，votes) では，サイズ，分類精度のいずれかが改善されいる．
- 1つのデータ (census-income) では，各決定木間に違いが見られない．
- 2つのデータ (mushroom, nursery) では，サイズが悪くなっている．

§2 大規模データへの適用可能性

次に，提案手法の大規模データへの適用可能性を確認するための実験を行った．提案手法の処理時間に影響を与える主な要因としては，訓練データ数と Apriori アルゴリズム中で生成される属性間相関ルールの候補数があげられる．ここでは，特に訓練データ数の増加が処理時間に与える影響について調べる．実験に用いるデータは，Monk1 データから任意の事例データを選択したものに対して属性値に5%のノイズを加えることを繰り返し，必要なデータ数分を作成した．データ数は，1万，5万，10万，50万の4通りについて実験を行った．また，実験に用いた計算機は OS が Linux，CPU に Pentium 166 MHz，メインメモリを 128 M を搭載した PC である．

実験から確認された内容を以下にまとめる．

- データ前処理時間は，訓練データ数が増えるにつれて増加するが，その程度はおよそ訓練データ数に比例する．また，50万件のデータでも2分弱でデータ前処理を終えることができる．
- データ前処理の効果は，訓練データ数が増加しても劣化しない．

表 4 実験に用いた評価用データ

データ名	訓練データ数	テストデータ数	説明属性数	クラス数
census-income	32562	交差検定 (10 分割)	8	2
car	1728	交差検定 (10 分割)	6	4
Monk1	124	432	6	2
Monk2	169	432	6	2
Monk3	122	432	6	2
mushroom	8124	交差検定 (10 分割)	22	2
nursery	12960	交差検定 (10 分割)	8	5
tic-tac-toe	958	交差検定 (10 分割)	9	2
votes	435	交差検定 (10 分割)	16	2

表 5 データ前処理の決定木サイズに対する効果

データ名	決定木（元データ）			決定木（前処理データ）		
	サイズ	誤分類率 (%)	属性数	サイズ	誤分類率 (%)	属性数
census-income	221	16.9	8	221	16.9	31
car	173.2	6.8	6	88.7	4.5	18
Monk1	18	24.3	6	8	0.0	22
Monk2	31	35.0	6	35	24.5	19
Monk3	12	2.8	6	11	7.2	27
mushroom	32.0	0.0	22	32.6	0.0	42
nursery	508.3	2.9	8	571.8	3.0	18
tic-tac-toe	139.3	15.2	9	23.7	0.3	25
votes	15.4	3.7	16	13.8	3.9	186
平均	115.0	10.8	8.7	100.6	6.0	38.8

表 6 データ数とデータ前処理時間

データ数	決定木（元データ）		決定木（前処理データ）		前処理に要する計算時間 (sec.)
	サイズ	誤分類率 (%)	サイズ	誤分類率 (%)	
10,000	90	21.3	9	5.0	4
50,000	79	17.1	12	4.9	13
100,000	79	14.6	12	5.0	22
500,000	90	19.7	13	5.0	114

5. 考　　察

5·1 提案手法の基本的特性

まず，提案したデータ前処理手法の基本的な特性についての理解を深めるために，Monk's データ [Thrun 91] を例にとって導出された決定木の詳細を見ながら議論する．Monk's データは，人工的に作成された分類問題であり，Monk1，Monk2，Monk3 の 3 種類のデータがある．各データは，それぞれ $a1,\ldots,a6$ の 6 つの説明属性と 2 つのクラス $Class=\{0,1\}$ を持つ [Quinlan 93]．ただし，各データごとに説明属性間に含まれる相関関係が異なる．各属性間相関ルールは以下のとおりである．

Monk1 : $(a1=a2)$ or $(a5=1) \Rightarrow Class=1$

Monk2 : $\{a1=1,a2=1,a3=1,a4=1,a5=1,$ $a6=1\}$ のうち 2 つのみが充たされる $\Rightarrow Class=1$

Monk3 : $(a5=3$ and $a4=1)$ or $(a5\neq 4$ and $a2\neq 3) \Rightarrow Class=1$

また，Monk3 データでは，クラスに 5%のノイズを含む．以下では，各データごとに導出された決定木をもとに議論する．

Monk1 : Monk1 の訓練データについてデータ前処理を行った結果，16 の新規属性 $a7,\ldots,a22$ が生成された．元データ，および前処理データから導出された各決定木を図 1 に示す．前処理データから導出された決定木では，ルートノードに新規属性 $a7(=\langle a1,a2\rangle)$ が採用され，さらに子ノードのレベルでも $a10(=\langle a3,a5\rangle)$ が採用されている．これらノードに採用された新規属性 $a7,a10$ は説明属性間の相関をよく表現したものであり，決定木アルゴリズムの中でも分類に有効な属性としてノードに採用されていることが分かる．また，サイズ，分類精度ともに向上して

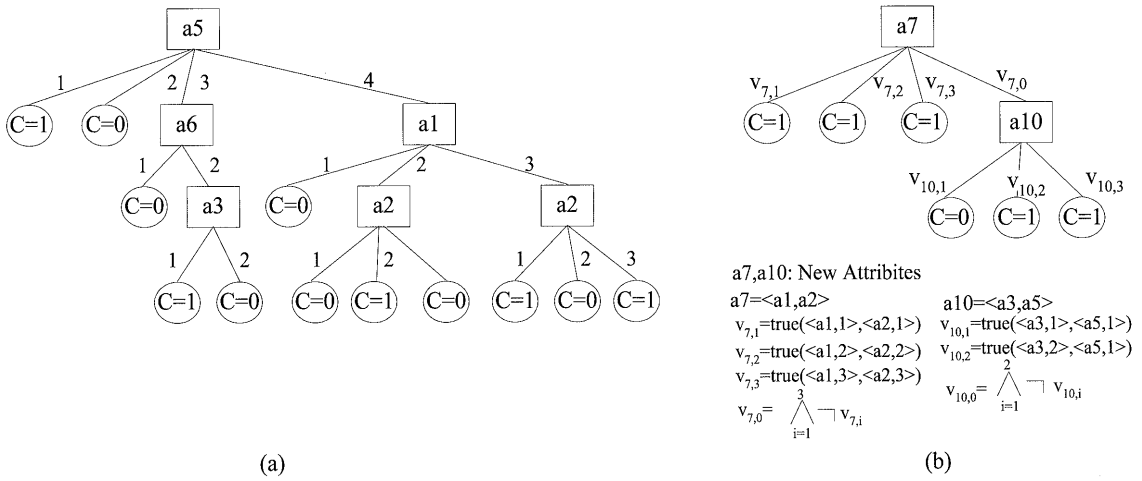

図 1 Monk1 データから導出された決定木：(a) 元データ，(b) 前処理データ

おり，データ前処理とこれにより生成される新規属性が，導出される決定木の質に良い影響を与えていることが確認された．

Monk2： データ前処理によって，13の新規属性 $a7,\ldots,a19$ が生成された．この例では，提案データ前処理手法の特長と限界が確認できる．提案手法では，元データに含まれる属性を連言で統合したものしか生成できず，選言で統合したものは生成できない．よって，上に説明した Monk2 に含まれる属性間相関は，$\langle a1,a2,a3,a4,a5,a6\rangle = \{\langle 1,1,0,0,0,0\rangle,\ldots,\langle 0,0,0,0,1,1\rangle\}$ というように全ての属性の連言の集合でしか正しく表現できない．しかしながら，このような多くの属性の連言からなるパターンは事例としてデータ中に頻出しない．このため，出現頻度の低い共起パターンを抽出しない Apriori アルゴリズムの特性から，これらパターンは属性間相関ルールとして抽出されにくいので，結果として新規属性として生成されない．提案手法では，これに代わって連言に含まれるべき属性の部分集合からなる $a15 = \langle a2,a4,a6\rangle$ などを生成しており，これが決定木のノードとして採用されることにより，分類性能の改良に貢献している．

Monk3： データ前処理によって21の新規属性 $a7,\ldots,a27$ が生成された．データ前処理した訓練データから導出された決定木では，$a13(=\langle a4,a5\rangle)$ のように属性間に含まれる相関関係をよく記述している新規属性がノードに採用されており，サイズを小さくしているが，分類性能は悪化している．

以上のように，提案したデータ前処理手法では，少数の元データに含まれる属性の相関については，属性間相関ルールとしてよく抽出し，決定木の改良に有用な属性を生成することができる．一方，多くの属性の相関によるものについては，属性間相関ルールとして抽出，あるいは新規属性として生成しにくいという限界もある．

5·2 決定木の改良

提案手法によってデータ前処理を行うことにより，多くの場合において決定木のサイズや分類精度が改良された．しかしながら，census-income，mushroom，nursery のデータでは改良が見られなかった．実験において見られた特徴として，これらのデータから導出された決定木に，新規属性が決定木のルートに近い上位ノードで採用されなかったことがあげられる．この理由として，データ前処理によって生成された新規属性よりも元から訓練データに含まれていた属性の方が分類に有効な属性を多く含んでいたことが考えられる．

5·3 最小支持度と最小確信度の設定

提案手法では，属性間相関ルール抽出時の最小支持度と最小確信度の設定が生成される新規属性および属性値の内容に影響を与える．そこで，先の実験で用いた Monk1 データと car データを用いて，これら最小支持度と最小確信度を変化させて実験を行った．その結果を，それぞれ表7，表8に示す．ここでの決定木の深さとは，決定木中のルートノードから葉ノードに至るまでに通るノード数の最大値である．

実験結果からも確認できるように，最小支持度や最小確信度を小さく設定すると生成される属性間相関ルールが多くなる．その結果，生成される新規属性数および1新規属性あたりの属性値数も増加する．このため，設定値を小さくした場合に導出される決定木は，これら新規属性をノードとして採用した結果として，1つのノードから多くの枝の出た幅の広いものとなり，全体的なサイズも大きくなっている．さらに導出された決定木は，数少ない事例に対して1つの枝を生成したようなものとなり，誤分類率も増加している．

このように提案手法では，最小支持度と最小確信度を小さく設定すると，分類精度が悪くなったり，サイズが極端に大きくなるため，実験結果からは，最小支持度を0.9，最小確信度を0.01程度に設定するのが良いことが

表 7 提案手法における最小支持度・最小確信度の設定と導出された決定木 (Monk1 データ)

提案手法					決定木		
設定値		属性生成結果			(前処理データ)		
最小支持度	最小確信度	新規属性数 (a)	新規属性値数 (b)	平均属性値数 (b/a)	誤分類率 (%)	サイズ	深さ
0.1	0.9	4	11	2.8	0.0	13	4
	0.7	7	22	3.1	4.2	14	3
	0.5	12	43	3.6	4.9	17	3
0.05	0.9	16	55	3.4	0.0	8	2
	0.7	24	97	4.0	0.0	12	2
	0.5	30	140	4.7	0.0	19	2
0.01	0.9	42	469	11.2	0.0	8	2
	0.7	51	570	11.2	0.0	12	2
	0.5	56	639	11.4	3.1	31	2

表 8 提案手法における最小支持度・最小確信度の設定と導出された決定木 (car データ)

提案手法					決定木		
設定値		属性生成結果			(前処理データ)		
最小支持度	最小確信度	新規属性数 (a)	新規属性値数 (b)	平均属性値数 (b/a)	誤分類率 (%)	サイズ	深さ
0.1	0.9	3	14	4.7	4.8	114.8	5
	0.7	3	14	4.7	4.9	115.7	5
	0.5	3	14	4.7	5.0	116.6	5
0.05	0.9	12	52	4.3	4.5	87.7	7
	0.7	15	64	4.3	3.6	92.6	9
	0.5	15	66	4.4	3.8	129.0	9
0.01	0.9	32	335	10.5	3.1	107.5	8
	0.7	35	370	10.6	2.4	172.9	10
	0.5	35	442	12.6	1.9	365.7	11
0.005	0.9	47	1398	29.7	2.2	379.3	10
	0.7	50	1463	29.3	2.3	497.6	12
	0.5	50	1579	31.6	2.4	706.7	12

確認された．特に最小支持度については，導出される決定木のサイズの問題から，小さく設定しすぎないことが必要である．

5·4 大規模データへの適用可能性

訓練データ数が大量である場合にもデータ前処理の効果は変化がなく良好であった．さらにデータ数が増えるにつれてデータ前処理に要する計算時間は増加するが，その率はほぼ一定である．これら実験から確認された特徴は，訓練データ数が多い大規模データを扱うデータマイニングでも提案手法が適用可能であることを示している．計算時間に影響を与えるもう 1 つの要因である属性間相関ルール候補数については，最小支持度と最小確信度の設定に依存する．**5·3 節**の議論と関連して，適切な最小支持度，最小確信度を設定することが必要である．

また，データ前処理により生成した新規属性を加えた結果として属性数が増加するため，決定木の導出に必要な計算時間もデータ前処理を行わない場合に比べると増加している．

6. 関 連 研 究

先験知識をもとに論理プログラミングにより新たに属性生成を行う方法については，Lavrăc[Lavrăc 98] らによる研究があり，属性に関する先験知識がある領域では効果的に新規属性を生成できるという結果が得られている．しかしながら，データマイニングで扱われるような先験知識の豊富でない大規模なデータを対象とした場合には適用困難な可能性がある．

Liu ら [Liu 98] は，属性間の相関ルールをもとにして分類子を導出するアルゴリズムを提案し，C4.5 よりも良い分類性能が出ることを示しているが，導出される分類子は決定木の形式で提示されるものではない．また，訓練データの事例数と計算時間の関係について言及していない．

7. お わ り に

本論文では，決定木を改良するためのデータ前処理手法について提案し，その特長について評価用データを用いた実験を通して確認した．提案手法は，データ中の属性に関する先験的な知識を必要とせず，代わりに訓練デー

タから抽出された属性間の相関ルールをもとに相関のある属性を統合した新規属性を生成して訓練データに付加することにより，データの記述を変更する．また，相関ルール抽出にAprioriアルゴリズムを採用することにより少ない計算時間での前処理を実現しているため大規模データにも適用可能である．これら本提案手法の特長は，データマイニングに適したものである．

さらなる研究の方向として，以下のような内容について現在検討を行っている．

- **5·3節**で実験結果をもとに考察したように，Aprioriアルゴリズムでは，最小支持度および最小確信度と生成される属性間相関ルールの数の間にトレードオフがあり，その結果は提案手法の属性生成に影響を与える．これらの値の設定方法や新規属性，属性値の評価について，さらに検討を加える必要がある．例えば，各属性間相関ルールの支持度や式(13)で定義した情報量利得を用いて，生成される属性や属性値を評価することが考えられる．
- 提案手法をC4.5など決定木アルゴリズムの一部とすることにより，さらに良質な決定木の導出，アルゴリズムの効率が期待できる．例えば，C4.5でオプションとして採用されている属性値グルーピング[Quinlan 93]のように決定木を導出する中で逐次新規属性を生成することによる決定木の改良，アルゴリズム効率化に関する効果について確認したいと考えている．

謝　辞

本論文に対して有益なご助言を頂いた査読者の方々に感謝します．

◇参考文献◇

[Agrawal 94] Agrawal, R. and Srikant, R.: Fast Algorithms for mining association rules, *Proceedings of the 20th VLDB Conference*, pp.487–499 (1994).

[Berry 97] Berry, M.J.A., et al.: *Data Mining Techniques For Marketing, Sales, and Customer Support*, Wiley (1997).

[Blake 98] Blake, C., Keogh, E. and Merz, C.J.: UCI Repository of machine learning databases, http://www.ics.uci.edu/ mlearn/MLRepository.html, Irvine, CA: University of California, Department of Information and Computer Science (1998).

[Bloedorn 98] Bloedorn, E., et al.: Data-Driven Constructive Induction: A Method and Its Application, *IEEE Intelligent Systems & their Applications*, Vol.13, No.2, pp.30–37 (1998).

[Lavrăc 98] Lavrăc, N., et al.: A relevancy filter for constructive induction, *IEEE Intelligent Systems & their Applications*, Vol.13, No.2, pp.50–56 (1998).

[Liu 98] Liu, B., Hsu, W., and Ma, Y.: Integrating Classification and Association Rule Mining, *The 4th Conference on Knowledge Discovery and Data Mining*, pp.80–86 (1998).

[Quinlan 93] Quinlan, R.: *C4.5: Programs for Machine Learning*, Morgan Kaufmann (1993).

[Russell 95] Russell, S. and Norvig, P.: *Artificial Intelligence, A Modern Approach*, Prentice-Hall (1995).

[Thrun 91] Thrun, S., et al.: The MONK's Problems: A Performance Comparison of Different Learning Algorithms, *Technical Report of Carnegie Mellon University*, CMU-CS-91-197 (1991).

〔担当委員：阿久津達也〕

1999年5月11日　受理

◇付　　録◇

A. 新規属性候補の評価

新規属性候補 AN_j(属性 $\mathcal{A}_j$，属性値：$\mathcal{V}_j=\{V_{j_1},\ldots,V_{j_h},V_{j_0}\}$)の評価式(13)の導出過程について補足する．新規属性候補は，これを決定木のルートノードに採用した場合の情報量利得について以下のように計算される近似値 $Gain(AN_j)$ により評価される．

まず，ルートノード $root$ でクラスを正確に言い当てるのに必要な情報量 G_{root} は，各クラスに属する事例数 $n(c_i)$ を用いて，以下のように計算することができる．

$$G_{root}=-\sum_{i=1}^{|C|}\frac{n(c_i)}{N}\log_2\frac{n(c_i)}{N} \tag{A.1}$$

ここで，

N：訓練データ数

$C=\{c_i|c_i$ はデータ中に現れるクラスの種類$\}$

$n(c_i)$：クラスが c_i である事例数

次に，新規属性候補の属性値 $\mathcal{V}_j$ のうち，属性間相関ルール R から定義された属性値 $\{V_{j_k}|k=1,\ldots,h\}$ に対応する枝に続く各子ノード $child_k$ でクラスを正確に言い当てるのに必要な情報量は，以下のように計算することができる．各子ノードに属する事例数 $N^1_{j_k}$ は，属性間相関ルールの支持度 sup_k，$conf_k$ の定義から，元の属性間相関ルール R_k の条件部が一致する事例数に等しいので，これを用いて以下のように計算できる．

$$N^1_{j_k}=N\cdot\frac{sup_k}{conf_k} \tag{A.2}$$

さらに，このうち R_k の結論部のクラスが c_k である事例数 $N^2_{j_k}$ についても，属性間相関ルールの確信度 sup_k の定義から，以下のように計算できる．

$$N^2_{j_k}=N\cdot sup_k \tag{A.3}$$

残りのクラスに属する事例数は，属性間相関ルールの情報からは計算することができない．そこで，クラスを言い当てるのに必要な情報量が最も大きくなる場合を考える．必要な情報量が最も大きくなるのは，各クラスに属する事例数が等しい場合である．このときの事例数 $N^3_{j_k}$ は，以下のように計算することができる．

$$\begin{aligned}N^3_{j_k}&=\frac{N^1_{j_k}-N^2_{j_k}}{|C|-1}\\&=\frac{N\cdot sup_k(1.0-conf_k)}{conf_k(|C|-1)}\end{aligned} \tag{A.4}$$

以上から，子ノード $child_k$ でクラスを言い当てるのに必要な情報量の近似値 G_{child_k} は，以下のように見積もることができる．

$$\begin{aligned}&G_{child_k}\\&=\frac{N^1_{j_k}}{N}\left\{-\left(\frac{N^2_{j_k}}{N^1_{j_k}}\right)\log_2\left(\frac{N^2_{j_k}}{N^1_{j_k}}\right)\right.\\&\quad\left.-(|C|-1)\cdot\left(\frac{N^3_{j_k}}{N^1_{j_k}}\right)\log_2\left(\frac{N^3_{j_k}}{N^1_{j_k}}\right)\right\}\end{aligned}$$

$$= \frac{sup_k}{conf_k}\left\{-conf_k \log_2 conf_k - p_k \log_2 \frac{p_k}{q}\right\} \quad \text{(A.5)}$$

ただし，

$p_k = 1.0 - conf_k$

$q = |C| - 1$

属性間相関ルールから生成された属性値以外の値をとる場合に準備されている属性値 V_{j_0} に続く子ノード $child_0$ に属する事例数 $N^1_{j_0}$ は，全事例数 N から他の子ノード $child_k$ に属する事例数を引いたものとして，以下のように計算できる．

$$N^1_{j_0} = N - \sum_{k=1}^{|\mathcal{V}_j|-1} N^1_{j_k}$$

$$= N\left\{1.0 - \sum_{k=1}^{|\mathcal{V}_j|-1} \frac{sup_k}{conf_k}\right\} \quad \text{(A.6)}$$

ここで，属性間相関ルールから生成された属性値で見積もった事例数をもとにして，$child_0$ に属する事例の各クラスごとの事例数を見積もる．各クラスごと c_i の事例数 $N_{j_0}(c_i)$ は，(A.3)，(A.4) のように子ノード $child_k$ において属すると見積もった結果をもとにして，以下のように見積もることができる．

$$N_{j_0}(c_i) = n(c_i) - N_j(c_i) \quad \text{(A.7)}$$

ここで，

$$N_j(c_i) = \sum_{k=1}^{|\mathcal{V}_j|-1} N_{j_k}(c_i)$$

$$N_{j_k}(c_i) = \begin{cases} N \cdot sup_k & (c_k \equiv c_i) \\ \dfrac{N \cdot sup_k(1.0 - conf_k)}{conf_k(|C|-1)} & (\text{otherwise}) \end{cases}$$

$N_j(c_i)$: $child_0$以外の子ノードに属するデータのうちクラスが c_iである事例数

$N_{j_k}(c_i)$: 子ノード $child_k$に属するデータのうちクラスが c_iである事例数

以上から，子ノード $child_0$ でクラスを言い当てるのに必要な情報量の近似値 G_{child_0} は，以下のように見積もることができる．

$$G_{child_0}$$

$$= \frac{N^1_{j_0}}{N}\sum_{i=1}^{|C|}\left\{-\left(\frac{N_{j_0}(c_i)}{N^1_{j_0}}\right)\log_2\left(\frac{N_{j_0}(c_i)}{N^1_{j_0}}\right)\right\}$$

$$= \left\{1.0 - \sum_{k=1}^{|\mathcal{V}_j|-1}\frac{sup_k}{conf_k}\right\}\sum_{i=1}^{|C|}\{-r_i \log_2 r_i\} \quad \text{(A.8)}$$

ただし，

$$r_i = \frac{N_{j_0}(c_i)}{N - N\sum_{k=1}^{|\mathcal{V}_j|-1}(sup_k/conf_k)}$$

よって，新規属性候補 AN_j を決定木のルートノードに採用した場合の情報量利得の近似値 $Gain(AN_j)$ は，式 (A.1)，(A.5)，(A.8) より，

$$Gain(AN_j) = G_{root} - \sum_{k=1}^{|\mathcal{V}_j|-1} G_{child_k} - G_{child_0} \quad \text{(A.9)}$$

のように見積もることができ，これは式 (13) のようになる．

著 者 紹 介

寺邊 正大(正会員)

1970 年生まれ．1993 年京都大学工学部精密工学科卒業．1995 年京都大学大学院工学研究科精密工学専攻修士課程修了．同年（株）三菱総合研究所入社．現在，総合安全研究センターで，データマイニング，機械学習，マルチエージェント技術の大規模システム運転支援・診断への適用，およびヒューマン・マシンシステムの設計に関する研究に従事．計測自動制御学会，日本ファジィ学会，日本原子力学会，AAAI，各会員．

片井 修(正会員)

1969 年 3 月京都大学工学部機械工学科卒業．同大学機械工学第二専攻修士・博士課程を経て 1974 年同大学助手（精密工学）．1983 年同助教授．1994 年同教授，1996 年同大学院工学研究科教授（精密工学専攻）．1998 年同大学院情報学研究科設立とともに移行（システム科学専攻）．その間，1980-81 年フランス INRIA （国立情報処理自動化研究所）客員研究員．主としてシステムの知能化に関する研究に従事．本学会の KBS,HICG,HIDSN 研究会の研究連絡委員を歴任．計測自動制御学会の知能工学部会の運営に携わる．また，創刊号より日本ファジィ学会の編集に携わり現在同会誌編集委員長．計測自動制御学会論文賞（1989,1991 年度），同著述賞（1992 年度）など受賞．

椹木 哲夫(正会員)

1983 年京都大学大学院工学研究科精密工学専攻修士課程修了．1986 年同博士後期課程指導認定退学．同年より京都大学工学部精密工学教室助手．1994 年同大学大学院工学研究科精密工学専攻助教授．現在に至る．1991 ～ 92 年米国スタンフォード大学客員研究員．現在，人間-機械共存環境下での協調システムの設計と知的支援に関する研究に従事．京都大学工学博士．計測自動制御学会学術奨励賞 ('85)，論文賞 ('89,'91)，著述賞 ('92) 等受賞．計測自動制御学会，日本機械学会，日本ファジィ学会，航空運航システム研究会，IEEE，各会員．

鷲尾 隆(正会員)は，前掲(Vol.15，No.1，p.186)参照．

元田 浩(正会員)は，前掲(Vol.15，No.1，p.186)参照．