# 基于不完备信息系统的决策树生成算法

关晓蕾 钱宇华
(山西大学计算机与信息技术学院 计算智能与中文信息处理省部共建重点实验室 太原 030006)

**摘 要** 决策树是一种有效地进行实例分类的数据挖掘方法。在处理不完备信息系统中的缺省值数据时,现有决策树算法大多使用猜测技术。在不改变缺失值的情况下,利用极大相容块的概念定义了不完备决策表中条件属性对决策属性的决策支持度,将其作为属性选择的启发式信息。同时,提出了一种不完备信息系统中的决策树生成算法IDTBDS,该算法不仅可以快速得到规则集,而且具有较高的准确率。

**关键词** 决策树,不完备信息系统,决策支持度
**中图法分类号** TP181 **文献标识码** A

## Algorithm for Generating Decision Tree Based on Incomplete Information Systems

GUAN Xiao-qiang QIAN Yu-hua
(Key Laboratory of Ministry of Education for Computation Intelligence and Chinese Information Processing, School of Computer and Information Technology, Shanxi University, Taiyuan 030006, China)

**Abstract** Decision trees are a kind of effective data mining methods to case classification. During processing objects with missing values in the incomplete information systems, the guessing technologies are often used in most of the existing decision tree algorithms. In this paper, we defined a condition attribute's decision support degree with respect to the decision attribute with the concept of a maximal consistent block, which can be regarded as the heuristic information. Moreover, we proposed an algorithm for generating a decision tree from an incomplete information system, which called IDTBDS. Note that the proposed algorithm not only fast extract the rule sets, and but also these rules possess more classification accuracy.

**Keywords** Decision tree, Incomplete information systems, Decision support degree

## 1 引言

决策树学习是以实例为基础的归纳学习算法,它着眼于从一组无次序、无规则的事例中推理出决策树表示形式的分类规则,在机器学习、数据挖掘、智能控制等人工智能领域有着相当重要的理论意义与实用价值[1]。

不确定环境下的决策研究一直是一个热点,决策树技术就是其中一个重要的方法。对不完备信息处理的方法有很多,其中最简单的就是去除带有丢失数据的例子,或者用最常出现的数据值代替丢失的数据。Quinlan J. R. 等人建议,基于其他已知属性的值和分类信息来预测一个数据的丢失值。2000年,Eloudi Z. 等人提出了一种基于置信度函数的决策树[2],此算法利用置信度函数原理来代表分类问题中的参数不确定性,在不确定环境下决策树的构造和分类均得到了比较好的效果。2002年,于海跃等人在粗糙集理论的知识表达系统中引入空值,提出了知识表达系统中空值的分类处理,并给出了分类处理的算法[3]。这些方法都是对缺值属性使用了猜测技术,但是猜测的数据并不能保证是正确的数据。

本文利用极大相容块的概念定义了条件属性对正确决策的支持程度,提出了一种基于决策支持度的决策树生成算法,即在不改变缺失值的情况下,对不完备信息系统可以快速得到其规则集,同时对未知数据进行分类时可以得到较高的准确率。

## 2 不完备信息系统

设 $S=(U,A)$ 是一个信息系统,其中 $U$ 是对象的非空有限集合,$A$ 是属性的非空有限集合。

在一些情况下,对一个对象而言,一些属性值可能是缺损的。为了表示这种情况,通常将一个区分值(即空值)安排给这些属性。如果对于至少一个属性 $a\in A$,$V_a$ 包含空值,则称 $S$ 是一个不完备信息系统(incomplete information system),否则它是完备的。这表明完备信息系统是不完备信息系统的一种特殊情形。进一步,我们将用 $*$ 表示空值。

**定义 1**[4] 不完备信息系统 $S=(U,A)$ 中,定义任意 $P\subseteq A$ 上的二元关系 $SIM(P)$ 为

$SIM(P)=\{(u,v)\in U\times U \mid \forall a\in P, f(u,a)=f(v,a)$ 或 $f(u,a)=*$ 或 $f(v,a)=*\}$

在二元关系 $SIM(P)$ 下,$\forall x\in U$,令 $S_P(u)=\{v\in U \mid (u,$

到稿日期:2011-02-27 返修日期:2011-06-25 本文受国家自然科学基金(60903110),山西省青年科技基金(2009021017-1)资助。
关晓蕾(1979-),女,讲师,主要研究方向为粗糙集理论、数据挖掘,E-mail:gxq0079@sxu.edu.cn;钱宇华(1976-),男,博士生,讲师,主要研究方向为粗糙集理论、粒度计算。

$v)\in SIM(P)\}$，$S_P(u)$是与 $u$ 可能不可区分的对象的最大集合(相对 $P$ 而言)，称为 $u$ 的相容类(也称 $u$ 的相容块)。容易得到二元关系 $SIM(P)$是满足自反性和对称性的相容关系。

**定义 2**[5] 设 $S=(U,A)$是不完备信息系统，$P\subseteq A$ 是一个属性子集，$X\subseteq U$ 是一个对象子集，称 $X$ 关于 $P$ 是相容的，如果对任意的 $x,y\in X$ 有$(x,y)\in SIM(P)$。如果不存在一个对象子集 $Y\subseteq U$ 使得 $X\subset Y$ 且 $Y$ 关于 $P$ 是相容的，则称 $X$ 为一个极大相容子集或极大相容块。

以 $C(P)$来记由属性 $P\subseteq A$ 所决定的所有极大相容块构成的集合。

一个决策表($DT$)是一个信息系统 $S=(U,C\cup\{d\})$，这里 $d\notin C$ 且 $*\notin V_d$ 是一个区别于 $C$ 中的属性的另一个被称之为决策的属性。相对应地，$C$ 中的属性被称之为条件属性。如果一个 $DT$ 是一个完备的信息系统，则称其为一个完备的决策表，否则称其为一个不完备决策表。

## 3 基于不完备信息系统的决策树生成算法

### 3.1 属性选择原理

对于不完备的信息系统进行分类，现有的决策树算法大多对缺失值都使用了猜测的技术，即对缺失的值进行补值再建立分类模型。但是，预测的值却不一定是正确的。因此在不改变缺失值的情况下，研究决策树的生成算法。由于采用相容类去计算，时间复杂度较高，因此我们利用极大相容块的技术定义了不完备决策表中条件属性对决策属性的决策支持度，利用决策支持度去选择测试属性(这里每次只求一个属性的极大相容块，其时间复杂度和划分的时间复杂度相同)。

**定义 3**[6] 令 $K=(U,R)$是一个近似空间，$R$ 是 $U$ 上的一个划分，$R$ 的组合熵定义为

$$CE(R)=\sum_{i=1}^{m}\frac{|R_i|}{|U|}\frac{C_{|U|}^2-C_{|R_i|}^2}{C_{|U|}^2}=\sum_{i=1}^{m}\frac{|R_i|}{|U|}\left(1-\frac{C_{|R_i|}^2}{C_{|U|}^2}\right)\qquad(1)$$

式中，$C_{|U|}^2=\frac{|U|\times(|U|-1)}{2}$，$\frac{|R_i|}{|U|}$表示等价类 $R_i$ 在论域 $U$ 上的概率；$\frac{C_{|U|}^2-C_{|R_i|}^2}{C_{|U|}^2}$表示论域上互相可以区分的元素的对数在论域 $U$ 上总的元素对数中所占的比率。

**定义 4**[7] 令 $S=(U,C\cup D)$是一个决策系统，$C\cap D=\varnothing$，$C$ 称为条件属性集，$D$ 称为决策属性集，$R\subseteq C$，$U/R=\{R_1,R_2,\cdots,R_m\}$，$U/D=\{D_1,D_2,\cdots,D_n\}$。定义条件属性子集 $R$ 对决策属性集 $D$ 的决策支持度为

$$S(R,D)=1-\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{n}\frac{|R_i\cap D_j|\times|R_i-D_j|}{C_{|U|}^2-\sum_{l=1}^{n}C_{|D_l|}^2}\qquad(2)$$

式中，$C_{|U|}^2-\sum_{l=1}^{n}C_{|D_l|}^2$表示论域 $U$ 上需要区分的元素对数，$\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{n}|R_i\cap D_j|\times|R_i-D_j|$表示相对于 $D$ 利用条件属性 $R$ 不能区分的元素对数。

**定理 1**[7] 令 $S=(U,C\cup D)$是一个决策表，$C\cap D=\varnothing$，$C$ 称为条件属性集，$D$ 称为决策属性集，$R\subseteq C$，$U/R=\{R_1,R_2,\cdots,R_m\}$，$U/D=\{D_1,D_2,\cdots,D_n\}$，则 $R\preceq D$ 当且仅当 $S(R,D)=1$。

**定义 5** 令 $S=(U,C\cup D)$是一个不完备决策表，$C\cap D=\varnothing$，$C$ 称为条件属性集，$D$ 称为决策属性集，$R\subseteq C$，$C(R)=\{R_1,R_2,\cdots,R_m\}$，$U/D=\{D_1,D_2,\cdots,D_n\}$，令 $|C(R)|=\sum_{i=1}^{m}|R_i|$ 定义条件属性子集 $R$ 对决策属性集 $D$ 的决策支持度为

$$S(R,D)=1-\sum_{i=1}^{m}\sum_{j=1}^{n}\frac{|R_i\cap D_j|\times|R_i-D_j|}{|C(R)|\times(|U|-1)-\sum_{l=1}^{n}(|D_l|\times(|D_l|-1))}\qquad(3)$$

式中，$|C(R)|\times(|U|-1)-\sum_{l=1}^{n}(|D_l|\times(|D_l|-1))$表示由属性 $R$ 得到的覆盖和 $U$ 中的元素需要区分的元素对数，$\sum_{i=1}^{m}\sum_{j=1}^{n}|R_i\cap D_j|\times|R_i-D_j|$表示相对于 $D$ 利用条件属性 $R$ 不能区分的元素对数。

决策支持度 $S(R,D)$表示属性 $R$ 对划分 $U/D$ 的支持程度。$S(R,D)$的值越大，$C(R)$越接近 $U/D$，表明子集 $R$ 对分类的贡献越大，那么选择属性集 $R$ 进行分类的确定性就越大。

决策支持度 $S(R,D)$具有以下性质。

**性质 1** $0\leqslant S(R,D)\leqslant 1$。

**性质 2** 当 $C(R)=U/D$ 时，$S(R,D)=1$。

**性质 3** $C(R)\subseteq U/D$ 时，$S(R,D)=1$。

证明：当 $C(R)=U/D$ 或 $C(R)\subseteq U/D$ 时，由属性 $R$ 得到的 $C(R)$是 $U$ 上的一个划分，即 $C(R)=U/R$，$|C(R)|=|U|$，则 $S(R,D)=1-\sum_{i=1}^{m}\sum_{j=1}^{n}\frac{|R_i\cap D_j|\times|R_i-D_j|}{|U|\times(|U|-1)}$，根据定理 1，结论成立。

**性质 4** 当 $C(R)=\delta$，$U/D=\omega$ 时，$S(R,D)=0$，其中 $\omega$ 是对论域最细的划分。

**定理 2** 设 $S=(U,C\cup D)$是一个完备决策表，$C\cap D=\varnothing$，$C$ 称为条件属性集，$D$ 称为决策属性集，$R\subseteq C$，$C(R)=\{R_1,R_2,\cdots,R_m\}$，$U/D=\{D_1,D_2,\cdots,D_n\}$，则条件属性子集 $R$ 对决策属性集 $D$ 的决策支持度退化为

$$S(R,D)=1-\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{n}\frac{|R_i\cap D_j|\times|R_i-D_j|}{C_{|U|}^2-\sum_{l=1}^{n}C_{|D_l|}^2}$$

证明：因为 $S=(U,C\cup D)$是一个完备决策表，所以有

$$C(R)=U/R=\{R_1,R_2,\cdots,R_m\},\ |C(R)|=\sum_{i=1}^{m}|R_i|=|U|$$

因此

$$\begin{aligned}S(R,D)&=1-\sum_{i=1}^{m}\sum_{j=1}^{n}\frac{|R_i\cap D_j|\times|R_i-D_j|}{|C(R)|\times(|U|-1)-\sum_{l=1}^{n}(|D_l|\times(|D_l|-1))}\\&=1-\sum_{i=1}^{m}\sum_{j=1}^{n}\frac{|R_i\cap D_j|\times|R_i-D_j|}{|U|\times(|U|-1)-\sum_{l=1}^{n}(|D_l|\times(|D_l|-1))}\\&=1-\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{n}\frac{|R_i\cap D_j|\times|R_i-D_j|}{C_{|U|}^2-\sum_{l=1}^{n}C_{|D_l|}^2}\end{aligned}$$

证毕。

### 3.2 算法描述

**算法** $IDTBDS(T,T\_attributelist)$

输入：决策表 $S=(U,C\cup D)$；

输出：一棵决策树。

说明：$T$ 代表当前样本集，$T\_attributelist$ 表示当前的候选属性集，$|T_attributelist|$表示候选属性集 $T\_attributelist$ 中的属性个数。

算法步骤：

(1) 创建根结点 $N$，包含的样本集为 $T$，$T\_attributelist=C$；

(2) 计算决策属性 $D$ 对样本集 $T$ 的划分；

(3) IF $T$ 都属于同一类 $C$，则返回 $N$ 为叶结点，标记为类 $C$；

(4) IF $T\_attributelist$ 为空，则返回 $N$ 为叶结点，将 $N$ 中所有的类别标记为该叶子的最终类别，并计算每种类别出现的概率；

(5) 计算每个条件属性对样本集的极大相容块集；

(6) FOR EACH 对 $T\_attributelist$ 中的条件属性 $R_i$ 计算其对决策属性 $D$ 的决策支持度 $S(R_i,D)$，其中 $1\leqslant i\leqslant|T\_attributelist|$；

(7) IF $T\_attributelist$ 中的所有属性的决策支持度都为 0，则返回 $N$ 为叶子结点，将 $N$ 中所有的类别标记为该叶子的最终类别，并计算每种类别出现的概率；结束；

(8)否则，$N$ 的测试属性 $Test\_attributelist=T\_attributelist$ 中具有最高决策支持度的属性 $R$；

(9) 将属性 $R$ 从 $T\_attributelist$ 中去掉，形成新的测试属性列表 $T'\_attributelist$；

(10) FOR EACH $R$ 的每一个极大相容块

{ 由 $N$ 结点长出一个新子结点，

IF 新叶结点对应的样本子集 $T'$ 为空，则删除此结点；

ELSE 在该结点上执行 $DTBDS(T',T'\_attributelist)$；

}

## 4 实验仿真与分析

考虑表 1 中几个小汽车的描述。

表 1 一个不完备的决策系统

| Car | Price | Mileage | Size | Max-Speed | d |
|---|---|---|---|---|---|
| 1 | high | low | full | low | good |
| 2 | low | * | full | low | good |
| 3 | * | * | compact | low | poor |
| 4 | high | * | full | high | good |
| 5 | * | * | full | high | excellent |
| 6 | low | high | full | * | good |

在本例中，分别用 $P,M,S,X$ 来表示条件属性 Price，Mileage，Size 和 Max-Speed。

利用定义 5 中公式分别计算每个条件属性对决策属性的决策支持度。

$$S(P,D)=1-\frac{2\times2+1\times3+1\times3+2\times2+1\times3+1\times3}{(6-1)\times8-12}$$

$$=0.2875$$

$$S(M,D)=1-\frac{3\times2+1\times4+1\times4+3\times2+1\times4+1\times4}{(6-1)\times10-12}$$

$$=0.2632$$

$$S(S,D)=1-\frac{4\times1+0+1\times4+0+0+0}{(6-1)\times6-12}=0.5556$$

$$S(X,D)=1-\frac{3\times1+1\times3+0+2\times1+0+1\times2}{(6-1)\times7-12}$$

$$=0.5652$$

显然，条件属性 $X$ 的决策支持度最大，所以 $X$ 将作为根；并将 $X$ 从属性列表中去除，形成新的属性列表 $T'\_attributelist=\{P,M,S\}$。根据 $X$ 的极大相容块把 $T$ 中的对象分成两个子集，构造出两个分支；把每个子集中的对象作为新的论域，依次对每个分支中的对象按算法重复上述操作，得到图 1 所示的决策树。

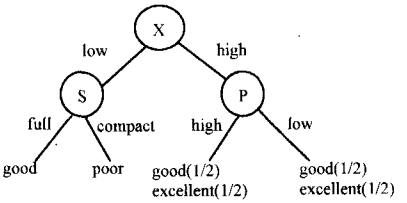


图 1 基于决策支持度的决策树

由图 1，可以得到 4 条规则。

$r_1$：$(X=\text{low})\wedge(S=\text{compact})\rightarrow d=\text{poor}$；

$r_2$：$(X=\text{low})\wedge(S=\text{full})\rightarrow d=\text{good}$；

$r_3$：$(X=\text{high})\wedge(P=\text{high})\rightarrow(d=\text{good})\vee(d=\text{excellent})$；

$r_4$：$(X=\text{high})\wedge(P=\text{low})\rightarrow(d=\text{good})\vee(d=\text{excellent})$。

为了验证 IDTBDS 算法的有效性，我们进行了实验。采用 Microsoft SQL Server 2000 中的 T-SQL 语言编写存储过程实现算法。使用 UCI 中的部分数据集作为实验数据，对 IDTBDS 算法和 C4.5 算法做进一步的性能测试比较。实验结果如表 2 所列。

表 2 算法性能比较

| | 规则数 | | 分类精度 | |
|---|---|---|---|---|
| | C4.5 | IDTBDS | C4.5 | IDTBDS |
| Lung-cancer | 29 | 29 | 90.61% | 90.90% |
| Soybean-large | 166 | 163 | 94.35% | 95.43% |
| Breast cancer | 102 | 98 | 92.36% | 96.12% |
| Balance scale | 313 | 313 | 26.86% | 36.15% |

由于决策支持度可以更好地刻画条件属性对数据进行正确区分的能力，因此根据决策支持度选择的属性对分类的贡献更大。实验证明，本文提出的算法得到的决策树的规模更小，分类精度更高。

**结束语** 本文利用极大相容块技术定义了不完备决策表中条件属性对决策属性的决策支持度，并提出一种在不完备信息系统中生成决策树的算法。理论分析和实验证明，本文算法在不完备信息系统中生成决策树，获取规则是可行的，并且算法简单，得到的决策树规模较小，精度较高。本文结果为信息系统中的不确定性度量提供了一种新的工具和方法。

## 参 考 文 献


[1] 史忠植. 知识发现[M]. 北京：清华大学出版社，2002

[2] Elouedi Z, Mellouli K, Smets P. Decision trees using the belief function theory[C]// Proceedings of the Eighth International Conference IPMU. 2000

[3] 于跃海，何建敏，邱海波，等. 空值环境下基于粗集理论的知识表达研究[J]. 系统工程学报，2002，17(1)：62-66，81

[4] 梁吉业，李德玉. 信息系统中的不确定性与知识获取[M]. 北京：科学出版社，2005

[5] Leung Y, Li D Y. Maximal consistent block technique for rule acquisition in incomplete information systems[J]. Information Science, 2003, 153: 85-106

[6] Liang J Y, Qian Y H. Combination entropy and combination granulation in rough set theory[J]. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 2008, 16(2): 179-193

[7] 关晓蕾，梁吉业，钱宇华，等. 基于决策支持度的决策树生成算法[J]. 计算机工程与应用，2008，44(27)：148-150