决策树的介绍

决策树是一种常见的分类模型。

决策树的主要优点：

1. 具有很好的解释性，模型可以生成可以理解的规则。
2. 可以发现特征的重要程度。
3. 模型的计算复杂度较低。

决策树的主要缺点：

1. 模型容易过拟合，需要采用减枝技术处理。
2. 不能很好利用连续型特征。
3. 预测能力有限，无法达到其他强监督模型效果。
4. 方差较高，数据分布的轻微改变很容易造成树结构完全不同。

在做决策树的时候，会经历两个阶段：**构造和剪枝**。

- 构造就是生成一棵完整的决策树。简单来说，**构造的过程就是选择什么属性作为节点的过程**。构造过程中存在 3 种节点：根节点、内部节点和叶节点
- 剪枝就是给决策树瘦身，这一步想实现的目标就是，不需要太多的判断，同样可以得到不错的结果。之所以这么做，是为了防止“过拟合”（Overfitting）现象的发生。

衡量决策树的指标：**纯度**和**信息熵**

纯度

让目标变量的分歧最小。经典的 “不纯度”的指标有三种：信息增益（ID3 算法）、信息增益率（C4.5 算法）以及基尼指数（Cart 算法）

1. ID3 算法

   ID3 算法计算的是**信息增益**，信息增益指的就是划分可以带来纯度的提高，信息熵的下降。的计算公式，是父亲节点的信息熵减去所有子节点的信息熵：

$$Gain(D,a) = Entropy(D) - \sum_{i=1}^{k} \frac{|D_i|}{|D|} Entropy(D_i)$$

   公式中 D 是父亲节点，Di 是子节点，Gain(D,a) 中的 a 作为 D 节点的属性选择。

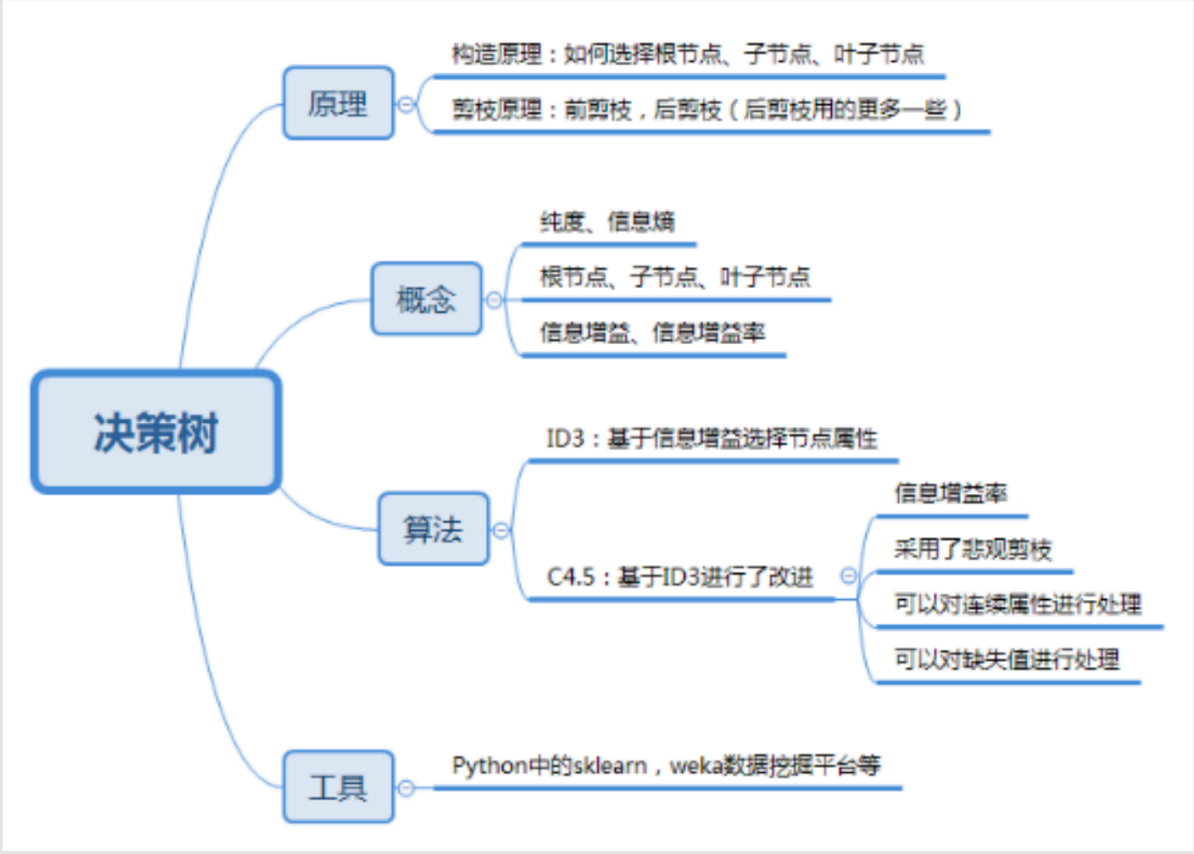
2. C4.5 算法

   C4.5 在 ID3 算法的基础上有以下改进：

   - ID3 在计算的时候，倾向于选择取值多的属性。为了避免这个问题，C4.5 采用信息增益率的方式来选择属性。信息增益率 = 信息增益 / 属性熵
   - ID3 构造决策树的时候，容易产生过拟合的情况。在 C4.5 中，会在决策树构造之后采用悲观

剪枝（PEP），这样可以提升决策树的泛化能力

- C4.5 可以处理连续属性的情况，对连续的属性进行离散化的处理
- 针对数据集不完整的情况，C4.5 也可以进行处理



3. Cart 算法

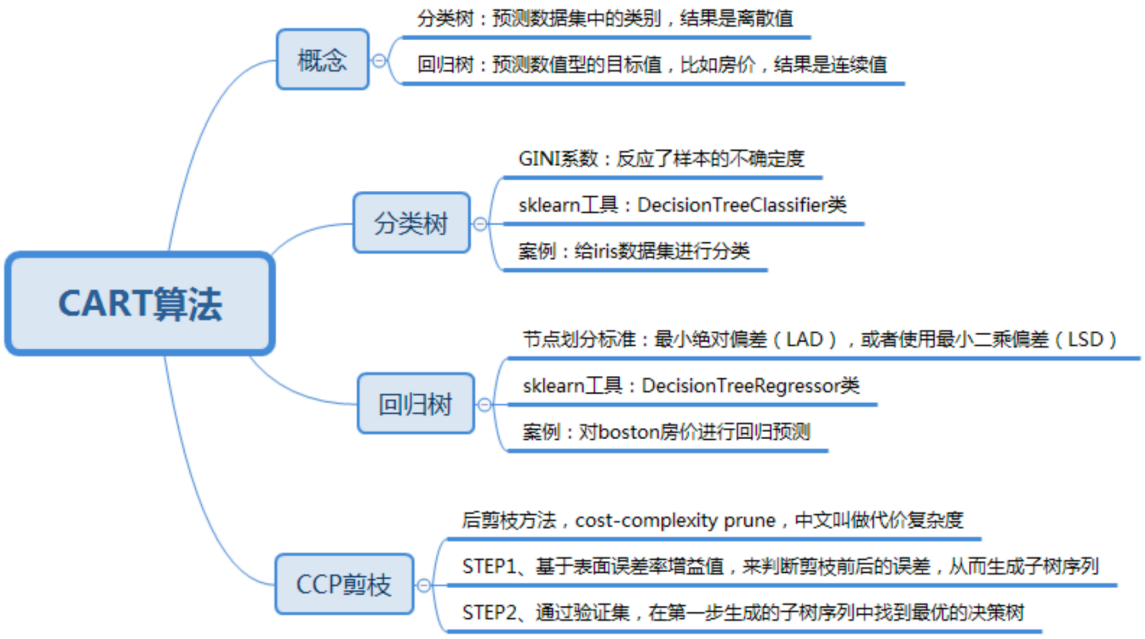
   CART 算法，英文全称叫做 Classification And Regression Tree，中文叫做分类回归树。ID3 和 C4.5 算法可以生成二叉树或多叉树，而 CART 只支持二叉树。同时 CART 决策树比较特殊，既可以作分类树，又可以作回归树。

   CART 分类树与 C4.5 算法类似，只是属性选择的指标采用的是基尼系数：

$$GINI(t) = 1 - \sum_{k} [p(C_k \mid t)]^2$$

这里 p(Ck|t) 表示节点 t 属于类别 Ck 的概率，节点 t 的基尼系数为 1 减去各类别 Ck 概率平方和。

基尼系数本身反应了样本的不确定度。当基尼系数越小的时候，说明样本之间的差异性小，不确定程度低。分类的过程本身是一个不确定度降低的过程，即纯度的提升过程。所以 CART 算法在构造分类树的时候，会选择基尼系数最小的属性作为属性的划分。



信息熵（entropy）

表示了信息的不确定度，公式：

$$Entropy(t) = -\sum_{i=0}^{c-1} p(i|t)\log_2 p(i|t)$$

p(i|t) 代表了节点 t 为分类 i 的概率，其中 log2 为取以 2 为底的对数。不确定性越大时，所包含的信息量也就越大，信息熵也就越高。

决策树构建的伪代码

输入： 训练集D={(x1x1,y1y1),(x2x2,y2y2),....,(xmxm,ymym)};
特征集A={a1a1,a2a2,....,adad}

输出： 以node为根节点的一颗决策树

过程：函数TreeGenerate(DD,AA)

1. 生成节点node
2. ifif DD中样本全书属于同一类别CC $then$then:
3. ----将node标记为CC类叶节点；$return$return
4. ifif AA = 空集 OR D中样本在AA上的取值相同 $then$then:
5. ----将node标记为叶节点，其类别标记为DD中样本数最多的类；$return$return

6. 从 AA 中选择最优划分属性 $a*$a*;
7. $for$for $a*$a* 的每一个值 $av*$a*v dodo:
8. ----为node生成一个分支，令DvDv表示DD中在$a*$a*上取值为$av*$a*v的样本子集；
9. ----ifif DvDv 为空 $then$then:
10. --------将分支节点标记为叶节点，其类别标记为DD中样本最多的类;$then$then
11. ----$else$else:
12. --------以 TreeGenerate(DvDv,AA{$a*$a*})为分支节点

决策树的构建过程是一个递归过程。函数存在三种返回状态：（1）当前节点包含的样本全部属于同一类别，无需继续划分；（2）当前属性集为空或者所有样本在某个属性上的取值相同，无法继续划分；（3）当前节点包含的样本集合为空，无法划分。