《统计学习方法》第五章读书笔记

一、对自己提出问题的理解：

1. **为什么以信息增益作为划分训练数据集的特征，会偏向于选择取值较多的特征？能从公式上给出数学证明吗？**

**讨论后的理解**：特征对训练数据集的信息增益定义为集合的经验熵与特征给定条件下的经验条件熵之差，即

从公式出发，信息增益越大即经验条件熵越小，什么样的属性会有极小的经验条件熵呢？举个极端的例子，如果将身份证号作为一个属性，那么每个人的身份证号都是不相同的，也就是说，有多少个人，就有多少种取值，如果用身份证号这个属性去划分原数据集，那么，原数据集中有多少个样本，就会被划分为多少个子集，这样的话，会导致信息增益公式的第二项整体为0，虽然这种划分毫无意义，但是从信息增益准则来讲，这就是最好的划分属性。从概念来讲，信息增益表示由于特征而使得数据集的分类不确定性减少的程度，信息增益大的特征具有更强的分类能力。

1. **ID3、C4.5和CART三种决策树的区别？**

**讨论后的理解**：ID3算法采用信息增益进行评估和特征的选择，每次选择信息增益最大的特征作为判断模块。C4.5是ID3的一个改进算法，继承了ID3算法的优点，采用信息增益率来选择划分属性。CART算法可用于分类和回归问题，采用基尼系数或均方差进行特征选择。进一步对比如下表所示：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 算法 | 支持模型 | 树结构 | 特征选择 | 连续值处理 | 缺失值处理 | 剪枝 |
| ID3 | 分类 | 多叉树 | 信息增益 | 不支持 | 不支持 | 预剪枝 |
| C4.5 | 分类 | 多叉树 | 信息增益比 | 支持 | 支持 | 预剪枝 |
| CART | 分类、  回归 | 二叉树 | 基尼系数、均方差 | 支持 | 支持 | 后剪枝 |

二、对别人提出问题的理解：

1. **分类树和回归树的不同之处？**

**个人的理解：**分类树是根据数据的相似性来进行数据的分类，主要用于将数据集分类到响应变量所对应的不同类别里；回归树主要用于响应变量是数值的或者连续的，例如预测商品的价格，其适用于预测一些非分类的问题。

1. **cart剪枝中，“以t为单节点树”得到的损失函数是什么含义？**

**个人的理解：**从整体树T0开始剪枝，每次剪枝剪的都是某个内部节点的子节点，也就是将某个内部节点的所有子节点回退到这个内部节点里，并将这个内部节点作为叶子节点。因此在计算整体的损失函数时，这个内部节点以外的值都没变，只有这个内部节点的局部损失函数改变了，因此，本需要计算全局的损失函数，但现在只需要计算内部节点剪枝前和剪枝后的损失函数。

对任意内部节点t，剪枝前的状态：有|Tt|个叶子节点，预测误差是C(Tt)；剪枝后的状态：只有本身一个叶子节点，预测误差是C(t)。

三、读书计划

1. 本周完成的内容章节：第五章
2. 下周计划：第六章

四、读书摘要及理解或伪代码的具体实现

1. 读书摘要及理解

**树的剪枝：**包括预剪枝和后剪枝，通过提前停止树的构造进行剪枝的方法称为预剪枝，后剪枝是首先构造完整的决策树，然后把置信度不够节点子树替代为叶子节点的过程。相对而言预剪枝比较简单，在实际的运用中运用最广的还是后剪枝。

预剪枝判断停止树的生长可以归纳为以下几种：

1. 树的高度限制：设定树的高度最大值，当达到限定值时，停止树的生长；
2. 训练样本限制：对一个拥有较少训练样本的节点进行分裂时容易出现过拟合现象，因此设定样本量阀值，当样本量少于阀值时停止生长；
3. 系统性能增益：当属性的信息增益小于某个指定的阀值时停止增长。

后剪枝算法主要有以下几类：

1. 降低错误剪枝REP(Reduced Error Pruning)；
2. 悲观错误剪枝PER(Pessimistic Error Pruning)；
3. 基于错误剪枝EBP(Error-Based Pruning)；
4. 代价-复杂度剪枝CCP(Cost-Complexity Pruning)；
5. 最小错误剪枝MEP(Minimun Error Pruning)

**决策树的优缺点**

优点：

1. 决策树易于理解和解释，可以可视化分析，容易提取出规则；
2. 可以处理标称型和数值型数据；
3. 比较适合处理有缺失属性的样本；
4. 能够处理不相关的特征；
5. 应用范围广，可用于分类和回归，而且非常容易做多类别的分类；
6. 测试数据集时，运行速度比较快，在相对短的时间内能够对大型数据源做出可行且效果良好的结果。

缺点：

1. 很容易在训练数据中生成复杂的树结构，造成过拟合（overfitting）。剪枝可以缓解过拟合的负作用，常用方法是限制树的高度、叶子节点中的最少样本数量。
2. 学习一棵最优的决策树被认为是NP-Complete问题。实际中的决策树是基于启发式的贪心算法建立的，这种算法不能保证建立全局最优的决策树。Random Forest（随机森林）的引入能缓解这个问题
3. 代码的具体实现

调用Weka api验证了C4.5决策树分类算法，数据集为鸢尾花iris.arff，测试得到分类的正确率0.98。