决策树 ID3 算法研究及其优化

武献宇[1],王建芬[2],谢金龙[1]
(1.湖南现代物流职业技术学院,湖南 长沙 410131;
2.长沙医学院,湖南 长沙 410131)

摘 要:重点研究了经典的、具有较大影响力的决策树分类算法——ID3 算法,并对其性能优劣作了比较分析。就 ID3 算法两个较为明显的缺陷进行了探讨,提出了优化算法。

关键词:数据挖掘;分类;决策树;信息增益

中图分类号:TP31 **文献标识码**:A **文章编号**:1674-7720(2010)21-0007-03

The research of ID3 decision tree algorithm and its optimization

WU Xian Yu[1], WANG Jian Fen[2], XIE Jin Long[1]
(1.Hunan Vocational College of Modern Logistics, Changsha 410131, China;
2.Changsha Medical University, Changsha 410131, China)

Abstract: The paper focus on the classic and affected decision tree classification algorithm-ID3 algorithm, and evaluates the pros and cons of ID3 decision tree algorithm. Then, it discusses the improvement of the two obvious deficiencies of ID3 algorithm, and puts forward an improved ID3 algorithm.

Key words: data mining; classification; decision tree; information gain

分类是一种非常重要的数据挖掘方法,也是数据挖掘领域中被广泛研究的课题。决策树分类方法是一种重要的分类方法,它是以信息论为基础对数据进行分类的一种数据挖掘方法。决策树生成后成为一个类似流程图的树形结构,其中树的每个内部结点代表一个属性的测试,分枝代表测试结果,叶结点则代表一个类或类的分布,最终可生成一组规则。相对其他数据挖掘方法而言,决策树分类方法因简单、直观、准确率高且应用价值高等优点在数据挖掘及数据分析中得到了广泛应用。

1 决策树分类过程

决策树的分类过程也就是决策树分类模型(简称决策树)的生成过程,如图 1 所示。从图中可知决策树分类的建立过程与用决策树分类模型进行预测的过程实际上是一种归纳-演绎过程。其中,由已分类数据得到决策树分类模型的过程称归纳过程,用决策树分类模型对未分类数据进行分类的过程称为演绎过程。需要强调的是:由训练集得到分类模型必须经过测试集测试达到一定要求才能用于预测。

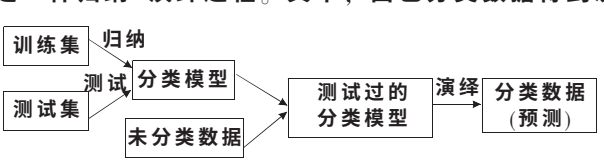

图 1 决策树分类过程图

2 ID3 算法

2.1 ID3 算法的理论基础

ID3 算法的理论依据为:

设 $E=F_1\times F_2\times\cdots\times F_n$ 是 n 维有穷向量空间,F_j 是有穷离散符号集,E 中的元素 $e=<V_1,V_2,\cdots,V_n>$ 称为例子,其中 $V_j\in F_j, j=1,2,\cdots,n$。设 PE 和 NE 是 E 的两个例子集,分别叫正例集和反例集。

假设向量空间 E 中的正例集 PE 和反例集 NE 的大小分别为 p 和 n。ID3 算法是基于如下两种假设:

(1)向量空间 E 上的一棵正确决策树对任意例子的分类概率同 E 中的正反例的概率一致。

(2)一棵决策树对一例子做出正确类别判断所需的信息量为:

$$E(p,n)=\frac{p}{p+n}\log\frac{p+n}{p}+\frac{n}{p+n}\log\frac{p+n}{n}$$

如果以属性 A 作为决策树的根，A 具有 V 个值 $\{V1, V2, V3, \cdots, Vv\}$，它们将 E 分成 V 个子集 $\{E1, E2, \cdots, Ev\}$，假设 E_i 中含有 pi 个正例和 ni 个反例，则子集 E_i 所需的期望信息是 $E(pi, ni)$，以属性 A 为根的期望熵为：

$$E(A)=\sum_{i=1}^{v}\frac{p_i+n_i}{p+n}E(p_i,n_i)$$

其中，$E(p_i,n_i)=\frac{p_i}{p_i+n_i}\log\frac{p_i+n_i}{p_i}+\frac{n_i}{p_i+n_i}\log\frac{p_i+n_i}{n_i}$

因此，以 A 为根的信息增益是：

$$GAIN(A)=E(p,n)-E(A)$$

信息增益是不确定性的消除，也就是接收端所获得的信息量。

2.2 ID3 算法多值偏向性分析

首先，设 A 是某训练样本集的一个属性，它的取值为 $A_1, A_2, \cdots, A_n$，创建另外一个新属性 A'，它与属性 A 唯一的区别：其中一个已知值 A_n 分解为两个值 A'_n 和 A'_{n+1}，其余值和 A 中的完全一样，假设原来 n 个值已经提供足够的信息使分类正确进行，很明显，则属性 A' 相对于 A 没有任何作用。但如果按照 Qulnina 的标准，属性 A' 应当优先于属性 A 选取。

综上所知，把 ID3 算法分别作用在属性 A 和属性 A' 上，如果属性选取标准在属性 A' 上的取值恒大于在属性 A 上的取值，则说明该算法在建树过程中具有多值偏向；如果属性选取标准在属性 A' 上的取值与在属性 A 上的取值没有确定的大小关系，则说明该决策树算法在建树过程中不具有多值偏向性。

2.3 ID3 算法的缺点

(1)ID3 算法往往偏向于选择取值较多的属性，而在很多情况下取值较多的属性并不总是最重要的属性，即按照使熵值最小的原则被 ID3 算法列为应该首先判断的属性在现实情况中却并不一定非常重要。例如：在银行客户分析中，姓名属性取值多，却不能从中得到任何信息。

(2)ID3 算法不能处理具有连续值的属性，也不能处理具有缺失数据的属性。

(3)用互信息作为选择属性的标准存在一个假设，即训练子集中的正、反例的比例应与实际问题领域中正、反例的比例一致。一般情况很难保证这两者的比例一致，这样计算训练集的互信息就会存在偏差。

(4)在建造决策树时，每个结点仅含一个属性，是一种单变元的算法，致使生成的决策树结点之间的相关性不够强。虽然在一棵树上连在一起，但联系还是松散的。

(5)ID3 算法虽然理论清晰，但计算比较复杂，在学习和训练数据集的过程中机器内存占用率比较大，耗费资源。

决策树 ID3 算法是一个很有实用价值的示例学习算法，它的基础理论清晰，算法比较简单，学习能力较强，适于处理大规模的学习问题，是数据挖掘和知识发现领域中的一个很好的范例，为后来各学者提出优化算法奠定了理论基础。表 1 是一个经典的训练集。

表 1 天气数据库训练数据集

属性	天气	温度	湿度	风	是否适合
1	多云	高	大	无风	不适合
2	多云	高	大	大风	不适合
3	多云	高	大	中风	不适合
4	晴	高	大	无风	适合
5	晴	高	大	中风	适合
6	雨	适中	大	无风	不适合
7	雨	适中	大	中风	不适合
8	雨	高	正常	无风	适合
9	雨	低	正常	中风	不适合
10	雨	高	正常	大风	不适合
11	晴	低	正常	大风	适合
12	晴	低	正常	中风	适合
13	多云	适中	大	无风	不适合
14	多云	适中	大	中风	不适合
15	多云	低	正常	无风	适合
16	多云	低	正常	中风	适合
17	雨	适中	正常	无风	不适合
18	雨	适中	正常	中风	不适合
19	多云	适中	正常	中风	适合
20	多云	适中	正常	大风	适合
21	晴	适中	大	大风	适合
22	晴	适中	大	中风	适合
23	晴	高	正常	无风	适合
24	雨	适中	大	大风	不适合

由 ID3 算法递归建树得到一棵决策树如图 2 所示。

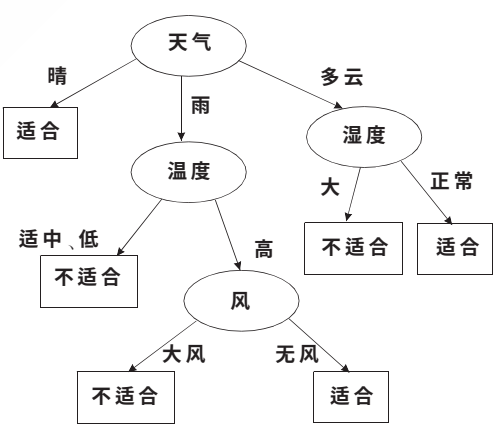

图 2 ID3 算法所生成的决策树

3 ID3 算法优化的探讨

ID3 算法在选择分裂属性时，往往偏向于选择取值较多的属性，然而在很多情况下取值较多的属性并不总是最重要的属性，这会造成生成的决策树的预测结果与实际偏离较大，针对这一弊端，本文提出以下改进思路：

(1)引入分支信息熵的概念。对于所有属性，任取属性 A，计算 A 属性的各分支子集的信息熵，在每个分支子集中找出最小信息熵，并计算其和，比较大小，选择其最小值作为待选择的最优属性。

(2)引入属性优先的概念。不同的属性对于分类或决策有着不同的重要程度，这种重要程度可在辅助知识的基础上事先加以假设，给每个属性都赋予一个权值，其大小为(0，1)中的某个值。通过属性优先法，降低非重要属性的标注，提高重要属性的标注。

4 实例分析

仍以表 1 为例，分别计算其 $H(A)$的值。在此通过反复测试，天气的属性优先权值为 0.95，风的属性优先权值为 0.35，其余两个的属性优先权值都为 0。

(1)确定根结点

选取天气属性作为测试属性，天气为多云时，信息熵为：

$$E(\text{天气}_{(\text{多云})}/\text{温度})=\frac{3}{9}\left(\frac{3}{3}\log_2\frac{3}{3}+0\right)+\frac{4}{9}\left(\frac{2}{4}\log_2\frac{4}{2}+\frac{2}{4}\log_2\frac{4}{2}\right)+\frac{2}{9}\left(\log_2\frac{2}{2}+0\right)=0.444\ 44$$

同理 $E(\text{天气}_{(\text{多云})}/\text{湿度})=0$，$E(\text{天气}_{(\text{多云})}/\text{风})=0.972\ 77$

从上面的计算可知，天气为多云时的最小信息熵为 0。

当天气为晴时，由于此时对应的类别全部为适合打高尔夫球，所以信息熵都为 0。

当天气为雨时的信息熵为：

$$E(\text{天气}_{(\text{雨})}/\text{温度})=\frac{2}{8}\left(\frac{1}{2}\log_2 2+\frac{1}{2}\log_2 2\right)+\frac{5}{8}\left(0+\frac{5}{5}\log_2\frac{5}{5}\right)+\frac{1}{8}\left(0+\frac{1}{1}\log_2 1\right)=0.25$$

同理 $E(\text{天气}_{(\text{雨})}/\text{湿度})=0.451\ 21$，$E(\text{天气}_{(\text{雨})}/\text{风})=0.344\ 36$

从上面的计算可知，天气为雨时的最小信息熵为 0.25。

$$E'(\text{天气})=\left(\frac{9}{24}+0.95\right)\times 0+\left(\frac{7}{24}+0.95\right)\times 0+\left(\frac{8}{24}+0.95\right)\times 0.25=0.320\ 83$$

$$E(U/\text{天气})=\frac{9}{24}\left(\frac{4}{9}\log_2\frac{9}{4}+\frac{5}{9}\log_2\frac{9}{5}\right)+\frac{7}{24}\left(\frac{7}{7}\log_2\frac{7}{7}+0\right)+\frac{8}{24}\left(\frac{1}{8}\log_2 8+\frac{7}{8}\log_2\frac{8}{7}\right)=0.552\ 84$$

$$H(\text{天气})=E(\text{天气})+(E(U/\text{天气})+0.95)=1.823\ 67$$

同理 $H(\text{温度})=1.083\ 83$，$H(\text{湿度})=1.068\ 7$，$H(\text{风})=2.775\ 54$

根据算法步骤(6)，选择值 $H(A)$为最小的作为候选属性，所以此时应选择湿度作为根结点。在 24 个训练集中对湿度的 2 个取值进行分枝，2 个分枝对应 2 个子集，分别为：

$F_1=\{1,2,3,4,5,6,7,13,14,21,22,24\}$(湿度为高对应)

$F_2=\{8,9,10,11,12,15,16,17,18,19,20,23\}$(湿度为正常对应)

(2)确定分支结点

计算 F_1 分支子集：

$H(\text{温度})=0.908\ 05$，$H(\text{天气})=0.95$，$H(\text{风})=1.554\ 56$

选择 $H(A)$值为最小的作为候选属性，所以此时应选择温度作为湿度为大的下一级结点。在湿度为大的 12 个训练集中对温度的 3 个取值进行分枝，3 个分枝对应 3 个子集，由于温度为低的子集不存在，所以此时也只有 2 个子集，分别为：

$F_{11}=\{1,2,3,4,5\}$(湿度为大且温度为高对应)

$F_{12}=\{6,7,13,14,21,22,24\}$(湿度为大且温度为适中对应)

同理，对于 F_2 分支子集：

$H(\text{温度})=0.863\ 71$，$H(\text{天气})=1.250\ 8$，$H(\text{风})=1.554\ 6$

选择 $H(A)$值最小的作为候选属性，所以此时应选择温度作为湿度为正常的下一级结点。在湿度为正常的 12 个训练集中对温度的 3 个取值进行分枝，3 个分枝对应 3 个子集，分别为：

$F_{21}=\{8,10,23\}$(湿度为正常且温度为高对应)

$F_{22}=\{17,18,19,20\}$(湿度为正常且温度为适中对应)

$F_{23}=\{9,11,12,15,16\}$(湿度为正常且温度为低对应)

继续对 F_{11}，F_{12}，F_{21}，F_{22}，F_{23} 等分支子集采用此优化算法递归建树，经过合并和简化后得到的决策树如图 3 所示。

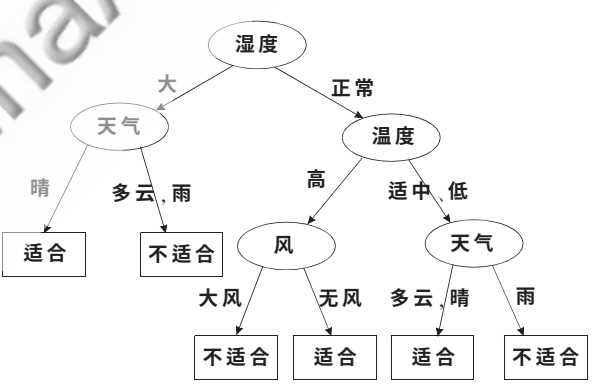

图 3 优化算法所生成的决策树

通过比较 ID3 算法和本文所提出的组合优化算法所生成的决策树可知，组合优化算法的改进为：

(1)从本实例所生成的决策树的形态来看，改进后的算法生成的是一棵二叉树，而 ID3 算法生成的是多叉树，简化了决策问题处理的复杂度。

(2)引入了分支信息熵和属性优先的概念，用条件熵、分支信息熵与属性优先三者相结合来选择分裂属性。从本实例来看，根结点选择湿度而未选择属性值最多的天气，所以本优化算法确实能克服传统 ID3 算法的多值偏向性。

参考文献
[1] 安淑芝.数据仓库与数据挖掘[M].北京:清华大学出版社,2005:104-107.
[2] 史忠植.知识发现[M].北京:清华大学出版社,2002:23-37.
[3] 徐洁磐.数据仓库与决策支持系统[M].北京:科学出版社,2005:77-86.
[4] 路红梅.基于决策树的经典算法综述[J].宿州学院学报,2007(4):91-95.
[5] 韩慧.数据挖掘中决策树算法的最新进展[J].计算机应用研究,2004(12):5-8.
[6] KANTARDZIC M. Data mining concepts, models, methods, and algorithms[M]. 北京:清华大学出版社,2003:120-136.
[7] OLARU C, WEHENKEL L. A complete fuzzy decision tree technique [J]. Fuzzy Sets and Systems, 2003,138 (2):221-254.
[8] AITKENHEAD M J. Aco-evolving decision tree classification method[J]. Expert System with Application, 2008(34):18-25.
[9] Norio Takeoka. Subjective probability over a subjective decision tree[J]. Journal of Economic Theory, 2007(136):536-571.

(收稿日期:2010-02-05)

作者简介:
武献宇,男,1974年生,讲师,硕士,主要研究方向:软件工程和数据挖掘技术。