决策树中数值型属性分裂的研究

刘友军　汪林林
（重庆邮电学院　重庆 400065）

摘　要　在介绍了现有数值型属性分裂方法的基础上，引出了纯区间的概念，提出了一种基于纯区间归约的数值型属性分裂方法。该方法将属性值域用等宽直方图的方法划分为多个区间，对纯区间和非纯区间分别处理。理论分析和实验结果表明该方法在保证了分裂精度的同时，减小了搜索空间。

关键词　决策树，数值型属性，纯区间归约，Gini 指数

Research on Numeric Attributes Splitting in Decision Trees

LIU You-Jun　WANG Lin-Lin
(Chongqing University of Posts and Telecommunications; Chongqing400065)

Abstract　This paper introduces methods about numeric attributes splitting and the concept of pure interval, propose a new splitting method based on pure intervals reduction. The method divides the numeric attributes to many intervals with equal-width histogram, uses different methods to deal with the pure and impure intervals. Theoretical analysis and experimental results showe that the proposed method ensures the accuracy and narrowes the search space.

Keywords　Decision tree, Numeric attributes, Pure intervals reduction, Gini index

1　引言

决策树分类是一种重要的数据分类技术。在大数据集上构造决策树时，通常会遇到数值型属性，其值域是一个具有全序关系的实数集合或整数的子集。在选择一个数值型属性来扩展决策树的结点时，有时会因该属性的取值众多，不能对其每一个取值形成一个分支。一种处理方法是采用聚类、直方图等手段将数值型属性的取值进行概化，形成概念或分层概念，以此来归约数据。在归约后的数据上生成决策树，计算的复杂性较低，生成的决策树比较容易理解，但也容易导致偏差，降低结果的精度。

另外的一种处理方法是对属性值进行预排序，然后寻找一个分裂点，将属性值划分为两部分，形成两个分支，C4.5[1]、SLIQ[2]、STRINT[3]等算法都采用了这种技术。为了找到最佳分裂点，SLIQ、STRINT 中采用了精确查找技术，对每一个属性都进行计算评估。虽然得到的结果精确，但计算量大。在 CLOUDS[4]算法中，针对在 SLIQ、STRINT 中采用的精确技术计算量大的缺点，采用了一种将数值型属性的值域划分为多个等深区间，然后再估计在每个区间上有无找到最佳分裂点的可能，最后在可能找到最佳分裂点的区间中逐一搜索。这种方法增加了区间评估的计算量，同时要进行三次读写数据集，存在对大数据集的 I/O 问题。

针对 CLOUDS 区间评估算法的缺点，结合了数值型属性数据归约的优点，本文提出了一种基于纯区间数据归约的分裂方法。在引出纯区间的概念的基础上，阐述了纯区间分裂法的原理和方法，然后从算法的时间代价和精度方面与区间评估法进行了比较。

2　数值型属性的分裂

2.1　分裂指数

分裂指数是用来度量属性分裂规则优劣程度的一个量度。常用的分裂指数计算方法有：信息增益（information gain）、信息增益率（gain ratio）以及 gini 指数等。Gini 指数已经被证明了能够有效地搜索极佳分裂点，而这对于生成一棵好的决策树是至关重要的[2,3]。

设一个数据集 S 有 n 条记录，它们分属于 c 个互不相关的类，则集合 S 的 gini 值是：

$$gini(S)=1-\sum_{j=1}^{c}P_j^2 \tag{1}$$

其中 $p_j=m/n$，m 为 S 中属于类 j 的记录数。

如果使用分裂规则 $cond$ 将 S 划分为 S_1 和 S_2 两个子集，则该规则的度量值记为 $gini^D(S,cond)$，定义如(2)所示：

$$gini^D(S,cond)=\frac{n_1}{n}gini(S_1)+\frac{n_2}{n}gini(S_2) \tag{2}$$

其中 n_1、n_2 分别为 S_1、S_2 的记录数。

$gini^D(S,cond)$越小，表明分裂规则越好。

2.2　精确计算分裂

对于一个数值型属性 A，它的分裂形式为 $A\leqslant v$。所以，可以先对数值型属性排序，假设排序后的结果是 $v_1,v_2,\cdots,v_n$，因为分裂只会发生在两个结点之间，所以有 $n-1$ 种可能性。通常取中点$(v_i+v_{i+1})/2$作为分裂点。从小到大依次取不同的分裂点，取 gini 值最小的点一个作为最佳分裂点。

这种寻找最佳分裂点的方法能够找到最精确的分裂点。但是对于数值型属性，它要将每一个取值都作为分裂点来计算 gini 值，工作量很大，特别是对于超大数据集，当属性含有大量的不同取值时，效率非常低。

2.3　区间评估分裂

针对精确计算方法中的缺点，在 CLOUDS 中提出了一种处理数值型属性的快速方法。采用了一种将数值型属性的值域划分为多个等深区间，然后再估计在每个区间上有无找到最佳分裂点的可能，最后在可能找到最佳分裂点的区间中逐一搜索。

刘友军　助教，硕士研究生，主要研究方向为数据挖掘；汪林林　教授，主要研究方向为数据库系统、GIS 系统、数据挖掘系统。

一个区间的评估是估计以该区间的值作为分裂点，可能得到最小的 $gini^D$。设待划分的数据集为 S，$[v_l, v_u]$为一个区间，其中：

n：数据集 S 的大小；

c：S 中类的个数；

x_i：类 i 中小于或等于 v_l 的记录数；

y_i：类 i 中小于或等于 v_u 的记录数；

c_i：类 i 的所有记录数；

n_l：小于等于 v_l 的记录数(即为 $\sum_{i=1}^{c} x_i$)；

n_u：小于等于 v_u 的记录数(即为 $\sum_{i=1}^{c} y_i$)；

由式(2)得 $gini^D$ 在点 v_l 的值如下：

$$gini^D(S, a\leqslant v_l) = \frac{n_l}{n}\left(1-\sum_{i=1}^{c}\left(\frac{x_i}{n_l}\right)^2\right)+\frac{n-n_l}{n}\left(1-\sum_{i=1}^{c}\left(\frac{c_i-x_i}{n-n_l}\right)^2\right) \tag{3}$$

对给定的区间使用爬山法来估计其 $gini^D$ 下限值，记为 $gini^{est}$。首先，计算各个类沿 $gini^D$ 曲线的斜率的最小值，x_i 沿此区间斜率的计算公式为：

$$\frac{\partial gini(S,a\leqslant v_l)}{\partial x_i} = \frac{2}{n_l(n-n_l)}\left(c_i\frac{n_l}{n}-x_i\right)-\frac{1}{n}\left(\frac{1}{(n-n_l)^2}\sum_{i=1}^{c}(c_i-x_i)^2-\frac{1}{n_l^2}\sum_{i=1}^{c}x_i^2\right) \tag{4}$$

然后，计算取得最小斜率的类的 $gini^D$ 下限值。对于区间$[v_l, v_u]$，从左至右取得的下限值记为 Est_GiniLR，从右至左取得的下限值记为 Est_GiniRL，由式(5)得到区间的下限值 $gini^{est}$：

$$gini^{est} = \min(Est\_GiniLR, Est\_GiniRL, gini^D(S, a\leqslant v_l), gini^D(S, a\leqslant v_u)) \tag{5}$$

一般的情况下，一个区间的 $gini^{est}$ 值越小，越有可能在此区间找到最佳分裂点。为了便于区间的筛选，记录全部区间边界的最小 $gini$ 值，记为 $gini_{min}$，将下限值 $gini^{est} < gini_{min}$ 的区间作为候选区间，在候选区间中进行精确的查找，找到最佳分裂点。

这种区间评估的方法减小了 gini 值的计算量和 I/O 的复杂度，在区间数足够多的情况下能得到最佳的分裂点。同时，这种方法增加了区间评估的计算量，存在对大数据集的 I/O 问题。为此，我们提出了一种纯区间归约的分裂方法，避免了区间评估的过程，减小了数据集的读写，提高了寻找最佳分裂点的效率。

3 纯区间归约分裂

3.1 分裂原理

定义 1 对于某个数值型属性，如果在区间$[v_b, v_t]$中的所有记录都属于同一个类 C_i，则称该区间为 C_i 的纯区间。

定理 1 设 $f(x)$在$[a, b]$上连续，在(a, b)内具有一阶和二阶导数，那么

(1)若在(a, b)内，$f''(x)>0$，则 $f(x)$在$[a, b]$上的图形是凹的；

(2)若在(a, b)内，$f''(x)<0$，则 $f(x)$在$[a, b]$上的图形是凸的；

对类 C_k 的一个纯区间$[v_b, v_t]$，在式(3)中，只有 x_k 变化，令 $x_k=x$，$c_k=C$，则式(3)可以转化为一个关于 x 的函数。对于式(3)，令：

$n_l=\sum_{i=1}^{c} x_i = A+x$，$\sum_{i=1}^{c} x_i^2 = B+x^2$，$\sum_{i=1}^{c}(c_i-x_i)^2 = D+(C-x)^2$

其中 A、B、C、D 均为大于等于 0 的常数，则式(3)转化为关于 x 的函数，如式(6)所示：

$$f(x)=1-\frac{B+x^2}{n(A+x)}-\frac{D+(C-x)^2}{n(n-A-x)} \tag{6}$$

式(6)的一阶导数为：

$$f'(x)=\frac{1}{n}\left(\frac{B+x^2}{(A+x)^2}-\frac{2x}{A+x}-\frac{D+(C-x)^2}{(n-A-x)^2}+\frac{2(C-x)}{(n-A-x)}\right) \tag{7}$$

式(6)的二阶导数为：

$$f''(x)=-\frac{2}{n}\left(\frac{A^2+B}{(A+x)^3}+\frac{(n-A-C)^2+D}{(n-A-x)^3}\right) \tag{8}$$

由于 A、B、C、D 均为大于等于 0 的常数，$(A+x)^3$ 和$(n-A-x)^3$ 是大于 0 的数，因此有 $f''(x)<0$。根据定理 1 可知，$f(x)$在纯区间$[v_b, v_t]$上是上凸函数。所以，式(3)在 C_k 的一个纯区间$[v_b, v_t]$内的极小值只可能出现在区间的边界点处。这样就只需要计算纯区间边界上的 gini 值，可以得到该纯区间的极小值，减小了计算量。

数值型属性一般都服从高斯分布[5]，在等宽划分的区间中存在大量的纯区间。因此，对于纯区间，只要计算各个纯区间的边界点处的 gini 值；对于非纯区间进行精确计算，就能得到整个属性值的最小 gini 值 $gini_{knw}$，找到最佳分裂点。

3.2 分裂方法

对数据集 S 中的某一个数值型属性分裂，分裂方法如下：

(1)由于数值型属性的分类一般服从高斯分布，因此用等宽直方图方法将属性值分为 q 个区间，同时构造区间直方图列表。区间直方图列表的字段有区间的左边界、右边界的值和在该区间中各个类的记录数。由于区间直方图列表较小，它可以存放在主存中，提高计算效率。

(2)对每一个区间计算 gini 值，并找出最小值 $gini_{knw}$：

i)对于纯区间$[v_b, v_t]$，则计算区间右边界处的 gini 值 $gini^D(S, a\leqslant v_t)$；

ii)对于非纯区间，先对区间进行排序，然后精确计算在该区间最小 gini 值。

(3)用最小 gini 值 $gini_{knw}$ 对属性表进行分裂。

举例说明上述的分裂方法。设有一个数据集 S，如表 1 所示，对 Salary 属性进行分裂。若直方图的宽度为 30，可以将属性值域分为[1,30]、[31,60]和[61,90] 等 3 个区间，建立如表 2 所示区间直方图列表。

表 1 数据集

Age	Salary	Class	Tid
30	65	G	1
25	20	B	2
50	85	G	3
40	75	G	4
45	60	G	5
52	40	B	6
23	15	B	7

表 2 区间直方图列表

左边界	右边界	B	G
1	30	2	0
31	60	1	1
61	90	0	3

利用式(3)计算各个区间的 gini 值：

在区间[1,30]，由于该区间是纯区间，计算该区间右边界

处的 gini 值：

$gini^D(S, Salary \leqslant 30) = 0.228571$；

在区间[31,60]，由于该区间是非纯区间，先对该区间进行排序并计算最小 gini 值：$gini^D(S, Salary \leqslant 50) = 0$；

在区间[61,90]，由于该区间是纯区间，计算该区间右边界处的 gini 值：

$gini^D(S, Salary \leqslant 90) = 0.489796$；

最后得到 Salary 属性的最小 gini 值 $gini_{low} = 0$，同时获得最佳分裂点为 50。

4 性能评价

数值型属性的分裂的主要评价指标有计算的时间代价和分裂结果的精度。下面将本文提出的纯区间归约法同区间评估方法进行比较。

4.1 时间代价

假设数据集 S 有 n 条记录，它们分属于 c 个互不相关的类，划分为 q 个区间，每个区间的记录数为 n_i；在区间评估法选取的候选区间数为 a，在纯区间归约法中非纯区间数为 b。

在区间评估法中时间代价如下：

(1)将数据集划分为 q 个区间的时间代价为 $O(n\log q)$；

(2)计算每一个区间边界中各个类的记录数的时间代价为 $O(n\log q)$；

(3)估算每一个区间边界的 gini 值的时间代价为 $O(qc)$；

(4)评估每一个区间的下限值的时间代价为 $O(qc^2)$；

(5)决定每一个记录所属的候选区间、对候选区间进行排序和在候选区间中的计算精确的 gini 值的时间代价为 $O(n\log q + \sum_{i=1}^{a} n_i \log n_i + n_i c)$；

在纯区间归约法中的时间代价如下：

(1)将数据集划分为 q 个区间，建立直方图的时间代价为 $O(n\log q)$；

(2)估算每一个区间边界的 gini 值的时间代价为 $O(qc)$；

(3)决定每个非纯区间的记录、对每个非纯区间排序和计算每个非纯区间中的精确 gini 值的时间代价为 $O(n\log q + \sum_{i=1}^{b} n_i \log n_i + n_i c)$；

从总体上看，虽然这两种算法总的时间代价都是 $O(n)$，但是区间评价法要对数据集进行两次读和一次写操作，纯区间归约法只要对数据集进行两次读操作，同时减小了区间下限值评估的过程。

在这两种方法中，划分的区间数 q 的取值对算法都有一定的影响。在区间评估法中 q 的取值小的话，实验证明，它将直接影响到计算的精度。在纯区间归约法中，q 的取值小的话，虽然对精度没有影响，但将增加非纯区间的数量，增加了排序时间代价。因此，在纯区间归约法中，划分的区间数要适当选取大些，可以增加纯区间数的比例，减小区间内排序的时间开销。

4.2 分裂精度

对于两种方法的分裂精度，笔者通过实验进行了验证。实验数据集的前四项取自分类领域中广泛使用的评价标准 STATLOG，“Synth1”和“Synth2”是分别用来评价 SLIQ 算法和 STRINT 算法的生成数据集。笔者分别使用了精确计算 gini 值的方法、区间评估法和本文提出的纯区间归约法对实验数据集进行了计算，结果如表 3 所示。

表 3 分裂精度实验结果

数据集名	精确的 gini 值	区间评估的区间数			区间归约法
		100	50	10	
Letter	0.940323	0.940323	0.940323	0.941751	0.940323
Satimage	0.653167	0.653167	0.653167	0.653167	0.653167
Segment	0.714286	0.714286	0.714286	0.715799	0.714286
Shuttle	0.175777	0.175777	0.175777	0.175777	0.175777
Synth1	0.546150	0.546150	0.546150	0.546150	0.546150
Synth2	0.541134	0.541134	0.541134	0.541134	0.541134

从表 3 可以看出，区间评价法在划分的区间数较小的时候，对部分数据集在精度上有一定的影响。纯区间归约法计算出来的 gini 值和精确计算得到的值是一致的，保证了属性分裂的精度。

结论 纯区间归约法利用了 gini 指数函数在纯区间上是凸函数和数值型属性一般服从高斯分布的特点，对纯区间进行归约。该方法在保证了分裂结果的精度的同时，减小了搜索空间。实验结果表明该方法是一种有效的分裂方法，可以采用该方法改进 SPRINT 等决策树分类算法。当然，划分的区间数对纯区间归约法的计算时间代价有一定的影响，如何有效地选取区间数，以便更好地减小非纯区间的数量及降低对非纯区间的排序代价，有待进一步的优化。

参 考 文 献

1 Ruggieri S. Efficient C4.5. IEEE Transactions on Knowledge and Data Engineering[J],2002,14(2)

2 Mehta M, Agrawal R, Rissanen J. SLIQ: A Fast Scalable Classifier for Data Mining[A]. In: Proc. of the 5th Int'l Conf. on Extending Database Technology (EDBT)[C], Avignon, France, March 1996

3 Shafer J, Agrawal R, Mehta M. SPRINT: A Scalable Parallel Classifier for Data Mining[A]. In: Proc. of the 22th Int'l Conf. on VLDB[C], Bombay, India. Sept. 1996

4 Alsabti K, Ranka S, Singh V. CLOUDS: A Decision Tree Classier for Large Datasets[A]. In: Proc. of the 4th Int'l Conf. on Knowledge Discovery and Data Mining[C],1998

5 Han J, Kamber M. Data Mining: Concepts and Techniques[M]. Beijing: High Education Press, 2001.279～301

6 Wang H, Zaniolo C. CMP: A Fast Decision Tree Classier Using Multivariate Predictions[A]. In: Proc. of the 16th Int'l Conf. on Data Engineering[C],2000

7 Agrawal R, Ghosh S, Imielinski T, Iyer B, Swami A. An interval classifier for database mining applications[A]. In: Proc. of the VLDB Conference[C]. Vancouver, British Columbia, Canada. August 1992

8 Catett J. Megainduction: Machine Learning on Very Large Databases[D]:[PhD thesis]. University of Sydney. 1991

9 Rajeev R, Kyuseok S. PUBLIC: A Decision Tree Classifier that Integrates Pruning and Building[A]. In: Preceedings of 24th VLDB Conference[C]. New York, 1998