基于粗糙集的多变量决策树构造方法*

苗夺谦 王 珏

(中国科学院自动化研究所人工智能实验室 北京 100080)

摘要 本文利用粗糙集理论中条件属性相对于决策属性的核,解决多变量检验中属性的选择问题.另外,定义了2个等价关系相对泛化的概念,并将它用于解决多变量检验的构造问题.通过一个例子,对本文提出的多变量决策树方法与著名的单变量决策树(ID3)方法进行了比较,结果表明前者比后者更简单.同时,对几种多变量决策树方法做了初步的对比分析.

关键词 粗糙集,单变量决策树,多变量决策树,归纳学习,属性的相对核.

中图分类号 TP18

目前,大多数归纳学习系统得到的结果是一棵决策树.决策树分类器有2个重要的优点:首先,决策树中的检验是沿着树的分枝序贯进行的.这样,只需对某一决策中所用到的属性进行评价.其次,决策树为确定某一事例类别的序贯决策方法提供了清晰的陈述.一棵小的具有简单检验的决策树是最受人欢迎的,因为人们很容易理解由它导出的决策.

大多数决策树被限制在每个结点上只检验单个属性.这样的决策树被称之为单变量决策树.其中著名的系统有ID3[1],AQ11[2],ASSISTANT[3]和GREEDY3&GROVE[4].这一限制使得对很多复杂概念的表达变得困难或无法表达.这种表示上的限制主要是以2种形式表现的:一棵决策树中子树的重复[4]和有些属性在一棵决策树中的某一路径上被多次检验.

为了克服这一限制,人们提出了多变量归纳学习系统,即在树的结点上可以同时检验多个属性.这种系统能够产生新的、更相关的属性,以及修改或去掉初始提供的不相关的属性.对于重复子树问题,Pagallo G[4]等人通过构造属性的Boolean组合,已经说明可以改进决策树的准确性,减小了训练所需的事例数和决策树的大小.Brodley C E[5]等人采用初始属性的线性组合来构造多变量决策树.这一方法的缺点是必须把符号属性转化成数值属性.

粗糙集理论是由Pawlak Z于1982年提出的.[6]这一理论从新的视角对知识进行了定义,把知识看作是关于论域的划分,认为知识是有粒度的.引入代数学中的等价关系来讨论知识.该理论主要用于知识的约简及知识相依性的分析.因此,可以作为机器学习和复杂数据分析的工具.

构造多变量决策树的关键问题是多变量检验的构造问题.它涉及2个方面:一是选择什

* 本文研究得到国家863高科技项目基金资助.作者苗夺谦,1964年生,讲师,主要研究领域为粗糙集理论,机器学习,数据库中的知识发现等.王珏,1948年生,研究员,主要研究领域为人工智能.

本文通讯联系人:苗夺谦,北京100080,中国科学院自动化研究所人工智能实验室

本文1996-07-08收到修改稿

么样的初始属性包含在多变量检验中？二是如何利用选择的属性来构造多变量检验？本文提出了一种基于粗糙集的构造多变量决策树的方法.本文第1节利用粗糙集理论中相对核的概念,解决了多变量检验中初始属性的选择问题.定义了2个等价关系相对泛化的概念,并将它用于构造多变量检验.第2节描述了构造多变量决策树的算法.第3节通过一个例子,对多变量决策树和单变量决策树ID3进行了比较.同时,对几种多变量决策树方法做了初步对比分析.第4节给出了本文的结论及进一步研究的问题.

1 属性的选择及相对泛化的定义

文献[4,5]分别利用选择的初始属性的线性组合和Boolean组合来形成多变量检验.本文将使用选择的初始属性的合取(不是简单的合取,而是由其导出的泛化)作为多变量检验.本节我们要回答引言中提出的构造多变量检验涉及的2个问题,即初始属性的选择和多变量检验的构造.

1.1 属性的选择

应该选择哪些初始属性来构造多变量检验呢?人们希望删除带噪声的和不相关的属性,减少检验中的属性数目,从而增加决策的可理解性.对于大多数数据集来说,要想通过尝试属性的每一种组合来找出最佳属性组合是不可能的,因为属性的各种可能的组合数是依属性数指数增长的.所以,在属性的选择中,必须使用某种启发式.粗糙集理论可以用于多变量检验中属性的选择.

由于属性与等价关系之间存在着一一对应关系,所以,这2个概念是等价的,可以相互替换.因此,在下面的讨论中,本文将不区分属性与等价关系.

设U是感兴趣的对象组成的有限集合,称为论域.R是定义在U上的一个等价关系.U/R表示R在U上导出的划分,$[x]_R$表示包含x的R的等价类,$x\in U$.在粗糙集理论中[6],将序对(U,R)称为一个近似空间.任何子集$X\subseteq U$,称为一个概念.对每个概念X可定义下、上近似如下:

$$\underline{R}X=\bigcup\{x\in U:[x]_R\subseteq X\}\quad \overline{R}X=\bigcup\{x\in U:[x]_R\cap X\neq\varnothing\}$$

$\underline{R}X$是由U中那些在现有知识R下肯定属于概念X的元素组成的集合;$\overline{R}X$是可能属于概念X的元素组成的集合.对于U上的2个等价关系P,Q,Q的P-正区域定义为

$$POS_P(Q)=\bigcup_{X\in U/Q}\underline{P}X \tag{1}$$

$POS_P(Q)$是U中所有那些通过知识P被肯定地分作U/Q的类的元素组成的集合.

设U是一个论域,P和Q是定义在U上的2个等价关系族.称一个等价关系$R\in P$是Q-不必要的(或多余的),如果式

$$POS_{IND(P)}(IND(Q))=POS_{IND(P-\{R\})}(IND(Q)) \tag{2}$$

成立.否则,R在P中是Q-必要的.其中$IND(P)=\bigcap P$(所有属于P的等价关系的交)也是一个等价关系,并且称为P上的一个不可区分(indiscernibility)关系.

P中所有Q-必要的等价关系组成的集合,称为P的Q-核,记作$CORE_Q(P)$.

当P与Q分别表示信息系统的条件属性和决策属性时,若一个属性$R\in P$是Q-不必要的,则从P中去掉属性R不会改变原来信息系统的决策.而去掉P的Q-核中的属性将改变

原信息系统的决策.所以,P 的 Q-核中的属性对于决策来说是至关重要的.我们将选择相对核中的属性作为构造多变量检验的属性.

1.2 相对泛化的定义

如何用所选的属性来构造多变量检验呢?通过分析,我们知道用所选属性的简单合取作为多变量检验,可能会导致对数据的过拟合问题.为此,我们定义了一个等价关系相对于另一个等价关系泛化的概念.

定义 1. 设 P 和 Q 是 U 上的 2 个等价关系族,且

$$U/IND(P)=\{X_1,X_2,\ldots,X_n\}\quad U/IND(Q)=\{Y_1,Y_2,\ldots,Y_m\}$$

令

$$Z_i=\bigcup_{X_j\in U/IND(P)}\{X_j:X_j\subseteq Y_i\}\quad i=1,2,\ldots,m. \tag{3}$$

$$Z_{m+1}=\bigcup_{X_j\in U/IND(P)}\{X_j:X_j\not\subset Y_i,\forall\, i\} \tag{4}$$

则称 $\{Z_1,Z_2,\ldots,Z_{m+1}\}$ 在 U 上确定的等价关系为 P 相对于 Q 的泛化,记作 $GEN_Q(P)$.

上述定义的合理性可通过如下命题说明.

命题 1. $\{Z_1,Z_2,\ldots,Z_{m+1}\}$ 构成了 U 上的一个划分,其中 $Z_i,i=1,2,\ldots,m+1$,由(3)(4)式定义.

证明:从 Z_i 的定义可知,$\bigcup_{i=1}^{m+1}Z_i=\bigcup_{i=1}^{n}X_i=U$.下证 $Z_i\cap Z_j=\varnothing,i\neq j,i,j=1,2,\ldots,m+1$.由定义显然有 $Z_i\cap Z_{m+1}=\varnothing$,对任意的 $i=1,2,\ldots,m$ 都成立.

对于 $i,j=1,2,\ldots,m$ 的情况,用反证法证明.假设 $Z_i\cap Z_j\neq\varnothing,i\neq j,i,j=1,2,\ldots,m$.则至少存在一个 $x\in U$,使得 $x\in Z_i\cap Z_j$.

$\Rightarrow x\in Z_i$ 且 $x\in Z_j$,从而有 $x\in Y_i$ 且 $x\in Y_j$.

$\Rightarrow x\in Y_i\cap Y_j$,所以 $Y_i\cap Y_j\neq\varnothing,i\neq j$.

这与 $\{Y_1,Y_2,\ldots,Y_m\}$ 是 U 的划分矛盾!所以,$\{Z_1,Z_2,\ldots,Z_{m+1}\}$ 构成了 U 上的一个划分.我们知道,U 上的划分与其上的等价关系是一一对应的.因此,这一划分唯一地确定了 U 上的一个等价关系.

相对泛化的概念将用于构造多变量检验.

2 多变量决策树的构造算法

多变量决策树的构造和单变量决策树的构造在很多方面是相同的.这 2 种算法都是从标有类别信息的事例中,采用递归的方法来构造决策树的.

本算法是从信息系统表示的数据中导出决策树.形式上,一个信息系统 S 定义为一个四元组 $S=(U,A,V,f)$.其中 U 为论域;A 为所有属性的集合,它进一步可分为条件属性 C 和决策属性 D,$C\cap D=\varnothing$;$V=\bigcup_{P\in A}V_P$,V_P 是属性 P 的值域;$f:U\times A\to V$,称为一个信息函数.一个自顶向下的决策树算法是,首先根据某种划分度量准则选择最佳检验,然后,用选择的检验去划分训练集,并且,相应于该检验的每一个结果产生一个分枝.这一算法递归地应用到该检验导出的每一个分类上.如果某一分类中的所有事例都来自于一个类别,那么,就产生一个标有该类别名的叶结点.

在决策树的构造过程中,人们希望在每个结点上都选择能把事例划分到它们类中的最

佳检验.单变量决策树和多变量决策树的差别在于前者的检验是基于单个属性的,而后者的检验是基于一个或多个属性的.单变量决策树中通常使用的划分度量准则是熵或不纯(impurity)测度.[1,7]

本文使用属性的区分能力作为划分度量准则.根据相对核的定义,我们知道条件属性集相对于决策属性集的核中的属性对于制定决策来说是至关重要的.利用第1节定义的有关概念,下面给出构造多变量检验的一般步骤:

(1) 计算条件属性集 C 相对于决策属性集 D 的核,即 $CORE_D(C)$.

若 $CORE_D(C)=\varnothing$,则转(2);

否则,不妨设 $CORE_D(C)=\{a_1,a_2,\ldots,a_k\}$,转(3).

(2) 用ID3的方法选择一个最佳属性,作为该结点的检验.

(3) 令 $P=a_1\wedge a_2\wedge\ldots\wedge a_k$,计算 P 相对于 D 的泛化 $GEN_D(P)$,将它作为该结点的检验.

3 与其它决策树方法的比较

3.1 与单变量决策树的比较

本节将利用文献[1]中的数据作为信息系统,如表1所示.通过这个信息系统,我们对本文提出的多变量决策树方法和著名的单变量决策树方法(ID3)进行了比较.

表1 一个信息系统

U	Condition attributes(C)				Decision attribute(D)
	Outlook(a_1)	Temperature(a_2)	Humidity(a_3)	Windy(a_4)	Class
1	sunny	hot	high	false	N
2	sunny	hot	high	true	N
3	overcast	hot	high	false	P
4	rain	mild	high	false	P
5	rain	cool	normal	false	P
6	rain	cool	normal	true	N
7	overcast	cool	normal	true	P
8	sunny	mild	high	false	N
9	sunny	cool	normal	false	P
10	rain	mild	normal	false	P
11	sunny	mild	normal	true	P
12	overcast	mild	high	true	P
13	overcast	hot	normal	false	P
14	rain	mild	high	true	N

文献[1]把属性的熵增量作为选择检验的准则,构造出了如图1的单变量决策树.该决策树可对所有的训练事例正确地分类.该树的复杂性(树中所有结点的个数)为8.

下面我们利用第2节给出的算法来构造多变量决策树.首先,在给定的信息系统中,计算条件属性集 C 相对于决策属性集 D 的核.通过简单的计算可知

$U/IND(C)=\{\{1\},\{2\},\{3\},\{4\},\{5\},\{6\},\{7\},\{8\},\{9\},\{10\},\{11\},\{12\},\{13\},\{14\}\}$

$U/IND(D)=\{\{1,2,6,8,14\},\{3,4,5,7,9,10,11,12,13\}\}$

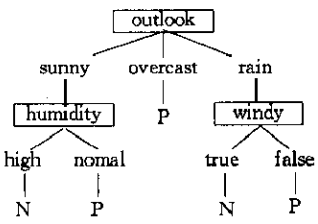

图1 单变量决策树

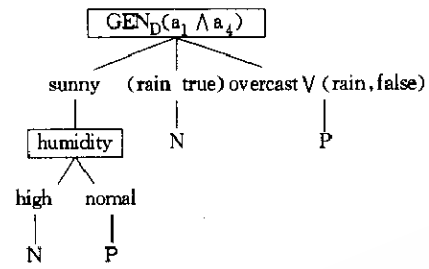

图2 多变量决策树

由公式(1),有 $POS_{IND(C)}(D)=\{1,2,3,4,5,6,7,8,9,10,11,12,13,14\}=U$

考察 $a_i(i=1,2,3,4)$ 在 C 中相对于 D 来说是否必要. 为此,从 C 中去掉 a_1,得到

$$POS_{IND(C-\{a_1\})}(D)=\{5,9,10,11,13\} \neq POS_{IND(C)}(D)$$

由公式(2)可知,a_1 在 C 中是 D-必要的.

同理可知,a_4 也是 D-必要的;而 a_2 和 a_3 是 D-不必要的. 因此,我们得到 $CORE_D(C)=\{a_1,a_4\}$.

把重要属性选出之后,接下来要利用选择的属性构造多变量检验. 首先,令 $P=a_1 \wedge a_4$,则有

$$U/P=\{\{1,8,9\},\{2,11\},\{3,13\},\{4,5,10\},\{6,14\},\{7,12\}\}$$

然后,由公式(3)和(4),可以算出 P 相对于 D 的泛化在 U 上导出的划分为

$$\{\{1,2,8,9,11\},\{3,4,5,7,10,12,13\},\{6,14\}\}$$

由于划分与属性之间存在着一一对应关系,所以,这一划分唯一地确定了 U 上的一个新属性,即我们构造的多变量检验 $GEN_D(P)$.

本文给出的算法将把 $GEN_D(P)$ 作为决策树的根,然后根据属性的值,把信息系统中的对象分成不同的子集. 对每一个子集将以类似的方式导出一棵树. 事实上,图 2 给出了用粗糙集方法从给定信息系统中产生的多变量决策树.

该多变量决策树也可以对所有的训练事例正确地分类. 但是,树的大小要比单变量决策树少 2. 我们知道,随着训练集的增大,从中导出的单变量决策树会迅速增大. 这就使人们对决策的理解变得更加困难. 同时,也会出现引言中所说的重复子树与过拟合问题. 通过一个简单的例子说明,本文提出的基于粗糙集的多变量决策树方法可以降低树的复杂性. 即使在最坏的情况下(每个结点上的核都空),也能导出与 ID3 相同的决策树. 由于这种情况出现的可能性很小,所以,一般来说本方法得到的决策树比 ID3 的决策树简单. 同时,由于使用了属性的区分能力作为划分度量准则,从而避免了在决策树的一条路径上多次检验某一属性的问题.

3.2 与多变量决策树的比较

Brodley C E[5]等人采用初始属性的线性组合来构造多变量决策树. 在属性的选择上,他们主要使用了 2 种策略:序贯后向删除(Sequential Backward Elimination)和序贯前向选择(Sequential Forward Selection). SBE 方法是以所有 n 个属性的线性组合开始,然后逐一删除对划分没有贡献的属性. SFS 方法是以零个属性开始,序贯地增加对划分最有用的属性. 在线性组合系数的确定上,使用了递归最小二乘方法(Recursive Least Squares)和 CART 方法. 其度量准则分别为均方误差最小和划分的不纯度(impurity)最小.

Rivest[8]研究了用属性的合取来表示 Boolean 函数的问题. 在其算法中,必须事先规定

合取项的最大值. 如果该值选择不当,则算法将找不到解. 在属性的选择上,使用的是完全随机的方法. 所以,其效率是极低的.

Pagallo G[4]在文献[8]的基础上,提出了一种带有启发式的多变量构造方法,即FRING方法. 该方法对决策树中深度至少为2的每一个正叶结点定义一个新的检验. 这一检验是由从树根到该叶结点路径上的最后2个属性或其否定的合取构成的. 通过迭代过程,该方法能构造出更长、更复杂的检验. 正如Pagallo所指出:"该方法通过迭代过程形成了有意义的多变量检验,进而解决了重复子树问题. 遗憾的是,该算法的分析是非常困难的".

本文提出的基于粗糙集的多变量决策树构造方法能够表达更多的概念类. 而文献[5]的RLS方法和文献[4]的FRING方法只适用于2个决策类的情况. CART方法可用于多个决策类的情况.

树中每个结点上多变量检验的大小(即它所包含的属性个数)完全是由该结点的训练集所决定. 而不象文献[8]事先人为规定一个最大值,或文献[4,5]通过迭代停止准则加以限制. 所以,它更具有客观性.

文献[4]使用熵作为划分度量准则,它受着样本大小的影响. 我们知道,在决策树构造过程中,每个分枝的样本数越来越小. 这样得到的熵就不会准确. 本文提出的方法不受样本大小的影响.

相对核的概念具有很好的解释性. 核中的属性对于制定分类决策来说是至关重要的、必需的. 它为本方法的合理性提供了依据.

4 结论及进一步研究的方向

本文提出了一种基于粗糙集的构造多变量决策树的新方法. 我们利用粗糙集理论中的条件属性集相对于决策属性集的核,解决了多变量检验中属性的选择问题. 对于多变量检验的构造,我们定义了一个等价关系相对于另一个等价关系的泛化. 利用这一概念,使得多变量检验并不是被选属性的简单合取,而是由它导出的一个新属性. 由于使用了属性的不可区分性作为划分度量准则,这一方法避免了在决策树的一条路径上多次检验某一属性的问题.

通过一个简单例子,对本文提出的多变量决策树和单变量决策树(ID3)方法进行了比较. 结果说明2种决策树都能对本例中的训练事例正确分类,但是,前者比后者更简单. 与其它几种多变量决策树方法做了初步的对比分析. 有关深入的理论分析是我们下一步研究的问题.

参考文献

1 Quinlan J R. Induction of decision trees. Machine Learning, 1986, **1**: 81～106.

2 Michalski R S, Larson J B. Selection of the most representative training examples and incremental generation of VL1 hypotheses. Rept. No. 78-867, Urbana-Champaign: Department of Computer Science, University of Illinois, 1978.

3 Cestnik B, Kononenko I, Bratko I. ASSISTANT 86: a knowledge elicitation tool for sophisticated users. In: Proceedings of EWSL-87, Bled, Yugoslavia, 1987. 31～45.

4 Pagallo G, Haussler D. Boolean feature discovery in empirical learning. Machine Learning, 1990, **5**: 71～99.

5 Brodley C E, Utgoff P E. Multivariate decision trees. Machine Learning, 1995, **19**: 45～77.

6 Pawlak Z. Rough sets: theoretical aspects of reasoning about data. Netherlands: Kluwer Academic Publishers, 1991.

7 Buntine W, Niblett T. A further comparison of splitting rules for decision-tree induction. Machine Learning, 1992, **8**: 75～85.

8 Rivest R. Learning decision lists. Machine Learning, 1987, **2**: 229～246.

ROUGH SETS BASED APPROACH FOR MULTIVARIATE DECISION TREE CONSTRUCTION

MIAO Duoqian　WANG Jue

(*Articial Intelligence Laboratory　Institute of Automation　The Chinese Academy of Sciences　Beijing*　100080)

Abstract　In this paper, the core of condition attributes with respect to decision attributes in rough sets theory is used for selection of attributes in multivariate tests. A new concept of generalization of one equivalence relation with respect to another one is introduced and used for construction of multivariate tests. The comparison between multivariate decision tree and univariate decision tree is done through an example. The results show that the former is more simple than the latter. The basic comparison among several multivariate decision trees is fulfilled.

Key words　Rough sets, univariate decision tree, multivariate decision tree, inductive learning, relative core of attributes.

Class number　TP18