**决策树基本算法**

输入：训练集 D = {(x1,y1),(x2,y2),…, (xm,ym)};

输入集 A ={ a1, a2,…, ad}

过程：函数TreeGenerate(D,A)

1：生成结点node；

2：if D 中样本全属于同一类别C then （如全为好瓜，不再分类）

3： 将node标记为C类也结点；return

4：end if

5：if A= OR D中样本在A上取值相同 then （A为空表示集合中的属性分配完了或没有属性，则只看样本中好瓜和坏瓜的数量，如果好瓜多于坏瓜则标记为好瓜，反之为坏瓜）

6： 将node标记为叶节点，其类别标记为D中样本数最多的类；return

7：end if

8：从A中选择最优划分属性a\* ;（如 纹理/根蒂/色泽等属性之一）

9：for a\* 的每一个值a\*v do （假设 a\*是纹理，则a\*v 可假设为清晰）

10： 为node生成一个分支；令Dv 表示D中在a\* 上取值为a\*v 的样本子集；

11： if Dv 为空 then（表示在前结点下取值a\*v的样本不存在，显然根结点不存在这种情况）

12： 将分支结点标记为叶结点，其类别标记为D中样本最多的类；return

13： else

14： 以 TreeGenerate(Dv , A\{a\*}) 为分支结点 （把Dv作为新的训练集，并属性集A扣除当前的属性值作为新属性集A 继续递归）

15： end if

16：end for

输出： 以node为根结点的一颗决策树。

如图：

