# 决策树作业

**2150248-姚天亮-自动化**

## 一、实验要求

试用python编程实现基于C4.5决策树算法来进行最优划分属性选择的决策树，并为表4.3 西瓜数据集3.0中去掉“密度属性和编号为9的西瓜”以后的数据生成一棵决策树。

## 二、实验原理

决策树算法是一种超参数学习算法。它通过计算每个属性的信息增益,选择信息增益最大的属性作为每个内部节点的切分条件,从而进行数据集的重复切分。决策树算法的原理是基于信息熵进行划分选择。

信息熵是度量样本集合纯度最常用的一种指标，假设当前样本集合 D 中第 k 类样本所占的比例为 pk，则 D 的信息熵定义为：Ent(D)= - ∑kp_klog_2p_k。

信息增益是在已知某个属性的条件下，集合 D 的信息熵与该属性的熵之差，即 Gain(D,a)=Ent(D) - ∑v|D_v|/|D|*Ent(D_v)，其中 a 是某个属性，v 是 a 的一个取值，D_v 是 D 中在 a 上取值为 v 的样本子集，|D_v|是 D_v 的样本个数。信息增益越大，说明使用属性 a 来进行划分所获得的“纯度提升”越大，应该优先选择信息增益大的属性来进行划分。

在构建决策树的过程中，每次都选择信息增益最大的属性作为某个内部节点的切分属性。这样重复进行属性切分和数据集细分，可以把相似样本分到同一个子节点，不同样本分到不同子节点,最大限度地减少每个子节点中的混杂程度，从而获得一个优化的决策树模型。

## 三、代码说明

本例中采用 python 语言，利用 math 库完成相关设计。

代码主要分为以下几个部分，

**STAGE 1.** 数据的预处理，输入西瓜数据集并对其进行简单操作以方便后续使用。

**STAGE 2.** 是计算样本集合的信息熵函数。首先， 它通过遍历样本集合中的数据，统计每个类 别标签出现的次数，然后计算每个类别标签出现的概率，最后根据信息熵的公式计算样本集 合的信息熵。

**STAGE 3.** 是计算信息增益，在该函数中，我们实现了对于离散属性和连续属性的信息增益计 算方法。对于离散属性，我们可以直接统计每个属性值对应的样本数和标签数， 然后计算出 每个属性值对应的信息熵，最后将所有属性值的信息熵加权平均即可得到该属性的信息增益。 对于连续属性，我们需要先使用二分法寻找最佳的分割点，然后将样本分为两部分，分别计 算每部分的信息熵， 最后将两部分的信息熵加权平均即可得到该属性的信息增益。

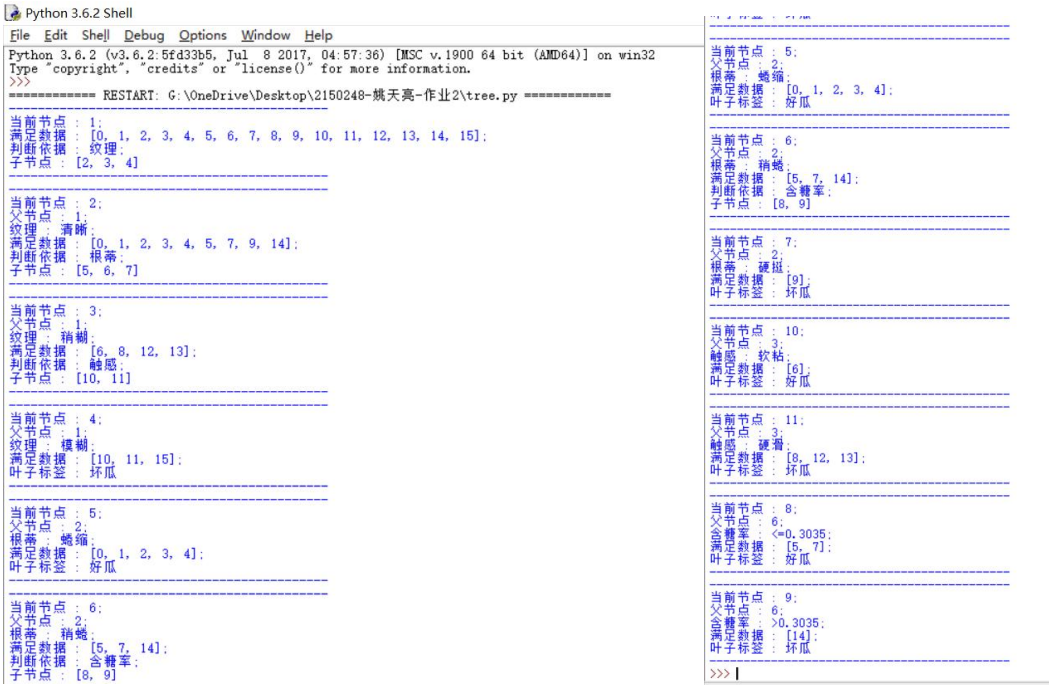
**STAGE 4.** 构造决策树，这部分实现了一个决策树的构建过程，其中包括了选择属性、划分 子节点等操作。具体来说， 我们构造了一个递归函数 finish_node，该函数接受一个当前结点 current_node、数据集 data、数据集的 label 以及剩余可用属性 rest_title 作为输入， 使用 id3 方法，最后输出为构建好的决策树。在函数中，首先判断当前结点的数据是否属于同一类， 如果是，直接标记为叶子结点并返回；否则，选择信息增益最大的属性作为当前结点的属性， 并根据该属性的值是否为连续数值进行不同的处理。如果该属性的值为连续数值，则根据该 属性的分隔值将数据集划分为两个子集，并分别构建两个子节点；如果该属性的值为离散值， 则根据该属性的每个取值将数据集划分为多个子集，并分别构建多个子节点。最后， 对于每 个子节点， 递归调用 finish_node 函数进行进一步的构建。

**STAGE 5.** 输出所构造的决策树。初版程序尝试使用 turtle 库进行绘制，绘制效果基本如下图。

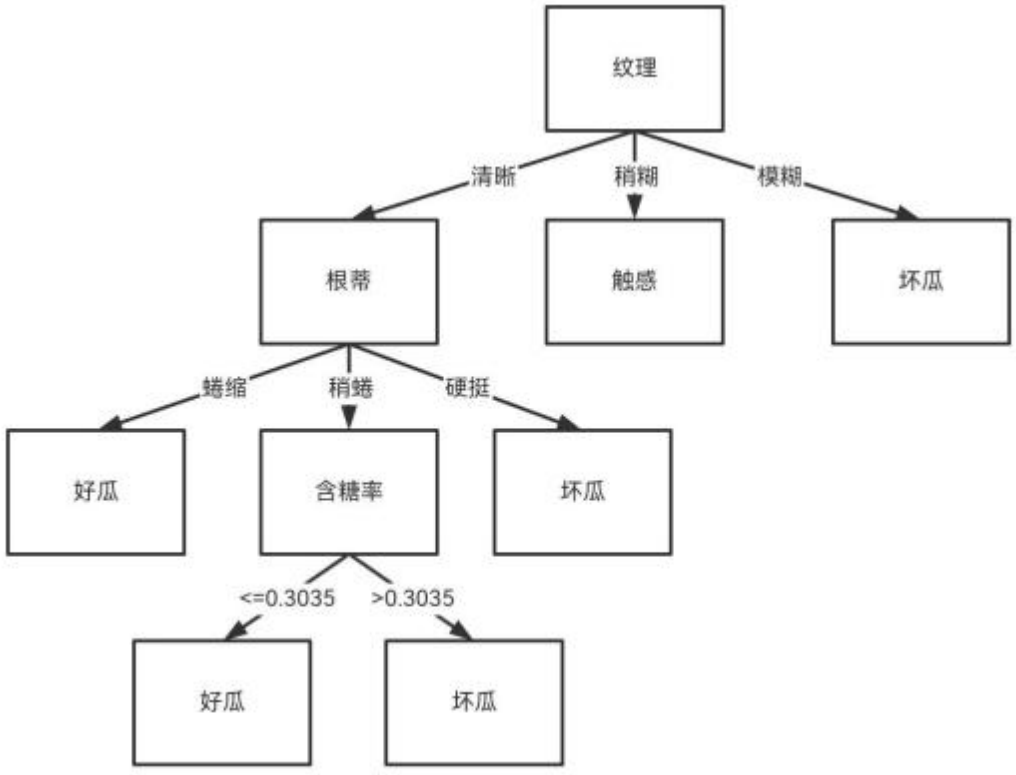
但由于本例中采用的数据集特征采用字符串和数字的混合，对于 turtle 绘图树枝位置的判断造成较大影响，最后采用文字描述，记录每个节点的父节点、子节点、判断特征、满足该节点的数据以及叶子标签（若节点为叶子）。

## 四、实验结果

运行程序得到如下结果：




转化为图像表示为：




## 五、心得体会

本次实验中，我设计实现了基于C4.5决策树算法的西瓜数据分类决策树构建。在设计时参考了决策树算法的工作原理，明确了关键步骤。在实现时，我设计了信息熵和信息增益的计算函数，利用这两个概念选择每个内部节点的最优属性。然后通过递归函数完成决策树的构建过程。

在此次实验中，我学会了利用信息论概念构建决策树分类模型。掌握了决策树算法的基本流程，理解了递归思想在算法实现中的应用。这为后续更多机器学习算法的学习奠定了基础。同时，我也明白了算法设计需要考虑各个细节,测试运行结果才是衡量算法是否正确的重要标准。今后的学习中，我会注意算法思路的解析，代码实现的完整性，以及结果的验证，以有效提升自身的专业水平。