问题

ID3、C4.5、CART算法总结与对比

前言

ID3、C4.5、CART算法是三种不同的决策树算法，区别主要在最优划分属性的选择上，下面把之前在随机森林中汇总过的复制过来，然后再总结下三者的不同。

三种算法所用的最优属性选择方法详述

信息增益 (ID3决策树中采用)

“信息熵”是度量样本集合纯度最常用的一种指标，假定当前样本结合 D 中第 k 类样本所占的比例为 $p_k(k=1,2,\ldots,c)$ ，则 D 的信息熵定义为：

$$Ent(D)=-\sum_{k=1}^{c}p_k log_2 p_k \tag{1}$$

$Ent(D)$ 的值越小，则 D 的纯度越高。注意因为 $p_k \leq 1$ ，因此 $Ent(D)$ 也是一个大于等于 0 小于 1 的值。

假定离散属性 a 有 V 个可能的取值 $\{a^1,a^2,\ldots,a^V\}$ ，若使用 a 来对样本集合 D 进行划分的话，则会产生 V 个分支结点，其中第 v 个分支结点包含了 D 中所有在属性 a 上取值为 a^v 的样本，记为 D^v 。同样可以根据上式计算出 D^v 的信息熵，再考虑到不同的分支结点所包含的样本数不同，给分支结点赋予权重 $\frac{|D^v|}{|D|}$ ，即样本数越多的分支结点的影响越大，于是可以计算出使用属性 a 对样本集 D 进行划分时所获得的“信息增益”：

$$Gain(D,a)=Ent(D)-\sum_{v=1}^{V}\frac{|D^v|}{|D|}Ent(D^v) \tag{2}$$

一般而言，信息增益越大越好，因为其代表着选择该属性进行划分所带来的纯度提升，因此全部计算当前样本集合 D 中存在不同取值的那些属性的信息增益后，取信息增益最大的那个所对应的属性作为划分属性即可。

缺点：对可取值数目多的属性有所偏好

增益率 (C4.5决策树中采用)

从信息增益的表达式很容易看出，信息增益准则对可取值数目多的属性有所偏好，为减少这种偏好带来的影响，大佬们提出了增益率准则，定义如下：

$$Gain\_ratio(D,a)=\frac{Gain(D,a)}{IV(a)} \tag{3}$$

$$IV(a)=\sum_{v=1}^{V}\frac{|D^v|}{|D|}log_2\frac{|D^v|}{|D|}$$

$IV(a)$ 称为属性 a 的“固有值”。属性 a 的可能取值数目越多，则 $IV(a)$ 的值通常会越大，因此一定程度上抵消了信息增益对可取值数目多的属性的偏好。

缺点：增益率对可取值数目少的属性有所偏好

> 因为增益率存在以上缺点，因此C4.5算法并不是直接选择增益率最大的候选划分属性，而是使用了一个启发式：**先从候选划分属性中找出信息增益高于平均水平的属性，再从中选择增益率最高的。**

基尼指数 (CART决策树中采用)

ID3中根据属性值分割数据，之后该特征不会再起作用，这种快速切割的方式会影响算法的准确率，因为这是种贪心算法，不能保证找到全局最优值。CART是一棵二叉树，采用二元切分法，每次把数据切成两份，分别进入左子树、右子树。而且每个非叶子节点都有两个孩子，所以CART的叶子节点比非叶子多1。相比ID3和C4.5，CART应用要多一些，既可以用于分类也可以用于回归。

这里改用基尼值来度量数据集 D 的纯度，而不是上面的信息熵。基尼值定义如下：

$$Gini(D) = \sum_{k=1}^{c}\sum_{k'\neq k} p_k p_{k'} = 1 - \sum_{k=1}^{c} p_k^2 = 1 - \sum_{k=1}^{c}(\frac{D^k}{D})^2 \tag{4}$$

直观来看，$Gini(D)$ 反映了从数据集 D 中随机抽取两个样本，其类别标记不一致的概率，因此 $Gini(D)$ 越小，则数据集 D 的纯度越高。

对于样本D，个数为|D|，根据特征A的某个值a，把D分成|D1|和|D2|，则在特征A的条件下，样本D的基尼系数表达式为：

$$Gini\_index(D,A) = \frac{|D^1|}{|D|}Gini(D^1) + \frac{|D^2|}{|D|}Gini(D^2) \tag{5}$$

于是，我们在候选属性集合A中，选择那个使得划分后基尼系数最小的属性作为最优划分属性即可。

三种算法对比总结

下面是根据自己的理解整理的，不知道全不全，应该差不多了。

ID.3

1. **最优划分属性选择方法**：信息增益
2. **分支数**：可多分支
3. **能否处理连续值特征**：不能
4. **缺点**：偏好与可取值数目多的属性

C4.5

1. **最优划分属性选择方法**：增益率
2. **分支数**：可多分支
3. **能否处理连续值特征**：能，C4.5 决策树算法采用的**二分法**机制来处理连续属性。对于连续属性 a，首先将 n 个不同取值进行从小到大排序，选择相邻 a 属性值的平均值 t 作为候选划分点，划分点将数据集分为两类，因此有包含 n－1 个候选划分点的集合，分别计算出每个划分点下的信息增益，选择信息增益最大对应的划分点，仍然以信息增益最大的属性作为分支属性。
4. **缺点**：增益率对可取值数目少的属性有所偏好，因此C4.5算法并不是直接选择增益率最大的候选划分属性，而是使用了一个启发式：**先从候选划分属性中找出信息增益高于平均水平的属性，再从中选择增益率最高的。**

CART

1. **最优划分属性选择方法**：基尼系数
2. **分支数**：二叉树

3. **能否处理连续值特征**：能，做法与C4.5一样。也可以用于回归，用于回归时通过最小化均方差能够找到最靠谱的分枝依据，回归树的具体做法可见机器学习的问题33。
4. **优点**：与ID3、C4.5不同，在ID3或C4.5的一颗子树中，离散特征只会参与一次节点的建立，但是在CART中之前处理过的属性在后面还可以参与子节点的产生选择过程。

参考资料

决策树算法原理 (CART决策树)
《机器学习》周志华
决策树模型 ID3/C4.5/CART算法比较