17.6.1（决策树，学习新语言的五点）

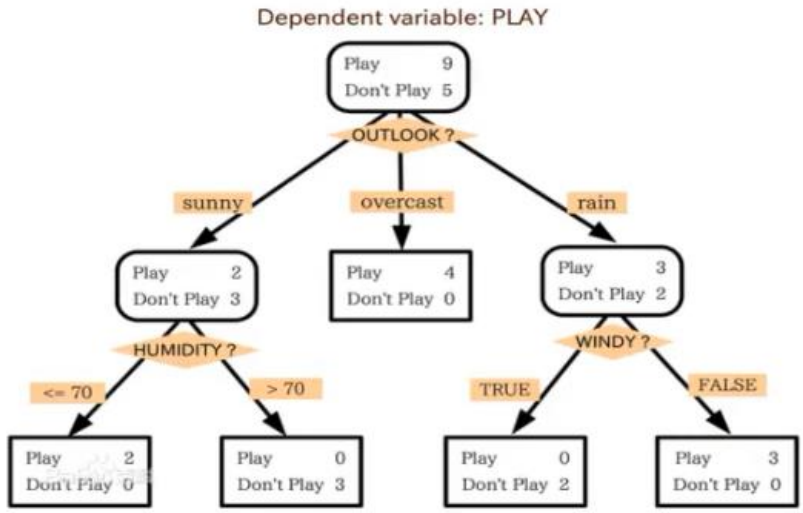
学习一门新的编程语言一般要掌握的五项:

1. 定义类,属性,函数的方法,实例化类创建对象,调用对象中函数的方法.
2. 控制结构,for循环,while循环,if--else--判断等.
3. 集合的使用,list列表,set集合,map
4. 形参和实参的使用(值传递,引用传递)
5. 多线程的使用

注:形参和实参的结合是在调用函数时.

决策树（decision tree）

决策树是类似于流程图的一个结构，每一个内部节点都是一个属性的测试（按照分类依据进行分类）。最顶点就是根节点。



我们要搞清楚一件事情需要很多的信息，针对这个信息量的度量，1948年，香农提出了 ‘“信息熵（entropy）”的概念，变量的不确定性越大，熵就越大。

公式就是所有情况发生概率乘以log（所有情况发生概率）之和。

$$H(X) = -\sum_{x} P(x) \log_2 [P(x)]$$

决策树归纳算法（ID3）1970-1980

ID3算法就是用来求每一个节点的分类依据，用来判断节点的方法是比较各个分类依据的信息获取量（information gain）：

Gain(A) = Info(D) - Infor_A(D)

公示的意思就是：不用A分类的信息熵减去用A分类的信息熵，得到的就是A的信息获取量。求出来每个分类依据的信息获取量，最大的那个就作为根节点。根据这个分类依据分成两类，然后每一类继续重复以上方法求得二层的节点，以此类推。直到：给定的节点下都属于同一类；没有剩余的属性（分类依据）用来进行下一步，这时候，使用多数表决。

其他决策树算法：

C4.5: Quinlan

Classification and Regression Trees (CART): (L. Breiman, J. Friedman, R. Olshen, C. Stone)

共同点：都是贪心算法，自上而下(Top-down approach)

区别：属性选择度量方法不同： C4.5 （gain ratio), CART(gini index), ID3 (Information Gain)

CART算法是用基尼指数进行分类的。

$$Gini(D) = 1 - \sum_{i=1}^{m} p_i^2$$

基尼指数是衡量数据不纯度的方法，所以分类越多，基尼指数越大，只有一类时基尼指数为0.同样的，用分类前后的基尼指数之差作为选取节点的依据。

使用决策树，所有的属性值必须是离散的。连续性的变两个要把它离散化。

4. 树剪枝叶 （避免overfitting）
   4.1 先剪枝
   4.2 后剪枝

决策树中也有过拟合的情况，在决策树算法中有两种解决方法，先剪枝：创建树的时候根据一定的阈值来限制我们树的创建，比方说节点下分类的比例达到一定程度就不再往下分枝了。后剪枝就是我们先把决策树完完全全给创建出来，然后再进行摘除相对于我们的阈值多余的节点。

决策树的优缺点：

5. 决策树的优点：
   直观，便于理解，小规模数据集有效

6. 决策树的缺点：
   处理连续变量不好
   类别较多时，错误增加的比较快
   可规模性一般（