决策树在计算机等级考试中的应用①

丁 勇, 武玉艳

(南京理工大学泰州科技学院 计算机科学与技术系, 泰州 225300)

摘要: 江苏省计算机等级考试是由江苏省教育厅推行的一项考试制度, 该考试越来越受到高校和学生们的重视. 首先基于历届学生的考试真实数据, 应用信息熵和 ID3 算法构造决策树. 然后, 提取分类规则, 并通过计算规则的准确率与覆盖率对规则进行约简, 从而挖掘出有价值的规则。最后利用该分类规则, 预测学生能否通过等级考试。通过对历史数据进行仿真实验, 表明决策树预测准确率高, 能挖掘出影响学生通过等级考试的关键因素, 对计算机等级考试课程教学有一定的指导作用.

关键词: 计算机等级考试; 决策树; 信息增益; ID3 算法

Application of the Decision Tree in the Computer Rank Examination

DING Yong, WU Yu-Yan

(Department of Computer Science and Technology, Nanjing University of Science and Technology, Taizhou 225300, China)

Abstract: Jiangsu Computer Rank Examination is a examination mechanism formed by Education Department, which gains the attention of universities and students. First, based on data collected from previous students' exeamination, this paper uses Information Gain and ID3 algorithm to construct a decision tree. Then, this paper extracts the classification rules from the decision tree, and discovers these rules of the accuracy and coverage. Finally, this paper predicts the ability of students to pass the exam by these rules. Experiments indicate that the decision tree has accurate prediction, which can dig out the key factors affecting students to pass the grade examination and plays a crucial role in the course teaching.

Key words: computer rank examination; decision tree; information gain; ID3 Algorithm

江苏省计算机等级考试是由江苏省教育厅推行的一项考试制度. 该考试的目的是加强普通高校非计算机专业的学生对计算机基础知识的理解和掌握, 考试设立多个语种和等级，考生通过相应等级的考试可获得证书. 这项考试制度实施至今，得到了全省普通高校和用人单位的广泛认可，因此也受到高校和学生们的高度重视. 但由于试题知识点多、难度大等因素，通过率普遍较低. 为了更好地指导学生顺利考过考试，挖掘考试隐含的相关信息，并进行有针对性的指导，是十分有必要的.

决策树方法产生于 20 世纪 70 年代后期，它是一种典型的分类方法，用于发现数据中蕴涵的分类规则. 该方法首先基于一组训练样本数据，通过相应的算法构造决策树，并从树中获得分类规则，然后对分类规则进行约简，利用规则对未来数据进行预测[1]. 决策树方法分类精确，预测准确率高，可应用于挖掘计算机等级考试中的隐含信息.

1 决策树算法

常见的决策树算法有 ID3 算法、C4.5 算法[2,3]. ID3 算法使用信息增益(Gain)作为属性选择的度量，C4.5 算法使用信息增益率(Gain Ratio)将信息增益规范化.

1.1 ID3 算法

ID3 算法是由 J.R.Quinlan 提出的归纳分类算法，

①收稿时间:2013-10-26;收到修改稿时间:2013-11-18

算法的基本思想是找出具有最大信息增益(Gain)的字段作为决策树的一个节点，再根据该字段的不同属性值建立树的分支，对每个分支重复建立树的下层节点和分支，直到分支的属性值属于同一类. 算法描述如下.

算法: Generate_Decision_Tree
输入: 训练样本 DataSets(D)
候选属性集 Attributes，包含分类属性
输出: 决策树
方法:
(1) 创建一个节点 N;
(2) If D 中的元组属于同一类 C then
(3) 返回 N 作为叶节点，以类 C 为标记;
(4) If attributes 为空 then
(5) 返回 N 作为叶节点，标记为普通类;
(6) 选择 attributes 中具有最高信息增益(Gain)的属性作为分裂属性 (split_attribute) ;
(7) 使用 split_attribute 标记节点 N;
(8) For split_attribute 的每个属性值 Vi
(9) 由节点 N 生长出一个条件为 split_attribute=Vi 的分枝;
(10) 设 Di 是 D 中满足 split_attribute=Vi 的数据集合;
(11) IF Di 为空 then
(12) 加一个叶节点 N，标记为普通类;
(13) Else
(14) 加一个由 Generate_Decision_Tree (Di，attributes-split_attribute)返回的节点;
(15) End for

1.2 信息增益计算

ID3 算法最关键的是对训练样本中属性的信息增益(Gain)的计算. 对 D 中元组按类标号 C 进行分类所需的期望信息由公式 1 表示:

$$\mathrm{Info(D)} = -\sum_{i=1}^{m} P_i \log_2^{pi} \qquad (1)$$

其中，P_i 是 D 中任意元组属于类 C_i 的概率，因为采用二进制编码，因此采用以 2 为底的对数函数. Info(D)又称为 D 的熵.

假设按 A 属性划分 D 中的元组，属性 A 有 v 个不同值{a1,a2,a3.....av}，用属性 A 将 D 划分为 v 个子集{D1,D2,...Dv}，其中 Dj 中的样本在属性 A 上具有相同的值 aj(j=1,2,...v). 设 Dij 是子集 Dj 中类 Ci 的样本数，由 A 划分成子集所需的期望信息由公式 2 表示:

$$Info_A(D) = \sum_{j=1}^{v} \frac{|Dj|}{|D|} \times Info(Dj) \qquad (2)$$

其中，$Info(Dj)$ 是属性 A 划分子集 Dj 所需的期望信息. 信息增益为基于类标号划分 D 的期望信息与基于 A 属性划分 D 的期望信息之间的差，由公式 3 表示:

$$Gain(A) = Info(D) - Info_A(D) \qquad (3)$$

C4.5 算法在 ID3 算法的基础上进行了改进，克服了信息分布可能存在的“偏倚”现象. 基于 A 属性划分 D 的期望信息增益率，由公式 4 表示:

$$GainRatio(A) = \frac{Gain(A)}{Info(A)} \qquad (4)$$

1.3 提取规则

规则用于表示数据集中属性之间的关系，可以用 IF-THEN 的形式来表示. IF 部分称作规则的前件，Then 部分称作规则的后件. 基于决策树提取规则的方法是沿着树中由根节点到每个树叶节点的路径，每个叶子节点都创建一条规则，每个分割都成为规则中的一个条件，叶子节点中的类别就是 Then 的内容，算法如下.

算法: Generate_Rules(Node)
输入: 决策树(Decision_Tree)
输出: 规则(Rules)
方法:
(1)Node=Root //根节点
(2)If Node 不为空 then
(3) for all Child in Node //每个子节点
(4) if Child=叶子节点 then
(5) Ruls.left=Child.Name;
(6) Rules.right=Child.Type;
(7) Genarate(Child) //递归调用
(8) Else
(9) Return //返回上一层节点
(10) End for
(11)End If
(12)Return Rules //返回规则

2 计算机等级考试成绩预测

2.1 数据预处理

学生信息包括学号、姓名、年龄、性别、年级、学院、专业、班级、所选课程、平时成绩、理论成绩、上机成绩等，选择具有代表性的特征信息，如语言基础、是否按时完成作业、理论成绩、上机成绩、是否通过等级考试等.

对学生成绩进行离散化处理，将理论成绩、上机成绩的数值泛化为离散的区间，成绩在 80-100 之间为 High(H)，60-79 之间为 Middle(M),<60 为 Low(L)，如表 1 所示.

表 1　学生计算机等级考试信息

语言基础 foundation	完成作业 exercises	理论成绩 theory	上机成绩 machine	通过 pass
yes	yes	H	M	yes
no	yes	M	H	yes
no	no	H	M	yes
yes	no	L	L	no
no	yes	M	M	yes
yes	no	L	M	no
no	no	L	L	no
no	no	M	L	no
yes	yes	L	M	yes
no	no	L	L	no
yes	yes	L	L	no
no	no	L	L	no
no	no	M	M	no
no	yes	L	L	no
yes	no	L	M	no

2.2 构造决策树

下面基于训练集(表 1)中的数据，给出构造决策树的步骤.

Step1 计算分类属性(pass)的期望信息. 属性 pass 有两个不同值(yes，no)，值为 yes 的有 5 个元组，值为 no 的有 10 个元组，因此对 D 中元组划分的期望信息为:

$$Info(D) = -\frac{5}{15}\log_2(\frac{5}{15}) - \frac{10}{15}\log_2(\frac{10}{15}) = 0.918296(位)$$

Step2 计算按每个属性对 D 中元组分类的期望信息. 属性 foundation 有两个不同值(yes，no)，值为 yes 的有 6 个元组(将 pass 划分为 2 个 yes，4 个 no)，值为 no 的有 9 个元组(将 pass 划分为 3 个 yes，6 个 no).

因此，基于属性 foundation 对 D 中元组划分的期望信息为:

$$Info_{foundation}(D) = \frac{6}{15}\times(-\frac{2}{6}\log_2(\frac{2}{6}) - \frac{4}{6}\log_2(\frac{4}{6})) + \frac{9}{15}\times(-\frac{3}{9}\log_2(\frac{3}{9}) - \frac{6}{9}\log_2(\frac{6}{9})) = 0.918296(位)$$

同理:

$$Info_{exercises}(D) = \frac{6}{15}\times(-\frac{4}{6}\log_2(\frac{4}{6}) - \frac{2}{6}\log_2(\frac{2}{6})) + \frac{9}{15}\times(-\frac{1}{9}\log_2(\frac{1}{9}) - \frac{8}{9}\log_2(\frac{8}{9})) = 0.669273(位)$$

$$Info_{theroy}(D) = \frac{2}{15}\times(-\frac{2}{2}\log_2(\frac{2}{2})) + \frac{4}{15}\times(-\frac{2}{4}\log_2(\frac{2}{4}) - \frac{2}{4}\log_2(\frac{2}{4})) + \frac{9}{15}\times(-\frac{1}{9}\log_2(\frac{1}{9}) - \frac{8}{9}\log_2(\frac{8}{9})) = 0.568622(位)$$

$$Info_{machine}(D) = \frac{1}{15}\times(-\frac{1}{1}\log_2(\frac{1}{1})) + \frac{7}{15}\times(-\frac{3}{7}\log_2(\frac{3}{7}) - \frac{4}{7}\log_2(\frac{4}{7})) + \frac{7}{15}\times(-\frac{7}{7}\log_2(\frac{7}{7})) = 0.459773(位)$$

step3 计算每个属性的信息增益.

$$Gain(foundation) = Info(D) - Info_{foundation}(D) = 0(位)$$

$$Gain(exercises) = Info(D) - Info_{exercises}(D) = 0.918296 - 0.669273 = 0.249023(位)$$

$$Gain(theory) = Info(D) - Info_{theory}(D) = 0.918296 - 0.568622 = 0.349674(位)$$

$$Gain(machine) = Info(D) - Info_{machine}(D) = 0.918296 - 0.459773 = 0.458523(位)$$

step4 构造决策树. 属性上机成绩(machine)的信息增益最高，因此被选用于构建第一个节点，并根据不同的属性值(H，M，L)将样本分成三个子集. 对每一颗子树，递归调用上面的计算方法，最后得到决策树如图 1 所示.

2.3 提取规则

基于 Generate_Rules 算法，根节点 machine 有 1 个子节点 exercise 和两个叶子节点 machine->yes、machine->no,叶子节点分别生成“正例”规则 R1 和“反例”规则 R4. 在表 1 中，满足规则 R1 的记录占样本数的比例为 1/15(6.67%)，满足规则 R4 的记录占样本数的比例为 7/15(46.67%). 子节点 exercise 递归调用 Generate_Rules 算法，为叶子节点 exercies->yes 和

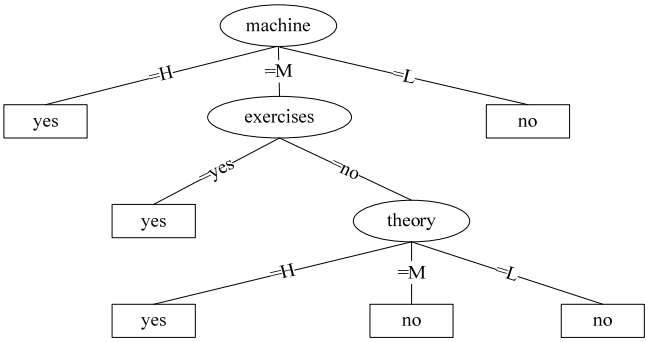

图 1 决策树

exerices->no 分别生成规则 R3、R5. 同理，子节点 theory 生成规则 R5、R6.

R1:IF 上机成绩=H THEN 通过=yes

R2:IF 上机成绩=M AND 按时完成作业=yes THEN 通过=yes

R3:IF 上机成绩=M AND 按时完成作业=no AND 理论成绩=H THEN 通过=yes

R4:IF 上机成绩=L THEN 通过=no

R5:IF 上机成绩=M AND 按时完成作业=no AND 理论成绩=M THEN 通过=no

R6:IF 上机成绩=M AND 按时完成作业=no AND 理论成绩=L THEN 通过=no

如果上机成绩高，则通过等级考试，通过率为 6.67%.

如果上机成绩中等，按时完成作业，则通过等级考试，通过率为 20%.

如果上机成绩中等，未按时完成作业，理论成绩高，则通过等级考试，通过率为 6.67%.

2.4 规则约简

规则 R 可以用它的覆盖率和准确率评价，给定类标记的数据集 D 中的一个元组 X，设 n_{covers} 为规则 R 覆盖的元组数，$n_{correct}$ 为规则 R 正确分类的元组数，|D| 是 D 中的元组数，则 $\mathrm{cov}erage(R)=\frac{n_{covers}}{|D|}$，$accuracy(R)=\frac{n_{correct}}{n_{covers}}$，依次计算“正例”规则的覆盖率和准确率:

coverage(R1)=6.67%,accuracy(R1)=100%

coverage(R2)=20%,accuracy(R2)=100%.

coverage(R3)=6.67%,accuracy(R3)=100%

coverage(R4)=46.67%,accuracy(R4)=100%.

coverage(R5)=6.67%,accuracy(R5)=100%

coverage(R6)=13.33%,accuracy(R6)=100%.

规则约简从三方面考虑，一是规则的长度 length(R)，二是规则的准确率 coverage(R)，三是规则的覆盖率 accuracy(R). 依次按照规则长度与准确率的乘积 leng(R)*accuracy(R)、规则的长度及覆盖率对规则集进行排序，length(R)*accuracy(R)大的优先度高于 length(R)*accuracy(R) 小的规则; 若两者相等，length(R)大的优先度高于 length(R)小的; 若前二者相等，则 accuracy(R)高的优先度高于 accuracy(R)小的规则; 若前者均相等，则覆盖率高的优先度高于覆盖率小的规则，如表 2 所示.

表 2 规则约简

优先级	规则	length(R)*accuracy(R)	Length (R)	Accuracy (R)	Coverage (R)
1	R6	3	3	1	0.1333
2	R3	3	3	1	0.0667
3	R5	3	3	1	0.0667
4	R2	2	2	1	0.2
5	R4	1	1	1	0.4667
6	R1	1	1	1	0.0667

3 真实数据

分别从我院非计算机专业 2009-2011 级学生等考数据库中随机抽取 600 条记录，按一定比例数据作为训练样本，其余作为预测数据，由训练样本构造决策树模型，用该模型预测数据，并与每届学生的真实通过情况进行对比，最终结果如表 3 所示.

表 3 仿真数据

年级	数据集(条记录)	训练样本比例	预测数据				预测准确率
			通过	不通过	正确	不正确	
2009	600	30%	291	129	363	57	86.4286%
2010	600	40%	253	107	310	50	86.1111%
2011	600	50%	207	93	257	43	85.6667%

4 结 语

鉴于以上结论，决策树用于预测学生等级考试成绩具有较高的准确率. 从挖掘的规则分析出，在计算机等级考试的课程教学过程中，需要强化上机练习，这与江苏省等级考试重点在上机操作是吻合的. 并要求学生按时完成作业，按时完成作业的学生通过的可能性较大. 而学生的计算机基础，对能否通过等级考试影响不大.

参考文献

1 段薇,马丽.基于信息增益和最小距离分类的决策树改进算法.科学技术与工程,2013,2(6):1671–1815.

2 吴铁洲,曾艺师.决策树分类算法在教学评估中的应用.中国高等教育评估,2013,6(2):24–26.

3 黄宇达.基于朴素贝叶斯与 ID3 算法的决策树分类.计算机工程,2012,14(7):41–44.

4 王守选,叶柏龙.决策树、朴素贝叶斯和朴素贝叶斯树的比较.计算机系统应用,2012,21(12):221–224

5 常秉琨.基于改进 ID3 的分类规则挖掘研究.微计算机信息,2009,5(12):218–220.

6 韩家炜.数据挖掘概念与技术.北京:机械工业出版社,2008.

7 孙林,徐久成.基于决策熵的决策树规则提取方法.计算机技术与发展,2007,2(6):97–100.