决策树

- 决策树是一种自上而下的，对样本数据进行树类的过程，由节点和有向边组成。
- 决策树的优点：可解释性好。
- 决策树的结点分为根节点、内部结点和叶结点。
- 决策树做为最基础、最常见的监督学习模型，常被用于分类问题和回归问题。
- 将决策树应用集成学习的思想可以得到随机森林等模型。
- 决策树分为特征选择，树的构造和树的剪枝三个过程。
- 常用的决策树算法有 ID3、C4.5、CART。
- 等概率熵的量化公式：

$$H = -\log_2 p$$

p 表示概率。

- 熵的一般量化公式：

$$Ent(D) = -\sum_{k=1}^{K} p_k \log_2 p_k$$

p_k 表示在样本集合D中第k类样本所占的比例，其中$k = 1,2,\cdots K$。

- ID3 算法树的构建准则：最大信息增益(辅助理解例子请参考课件决策树 Slides)

a. 经验熵

样本集合D，类别数K，数据集D的经验熵为：

$$H(D) = -\sum_{k=1}^{K} \frac{|C_k|}{|D|} \log_2 \frac{|C_k|}{|D|}$$

C_k是样本集合D中属于第k类的样本子集，$|C_k|$表示该子集的元素个数。$|D|$ 表示样本集合的元素个数。

b. 条件熵

条件熵$H(Y|X)$表示在已知随机变量X的条件下随机变量Y的不确定性，也即是已知随机变量X已经发生，随机变量Y的熵。

$$H(D|A)=\sum_{i=1}^{n}\frac{|D_i|}{|D|}H(D_i)=\sum_{i=1}^{n}\frac{|D_i|}{|D|}\left(-\sum_{k=1}^{K}\frac{|D_{ik}|}{|D_i|}\log_2\frac{|D_{ik}|}{|D_i|}\right)$$

D_i表示数据集D中特征A取第i个集的样本子集，D_{ik}表示D_i中属于第k类的样本子集。

c. 信息增益

信息增益又叫互信息，特征A对训练数据集D的信息增益写作$g(D,A)$，定义为集合D的经验熵$H(D)$与特征A给定条件下D的经验条件熵$H(D|A)$之差。

$$g(D,A)=H(D)-H(D|A)$$

- C4.5 算法树的构建准则：最大信息增益比

  特征A对训练数据集D的信息增益比写作$g_k(D,A)$，计算公式：

$$g_k(D,A)=\frac{g(D,A)}{H_A(D)}$$

  其中$H_A(D)$叫做数据集D关于特征的取值熵：

$$H_A(D)=-\sum_{i=1}^{n}\frac{|D_i|}{|D|}\log_2\frac{|D_i|}{|D|}$$

  n是特征A的取值个数。

- CART 算法树的构建准则：最大基尼指数$(Gini)$

  $Gini$描述了数据的纯度。

与 ID3 和 C4.5 算法不同，CART 选择基尼指数最小的特征及其对应的切分点进行分类。基尼指数的计算公式为：

$$Gini(D)=1-\sum_{k=1}^{n}\left(\frac{|C_k|}{|D|}\right)^2$$

- ID3、C4.5 和 CART 的比较

决策树算法	ID3	C4.5	CART
评价标准	信息增益	信息增益比	基尼指数
倾向性(缺陷)	倾向取值较多的特征	-	-
样本类型	离散型变量	连续性变量	连续性变量
应用角度	分类	分类	分类和回归
特征复用	No	No	Yes
准确性及泛化能力	剪枝	剪枝	所有可能的树的对比

- 决策树对数据的每个特征不断的分裂，所以特别不稳定，当数据产生噪音后，树构建后的样子可能会改变。解决的方法是使用集成学习的随机森林算法。
- 当数据特别复杂的时候，可以生成一个特别复杂的树，这个树会把数据中的各种情况都罗列出来，会生成大量的结点，导致过拟合现象。解决的办法是需要对决策树进行剪枝，剪掉一些枝叶。
- 决策树的剪枝通常有两种办法：预剪枝和后剪枝。
- 预剪枝：在生成决策树的过程中提前停止树的增长。
- 后剪枝：在已经生成的决策树上进行剪枝。
- 预剪枝的核心思想：在树的结点进行扩展之前，先计算当前的划分是否能带来模型泛化能力的提升，如果不能，则不再继续生长树。
- 后剪枝的核心思想：让算法生成一棵完全生长的决策树，然后从最底层向上计算是否剪枝。剪枝过程将子树删除，用一个叶子结点替代，该结点的类别同样按照多数投票的原则进行判断。如果剪枝过后准确率有所提升，则进行剪枝。
- 预剪枝虽然思想简单，但是需要一定的经验判断，所以容易造成欠拟合。相比于预剪枝，后剪枝方法通常可以得到泛化能力更强的决策树，但时间开销会更大。

- 决策树处理连续值的基本思想就是把连续的属性离散化。课件中使用的是二分法，就是把连续值处理成两个类。
- 决策树的缺失值处理：
    a. 如果数据集较大，可以直接删除带有缺失值的数据。
    b. 如果数据集较小，可以计算去掉缺失值的经验熵，并且求出去掉缺失值的信息增益。然后根据特征非缺失值的比例求出真正的信息增益。
- 多变量决策树：简讲。