第4章 决策树

4.1 决策树算法(递归)

递归终止:

① 当前结点样本属同类别,无需划分

② 属性为空,或所有样本在所有属性取值相同:无法划分

③ 当前结点的样本集合为空,不能划分

④ 利用当前结点后验分布

⑤ 父结点的样本分布作为当前结点的后验分布

4.2 划分选择

关键都在于选择划分属性(结点纯度要高)

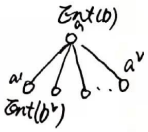
4.2.1 信息增益

D中k类占 p_k,则 $Ent(D) = -\sum_{k=1}^{|y|} p_k \log_2 p_k$

$Ent(D)$ ↓ 纯度↑

设属性a有V个取值{$a^1, a^2, \cdots, a^V$}

共有V个结点,每个结点计算 D^v 的信息熵(D^v包含 a^v 取值的样例)



然后加上权重 $\frac{|D^v|}{|D|}$

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

Gain越大,"纯度提升"越大

4.2.2 增益率

☆ 注意:信息增益准则对取值数目多的属性有偏好!

C4.5 使用信息增益率

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

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

通常而言,a的取值数目越多,则IV(a)通常越大

所以,增益率对数目较少的属性有偏好 ☆

所以C4.5使用启发式:先选信息增益高于平均水平的,再从这些中选信息增益率较高的

4.2.3 基尼指数

基尼指数 $Gini(D) = 1 - \sum_{k=1}^{|y|} p_k^2$

随机抽两个样本,不一致的概率

Gini(D)↓ → 纯度越高

$$Gini\_index(D,a) = \sum_{v=1}^{V} \frac{|D^v|}{|D|} Gini(D^v)$$

选择使得划分后Gini最小的作为最优属性 $a^* = \arg\min_a Gini\_index(D,a)$

4.3 剪枝处理

防止过拟合 → "贪心" 比较划分前/划分后验证集精度

预剪枝:划分前先估计,划分能否提升泛化

后剪枝:先得到完整的决策树,自底向上,将结点替换成叶结点能否提升泛化

如何估计泛化性能:留出法 {训练、验证

预:后剪枝较预剪枝:保留了更多的分枝,欠拟合风险更小,但训练时间开销更大

4.4 连续与缺失值

"二分法"

连续属性a出现了n个取值{$a^1, \cdots, a^n$}

n-1个候选划分点 $T_a = \left\{ \frac{a^i + a^{i+1}}{2} \mid 1 \le i \le n-1 \right\}$

再依基于划分点二分

$$Gain(D,a) = \max_{t \in T_a} Gain(D,a,t) = \max_{t \in T_a} Ent(D) - \sum_{\lambda \in \{-,+\}} \frac{|D_t^\lambda|}{|D|} Ent(D_t^\lambda)$$

☆ 当属性值缺失时,当前结点划分为连续属性:正反还可使用