实验三　决策树分类器的构建及应用实验

实验目标：理解决策树分类器的原理；

能用 python 构造一个决策树；

准确评估分类器精度。

实验工具：Python(推荐) 或 C/C++

实验步骤：

一、决策树算法原理:

决策树是属于机器学习监督学习分类算法中比较简单的一种，决策树是一个预测模型，代表对象属性与对象值之间的一种映射关系。树中每个节点表示某个对象，每个分叉路径代表某个可能的属性值，叶结点对应从根节点到该叶节点所经历的路径所表示的对象的值。决策树仅有单一输出，若欲有复数输出，可以建立独立的决策树以处理不同输出。

(一) ID3 算法

1. ID3 算法的概述

ID3 算法以信息论为基础，以信息熵和信息增益为衡量标准实现对数据的归纳分类。在 ID3 算法中，每次划分选取信息增益最高的属性为划分标准，重复这个过程，直至生成一个能完美分类训练样例的决策树。

信息熵（Entropy）：

$$H(x) = -\sum_{i=1}^{n} p(x_i) log_2 p(x_i) = \sum_{i=1}^{n} p(x_i) log_2 \frac{1}{p(x_i)}$$

,其中$p(x_i)$是选择i的概率。

熵越高，表示混合的数据越多。

信息增益（Information Gain）：

$$IG = H - \sum_{t \in T} p(t) H(t)$$

T是划分之后的分支集合，p(t)是该分支集合在原本的父集合中出现的概率，H(t)是该子集合的信息熵。

2. ID3 算法的流程

（1）数据准备：需要对数值型数据进行**离散化**
（2）ID3算法构建决策树：

- 如果数据集类别完全相同，则停止划分
- 否则，继续划分决策树：
  - 计算信息熵和信息增益来选择最好的数据集划分方法；
  - 划分数据集
  - 创建分支节点：
  - 对每个分支进行判定是否类别相同，如果相同停止划分，不同按照上述方法进行划分。

3. Python 实现

（以下面测试数据为例）

序号	不浮出水面是否可以生存	是否有脚蹼	是否属于鱼类
1	是	是	是
2	是	是	是
3	是	否	否
4	否	是	否
5	否	是	否

实现过程：

a. 构造函数 createDataSet
b. 计算信息熵
c. 利用构造的数据测试 calcShannonEnt
d. 按照最大信息增益划分数据集
e. 创建决策树构造函数 createTree
f. 将决策树运用于分类
g. 使用 Matplotlib 绘制决策树

详细代码参见：

https://blog.csdn.net/moxigandashu/article/details/71305273?locationNum=9&fps=1

（二）C4.5 算法

C4.5 在 ID3 的基础上改进，引入了新概念“信息增益率”,C4.5 是选择信息增益率最大的属性作为

树节点。

信息增益:

$$\text{Gain}_A(D) = H(D) - H(D \mid A)$$

信息增益率:

$$GainRatio_A(D) = Gain_A(D) / SplitInfo_A(D)$$

$$SplitInfo_A(D) = -\sum_{j=1}^{v} \frac{|D_j|}{|D|} * \log_2(\frac{D_j}{|D|})$$

以下面训练集为例：

训练集：

outlook	temperature	humidity	windy	
sunny	hot	high	false	N
sunny	hot	high	true	N
overcast	hot	high	false	Y
rain	mild	high	false	Y
rain	cool	normal	false	Y
rain	cool	normal	true	N
overcast	cool	normal	true	Y

$$SplitInfo_{A="outlook"}(D) = -\frac{num_{sunny}}{num} * \log_2 \frac{num_{sunny}}{num} - \frac{num_{rain}}{num} * \log_2 \frac{num_{rain}}{num} - \frac{num_{overcast}}{num} * \log_2 \frac{num_{overcast}}{num} = -\frac{2}{7} * \log_2 \frac{2}{7} - \frac{3}{7} * \log_2 \frac{3}{7} - \frac{2}{7} * \log_2 \frac{2}{7}$$

完整代码参考：

https://blog.csdn.net/chenxiqilin/article/details/50395809

实验要求：

1. 参考以上两段代码，以 sklearn 中的 Iris 数据集作为训练集，构建两类决策树并用于对鸢尾花的分类。
2. 写出两种算法的流程图或者伪代码，并分析它们的区别。

注意：请将实验报告以文件形式提交到 QQ 群里，统一命名为：学号+姓名+实验几.doc/rar/zip/pdf