**决策树算法实验**

**一、实验目的**

1. 理解决策树算法的基本原理和信息增益计算方法。
2. 掌握递归构建决策树的实现逻辑。
3. 应用决策树解决分类问题，并评估模型性能。

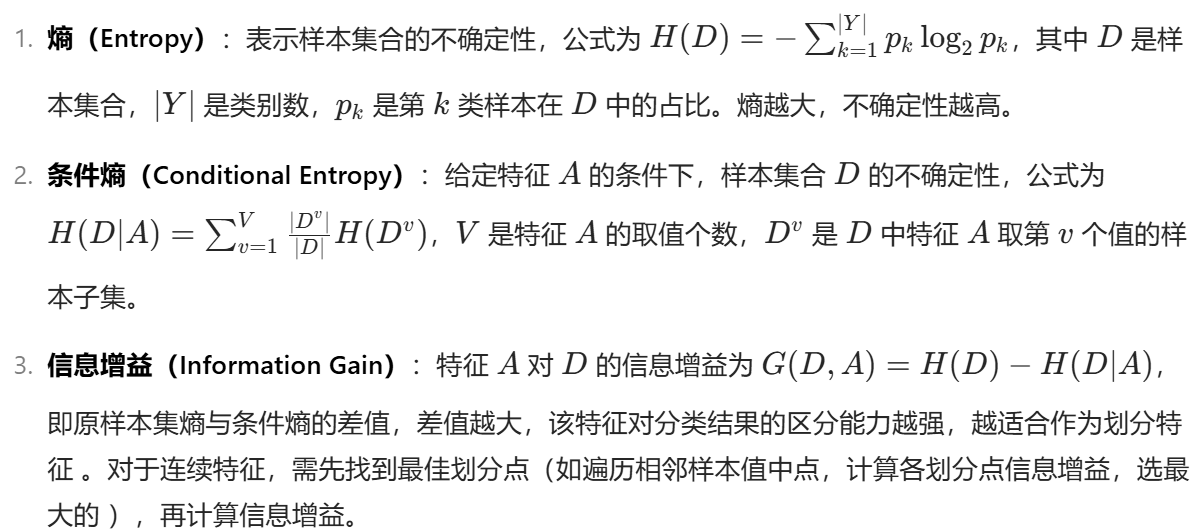
**二、实验内容与步骤**

1. 使用经典数据集（西瓜数据集3.0）。
2. 实现熵（Entropy）的计算函数，计算每个特征的信息增益，选择最优划分特征。
3. 递归构建决策树
4. 模型评估
   1. 划分训练集和测试集（如 80:20）。
   2. 使用测试集计算准确率、精确率、召回率等指标。
   3. 分析决策树深度对性能的影响（过拟合 / 欠拟合）。
5. （选做）决策树可视化。
6. **算法原理：**简述决策树的基本原理和信息增益计算方法
7. 决策树基本原理：

决策树是一种基于树结构进行决策的监督学习算法，可用于分类和回归任务 。它以递归方式构建，从根节点开始，每个内部节点对应一个特征的判断，根据特征取值将样本划分到不同子节点，叶节点为分类结果（或回归值）。构建过程中，通过选择最优特征对样本集进行划分，使划分后子节点的样本尽可能 “纯净”（即同类样本占比高），常用信息增益等指标选择特征，重复此过程直到满足停止条件（如样本标签相同、无特征可用等）。

1. 信息增益计算方法：

信息增益用于衡量特征对样本集合纯度的提升能力。



1. **实现细节：**信息增益计算的实现思路

**对离散特征：**

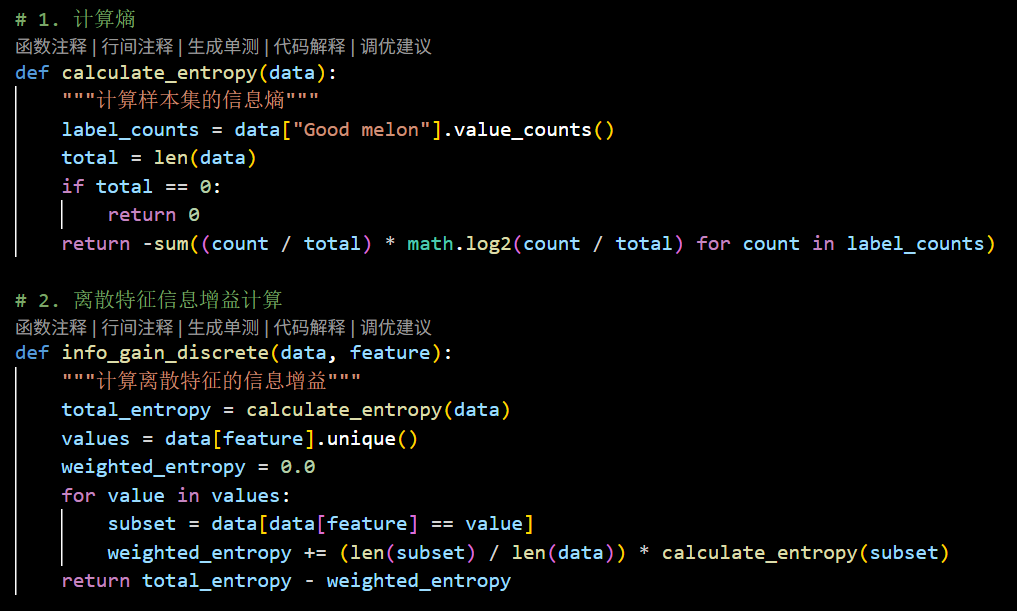
* 1. 先调用 calculate\_entropy 函数计算整个样本集的熵 H(D)。
  2. 遍历该离散特征的所有不同取值，对每个取值，筛选出对应样本子集，计算子集的熵 H()，并根据子集样本数占总样本数的比例，加权求和得到条件熵 H(D|A)。
  3. 最后用 H(D) - H(D|A) 算出该离散特征的信息增益，通过 info\_gain\_discrete 函数实现。

图1 离散特征的信息增益计算

**对连续特征：**

* 1. 同样先算整个样本集的熵 H(D)。
  2. 将样本按连续特征值排序，遍历相邻样本（仅在标签变化处尝试划分，优化计算），计算中间划分点，把样本分为两子集，分别算子集熵，加权得到条件熵，进而算出信息增益。同时记录最大信息增益及对应划分点，通过 info\_gain\_continuous 函数实现。

这样可找到对分类结果区分最有效的连续特征划分方式，用于构建决策树节点。

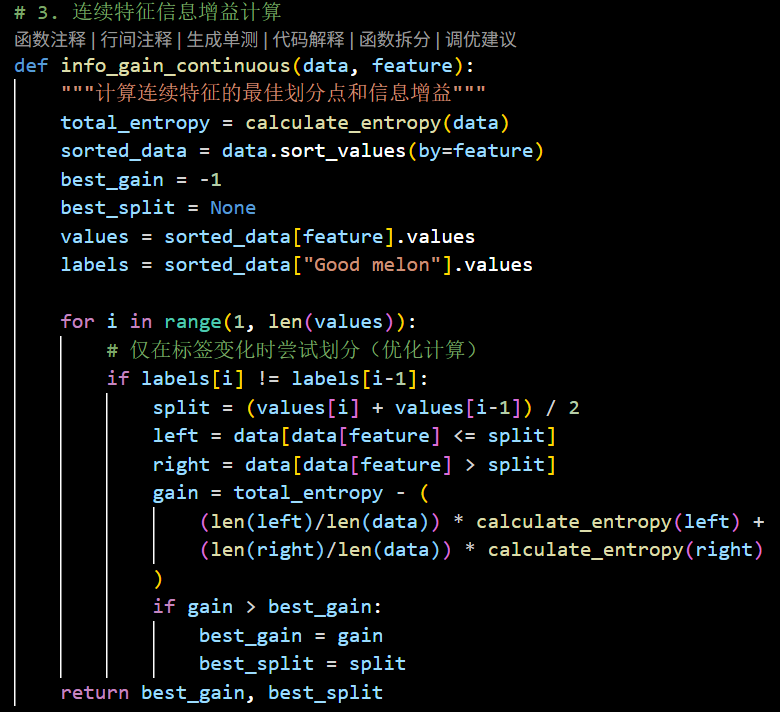


图2 连续特征的信息增益计算

1. **实验结果**
   * 1. 树的可视化结果（采用Mermaid 绘制）

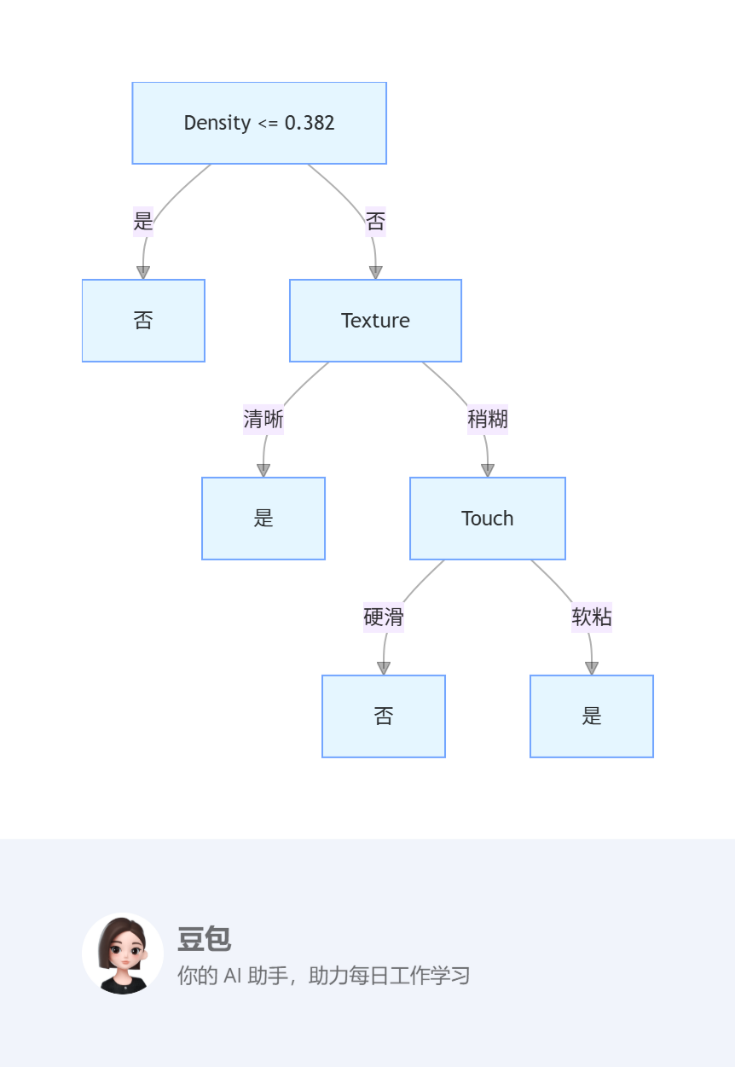


图3 西瓜数据集分类决策树

2、模型性能指标展示

划分训练集（80%）与测试集（20%）评估模型。预剪枝（max\_depth=3）模型经evaluate\_model计算，准确率、精确率、召回率、F1 分数均为1.0000，说明对西瓜 “好瓜”“坏瓜” 分类精准，模型拟合效果较好。

3、不同参数对结果的影响（以树深度为例）

设置最大深度为 3、5、10 ，调用analyze\_overfittin测试。各深度下，模型在测试集的准确率、精确率、召回率、F1 分数均保持 1.0000。深度增加时，未出现准确率下降情况，表明当前实验数据及设置下，模型未因深度增大而过拟合，也未因深度小产生欠拟合，模型泛化能力稳定。

表1 不同深度限制下决策树的性能评价指标

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 最大深度 | 准确率 | 精确率 | 召回率 | F1 分数 |
| 3 | 1.0000 | 1.0000 | 1.0000 | 1.0000 |
| 5 | 1.0000 | 1.0000 | 1.0000 | 1.0000 |
| 10 | 1.0000 | 1.0000 | 1.0000 | 1.0000 |

1. **总结**：讨论决策树的优缺点，以及避免过拟合的方法，实验收获等

1、决策树的优缺点

（1）优点：

* 模型直观易懂，生成的树结构可清晰展示决策逻辑，便于解释和分析。
* 无需对数据做复杂的预处理（如归一化），能处理离散和连续特征，适用场景广。
* 训练和预测过程相对高效，对于小规模数据可快速构建和使用。

（2）缺点：

* 容易过拟合，尤其是深度不加限制时，模型会学习到训练集噪声，泛化能力下降。
* 对数据分布变化敏感，样本轻微变动可能导致树结构大幅改变，稳定性欠佳 。

2、避免过拟合的方法

（1）预剪枝：构建树时设置限制条件，如最大深度（max\_depth），达到设定深度就停止分裂；设置最小样本划分数（min\_samples\_split），样本数不足时不再分裂，提前终止树生长，简化模型。

（2）后剪枝：先构建完整决策树，再从叶节点向上回溯，判断剪掉某子树后模型泛化性能是否提升，若提升则剪枝，使模型更简洁、泛化能力更强。

（本次实验主要用预剪枝）

3、实验收获

通过本次实验，深入理解了决策树算法原理，掌握信息增益计算、树构建及预剪枝实现。实际编码实现过程中，解决了特征处理（离散与连续）、递归构建逻辑、模型评估等问题，认识到参数（如深度）对模型性能的关键影响，学会用评估指标分析过拟合/欠拟合，为后续机器学习算法学习和应用打下坚实基础，也明白算法实践中需平衡模型复杂度与泛化能力，合理选择和调整参数。