决策树模型

目录 CONTENTS

01

决策树模型与学习

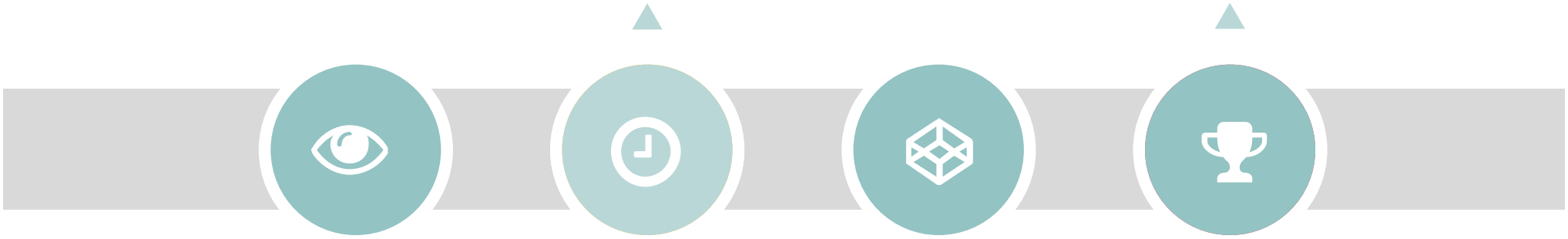
4 模 型 机 制

2. 训练模型

通过收集到的数据，使用某种算法进行模型的学习

4. 预测新数据

给一个新的输入数据，训练好的模型，计算出预测的结果



1. 数据集与特征

收集模型学习需要的数据；针对业务场景构建关键的特征

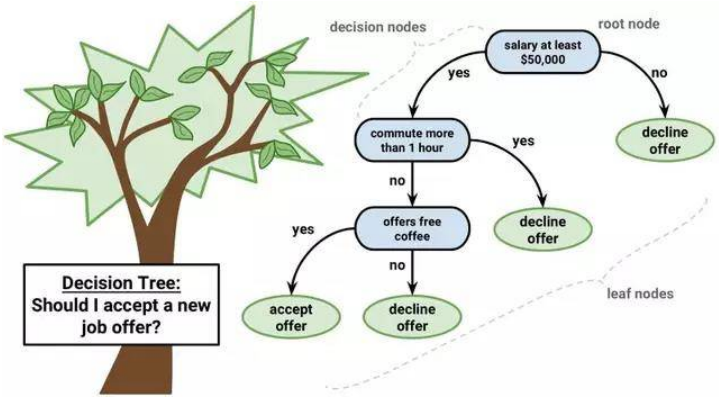
3. 模型评估

对模型的效果进行评估。最恰当的方式是在测试集上评估

5 决　策　树

- 决策树可以用来进行分类和回归建模。
  - ① **分类模型：目标变量是离散的**
  - ② 回归建模：目标变量时连续的

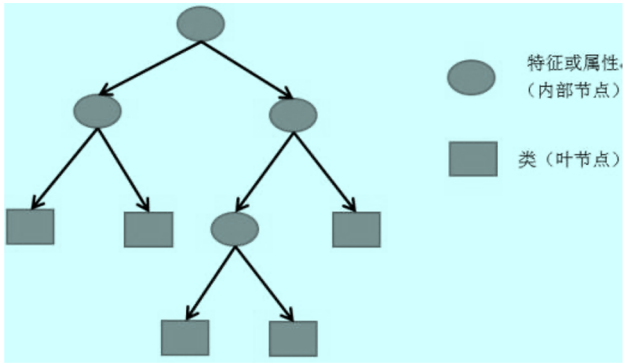
- 这一节的知识点涉及到的数学公式很多，较为枯燥，但是很重要！



6 决策树模型

决策树定义 分类决策树模型是一种描述对实例进行分类的树形结构。决策树由结点(node) 和有向边组成。结点有两种类型：内部结点(internal node)和叶结点(leaf node)。内部结点表示一个特征或属性，叶结点表示一个类。

用决策树分类，从根结点开始，对实例的某一特征进行测试，根据测试结果，将实例分配到其子节点；这时，每一个子节点对应着该特征的一个取值。如此递归地对实例进行测试并分配，直至达到叶结点。最后将实例分到叶结点的类中。



7 决策树学习

给定训练数据集

$$D = \{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)\}$$

其中$x_i = (x_i^{(1)}, x_i^{(2)}, \dots, x_i^{(n)})^T$为输入实例(样本)，$n$为特征个数，$y_i \in \{1,2, \dots, K\}$为类标记，$i = 1,2, \dots, N$，$N$为样本容量。

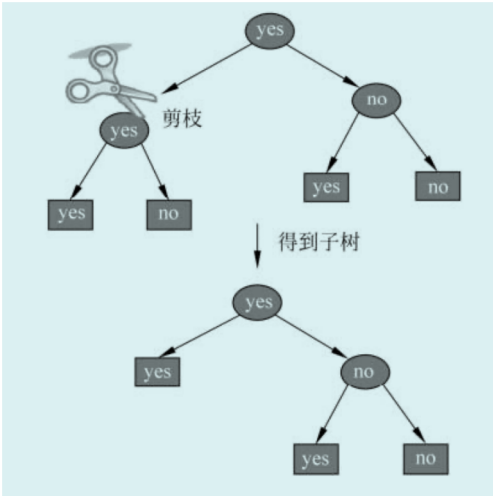
- 决策树学习的目标是根据给定的训练数据集构建一个决策树模型，使它能够对实例进行正确的分类。
- 从训练数据集中可能得到多个能对数据集正确分类的决策树，也可能一个都没有。我们需要的是一个与训练数据集矛盾（误差）较小的决策树，同时具有很好的泛化能力。
- 决策树学习是由训练数据集估计条件概率模型$P(Y|X)$，基于特征空间划分的类的条件概率模型有无穷多个。我们选择的条件概率模型应该不仅对训练数据有很好的拟合，而且对未知数据有很好的预测。

8 决策树学习

- 决策树学习的算法通常是一个递归地选择最优特征，并根据该特征对训练数据进行分割，使得对各个子数据集有一个最好的分类的过程。这一过程对应着对特征空间的划分，也对应着决策树的构建。
- 开始，构建根结点，将所有训练数据都放在根结点。选择一个最优特征，按照这一特征将训练数据集分割成子集，使得各个子集有一个在当前条件下最好的分类。如果这些子集已经能够被基本正确分类，那么构建叶结点，并将这些子集分到所对应的叶结点中去；如果还有子集不能被基本正确分类，那么就对这些子集选择新的最优特征，继续对其进行分割，构建相应的结点。
- 如此递归地进行下去，直至所有训练数据子集被基本正确分类，或者没有合适的特征为止。最后每个子集都被分到叶结点上，即都有了明确类，这就生成了一颗决策树。

9 决策树学习-剪枝

- 决策树可能对训练数据有很好的分类能力，但对未知的测试数据却未必有很好的分类能力，即可能发生过拟合现象。我们需要对已生成的树自下而上进行剪枝，将树变得更简单，从而使它具有更好的泛化能力。具体地，就是去掉过于细分的叶结点，使其回退到父结点，甚至更高的结点，然后将父结点或更高的结点改为新的叶结点。
- 决策树的生成对应于模型的局部选择，决策树的剪枝对应于模型的全局选择。决策树的生成只考虑局部最优，相对地，决策树的剪枝则考虑全局最优。



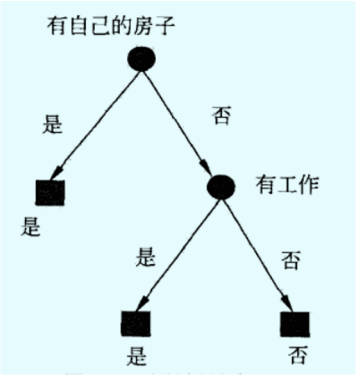
10 决策树学习

是否要贷款？

ID	年龄	有工作	有自己的房子	信贷情况	类别
0	青年	否	否	一般	否
1	青年	否	否	好	否
2	青年	是	否	好	是
3	青年	是	是	一般	是
4	青年	否	否	一般	否
5	中年	否	否	一般	否
6	中年	否	否	好	否
7	中年	是	是	好	是
8	中年	否	是	非常好	是
9	中年	否	是	非常好	是
10	老年	否	是	非常好	是
11	老年	否	是	好	是
12	老年	是	否	好	是
13	老年	是	否	非常好	是
14	老年	否	否	一般	否

特征选择 ➡ 树生成 ➡ 剪枝

如何从训练数据构建出一颗决策树？

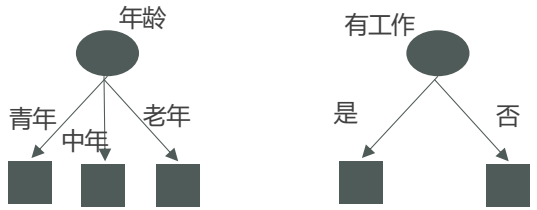


02

特　征　选　择

12 特 征 选 择

- 特征选择在于选取对训练数据具有分类能力的特征
- 案例中希望通过给定的训练数据学习一个贷款申请的决策树，用以对未来的贷款申请进行分类，即当新的客户提出贷款申请时，根据申请人的特征利用决策树决定是否批准贷款
- 如图所示，选择不同的特征会构建出不同的决策树。核心问题在于：究竟选取哪个特征更好呢



不同特征决定不同的决策树

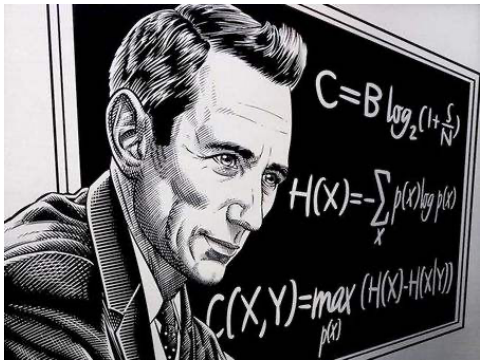
贷款申请样本数据表

ID	年龄	有工作	有自己的房子	信贷情况	类别
0	青年	否	否	一般	否
1	青年	否	否	好	否
2	青年	是	否	好	是
3	青年	是	是	一般	是
4	青年	否	否	一般	否
5	中年	否	否	一般	否
6	中年	否	否	好	否
7	中年	是	是	好	是
8	中年	否	是	非常好	是
9	中年	否	是	非常好	是
10	老年	否	是	非常好	是
11	老年	否	是	好	是
12	老年	是	否	好	是
13	老年	是	否	非常好	是
14	老年	否	否	一般	否

13 熵 的 定 义

在信息论与概率统计中，熵（entropy）是表示随机变量不确定性的度量。设X是一个取有限个值的离散随机变量，其概率分布为

$$P(X=x_i)=p_i,\quad i=1,2,\cdots,n$$



则随机变量X的**熵定义**为

$$H(X)=-\sum_{i=1}^{n}p_i\log p_i$$

熵越大，随机变量的不确定性就越大

$$0\leqslant H(p)\leqslant \log n$$

当随机变量只取两个值，例如1，0时，即 X 的分布为

$$P(X=1)=p,\quad P(X=0)=1-p,\quad 0\leqslant p\leqslant 1$$

熵为

$$H(p)=-p\log_2 p-(1-p)\log_2(1-p)$$

14 条件熵与经验熵

当$p=0$ 或 $p=1$时 $H(\mathrm{p})=0$，随机变量完全没有不确定性。当$p=0.5$时， $H(\mathrm{p})=1$，熵取值最大，随机变量不确定性最大。

设有随机变量（X,Y），其联合概率分布为

$$P(X=x_i,Y=y_j)=p_{ij},\quad i=1,2,\cdots,n;\quad j=1,2,\cdots,m$$

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。定义为X给定条件下Y的条件概率分布的熵对X的数学期望

$$H(Y|X)=\sum_{i=1}^{n}p_iH(Y|X=x_i)$$

15 信 息 增 益

信息增益（information gain）表示得知特征X的信息而使得类Y的信息的不确定性减少的程度。

信息增益定义 特征A对训练数据集D 的信息增益$g(D,A)$，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,即

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

- 经验熵$H(D)$表示对数据集D进行分类的不确定性
- 经验条件熵$H(D|A)$表示在特征A给定条件下对数据集D进行分类的不确定性
- 信息增益$g(D,A)$表示得知特征A的信息而使得类D的信息的不确定性减少的程度

信息增益特征选择方法： 对训练数据集D，计算其每个特征的信息增益，并且比较它们的大小，选择信息增益最大的特征

16 信息增益算法

算法：输入数据集D和特征A

（1）计算数据集D的经验熵

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

（2）计算特征A对数据集D的经验条件熵

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

（3）计算信息增益

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

设训练数据集为D，$|D|$表示其样本容量。设有K个类C_k，k = 1,2,3,...,K, $|C_k|$为属于类C_k 的样本个数，$\sum_{k=1}^{K}|C_k|=|D|$. 设特征A有n个不同的取值$\{a_1,a_2,\dots,a_n\}$,根据特征A的取值将D划分为n个子集$D_1,D_2,\dots,D_n$,$|D_i|$为D_i的样本个数，$\sum_{i=1}^{n}|D_i|=|D|$。记子集D_i中属于类C_k的样本集合为D_{ik}, 即D_{ik}=$D_i\cap C_k$，$|D_{ik}|$为D_{ik}的样本个数。

17 信息增益比

以信息增益作为划分训练数据集的特征，存在偏向于选择取值较多的特征的问题。使用信息增益比可以对这一问题进行校正。

信息增益比定义 特征A对训练数据集D 的信息增益比$g_R(D,A)$定义为其信息增益$g(D,A)$与训练数据集D关于特征A的值的熵$H_A(D)$之比

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

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

03

决 策 树 生 成

19 决策树算法

1. ID3: Quinlan 发明于1986年，该算法会计算分类变量能够针对分类目标产生的 information gain，然后挑选出最大的ig作为树的Node
2. C4.5：是ID3的继承者，摒弃了ID3只能计算分类特征的缺点，能够把连续特征进行分段转换成分类特征进行计算。C4.5使用信息增益比
3. C5.0：就是C4.5的一个改进
4. CART：是当前决策树最通用的算法，能够用于回归和分类

20 ID3算法

- 输入：训练数据集D，特征集A，阈值ϵ
- 输出：决策树T

1. 若D中所有的实例属于同一类C_{k}，则T为单节点树，并将类C_{k}作为该结点的类标记，返回T
2. 若A=ϕ，则T为单节点树，并将D中实例数最大的类C_{k}作为该结点的类标记，返回T
3. 否则，按照计算A中各特征对D的信息增益，选择信息增益最大的特征A_{g}
4. 如果A_{g}的信息增益小于阈值ϵ，则置T为单结点树，并将D中实例数最大的类C_{k}作为该结点的类标记，返回T
5. 否则，对A_{g}的每一可能值a_i，将D分割为若干非空子集D_i，将D_i中实例数最大的类作为标记，构建子结点，由结点及其子节点构成树T，返回T
6. 对第i个子节点，以D_i为训练集，以$A-\{A_{\mathrm{g}}\}$为特征集，递归地调用 1–5，得到子树T_i,返回T_i

21 ID3算法

- 输入：训练数据集D，特征集A，阈值ϵ
- 输出：决策树T

1. 若D中所有的实例属于同一类C_k，则T为单节点树，并将类C_k作为该结点的类标记，返回T

2. 若A=ϕ，则T为单节点树，并将D中实例数最大的类C_k作为该结点的类标记，返回T

22 ID3算法

- 输入：训练数据集D，特征集A，阈值ϵ
- 输出：决策树T

1. 若D中所有的实例属于同一类C_k，则T为单节点树，并将类C_k作为该结点的类标记，返回T
2. 若A=ϕ，则T为单节点树，并将D中实例数最大的类C_k作为该结点的类标记，返回T
3. 否则，按照计算A中各特征对D的信息增益，选择信息增益最大的特征A_g

23 ID3算法

4. 如果A_g的信息增益小于阈值ϵ，则置T为单结点树，并将D中实例数最大的类C_k作为该结点的类标记，返回T

5. 否则，对A_g的每一可能值a_i，将D分割为若干非空子集D_i，将D_i中实例数最大的类作为标记，构建子结点，由结点及其子节点构成树T，返回T

6. 对第i个子节点，以D_i为训练集，以$A-\{A_g\}$为特征集，递归地调用 1–5，得到子树T_i,返回T_i

26 ID3算法案例

贷款申请样本数据表

ID	年龄	有工作	有自己的房子	信贷情况	类别
0	青年	否	否	一般	否
1	青年	否	否	好	否
2	青年	是	否	好	是
3	青年	是	是	一般	是
4	青年	否	否	一般	否
5	中年	否	否	一般	否
6	中年	否	否	好	否
7	中年	是	是	好	是
8	中年	否	是	非常好	是
9	中年	否	是	非常好	是
10	老年	否	是	非常好	是
11	老年	否	是	好	是
12	老年	是	否	好	是
13	老年	是	否	非常好	是
14	老年	否	否	一般	否

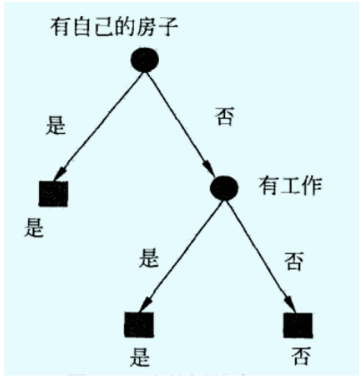
27 ID3算法案例

利用ID3算法构建决策树

由于特征A_3的信息增益值最大，所以选择特征A_3作为根节点的特征。它将训练数据集D划分为两个子集D_1（A_3取值为“是”）和D_2（A_3取值为“否”）。由于D_1只有同一类的样本点（y的值都为“是”），所以它成为一个叶结点，类标记为“是”。

对D_2继续从剩余三个特征中进行信息增益的计算，并且选取最大值特征

$$g(D_2, A_1) = H(D_2) - H(D_2|A_1) = 0.918 - 0.667 = 0.251$$
$$g(D_2, A_2) = H(D_2) - H(D_2|A_2) = 0.918$$
$$g(D_2, A_4) = H(D_2) - H(D_2|A_4) = 0.474$$



选择信息增益值最大的A_2作为结点的特征。A_2的取值有两个，所以继续将数据集D_2划分为两个子集（两个子结点）。一个子结点取值“是”，包含3个样本，属于同一类，所以这是叶结点，另一个子结点取值“否”，包含6个样本，属于同一类，所以这也是一个叶结点。因此，决策树构建完毕，所有样本已被完全分类！

28 C4.5算法

C4.5算法对ID3算法进行了改进

1. C4.5使用信息增益比作为特征选择的方法，可以克服信息增益作为标准容易偏向于取值较多的特征的问题
2. C4.5可以处理连续的输入特征:

   C4.5将连续特征离散化，比如特征A有m个样本，排序后a_1, a_2, a_3,… a_m，取相邻两个样本的均值，一共m−1个划分点，其中第i个划分点使用$T_i = \frac{a_i + a_{i+1}}{2}$表示。分别计算这m−1个点的信息增益，选择信息增益最大的点作为该连续特征的二元离散分类点

C4.5算法的缺点

- 生成的是多叉树
- 只能用于分类
- 使用了熵模型，涉及大量的对数运算，很耗时

29 C4.5算法

- 输入：训练数据集D，特征集A，阈值ϵ
- 输出：决策树T

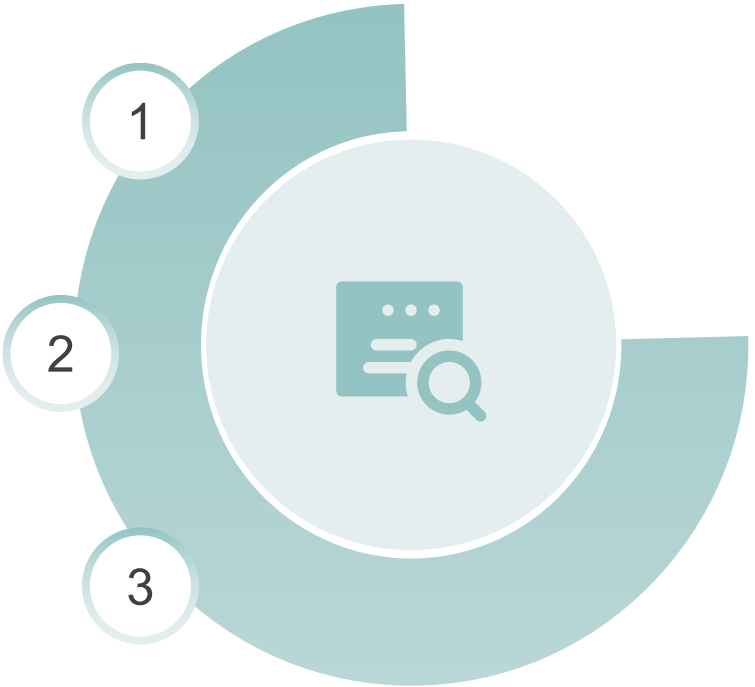
1. 若D中所有的实例属于同一类C_{k}，则T为单节点树，并将类C_{k}作为该结点的类标记，返回T
2. 若A=ϕ，则T为单节点树，并将D中实例数最大的类C_{k}作为该结点的类标记，返回T
3. 否则，按照计算A中各特征对D的信息增益比，选择信息增益比最大的特征A_{g}
4. 如果A_{g}的信息增益比小于阈值ϵ，则置T为单结点树，并将D中实例数最大的类C_{k}作为该结点的类标记，返回T
5. 否则，对A_{g}的每一可能值a_i，将D分割为若干非空子集D_i，将D_i中实例数最大的类作为标记，构建子结点，由结点及其子节点构成树T，返回T
6. 对第i个子节点，以D_i为训练集，以$A-\{A_{\mathrm{g}}\}$为特征集，递归地调用 1–5，得到子树T_i,返回T_i

04

CART分类回归树

31 CART算法

1. CART与C4.5非常类似，但是它支持分类与回归模型；即目标值可以是离散或者连续值
2. CART构建的是二叉树，每个结点只有两个分支
3. CART剪枝算法使用损失函数最小作为标准



32 基尼指数

- CART分类树算法使用基尼系数来代替信息增益比，基尼系数代表了模型的不纯度，基尼系数越小，则不纯度越低，特征越好。
- **基尼指数的定义**
- 在分类问题中，有K个类别，第k个类别的概率为p_{k}，则基尼系数的表达式为：
  - $Gini(p) = \sum_{k=1}^{K} p_{\mathrm{k}}(1 - p_{\mathrm{k}}) = 1 - \sum_{k=1}^{K} p_k^2$
- 对于二分类问题，若样本点属于第 1 类的概率是p，则概率分布的基尼指数为
  - $Gini(p) = 2p(1 - p)$
- 对于给定的样本D，假设有K个类别，第k个类别的数量为C_{k}，则样本D的基尼系数表达式为：
  - $Gini(D) = 1 - \sum_{k=1}^{K} (\frac{|C_{\mathrm{k}}|}{|D|})^2$

33 基尼指数

- 对于样本D，如果根据特征A的某个值a，把D分成D1和D2两部分，则在特征A的条件下，D的基尼系数表达式为：
  - $Gini(D,A)=\frac{D_1}{D}Gini(D_1)+\frac{D_2}{D}Gini(D_2)$
- 基尼指数 $Gini(D)$ 表示集合D的不确定性，基尼指数 $Gini(D,A)$ 表示A = a 分割后集合D的不确定性。基尼指数越大，样本集合的不确定也就越大。

34 CART算法

输入：训练数据集D，停止计算的条件；

输出：CART决策树

1. 设结点的训练数据集为D，计算现有特征对该数据及的基尼指数。此时，对每一个特征A，对其可能取的每个值a，根据样本点对A=a 的测试为“是”或“否”将D分割成D_1和D_2两部分，利用$Gini(D,A)$公式计算A=a时的基尼指数
2. 在所有可能的特征A以及它们所有可能的切分点a中，选择基尼指数最小的特征及其对应的切分点作为最优特征与最优切分点。依最优特征与最优切分点，从现结点生成两个子结点，将训练数据集依特征分配到两个子结点中去
3. 对两个子结点递归地调用1，2，直至满足停止条件。
4. 生成CART决策树

35 CART案例

针对之前的案例，应用CART算法生成决策树

- 计算特征的基尼指数，选取最优分裂特征及其最优切分点
- 求特征A_1的基尼指数

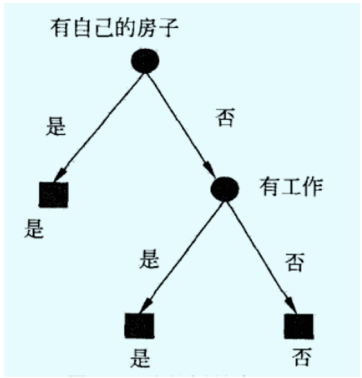
$$\text{Gini}(D, A_1 = 1) = \frac{5}{15}\left(2 \times \frac{2}{5} \times \left(1 - \frac{2}{5}\right)\right) + \frac{10}{15}\left(2 \times \frac{7}{10} \times \left(1 - \frac{7}{10}\right)\right) = 0.44$$
$$\text{Gini}(D, A_1 = 2) = 0.48$$
$$\text{Gini}(D, A_1 = 3) = 0.44$$

- 求特征A_2和A_3的基尼指数

$$\text{Gini}(D, A_2 = 1) = 0.32$$
$$\text{Gini}(D, A_3 = 1) = 0.27$$

- 求特征A_4的基尼指数

$$\text{Gini}(D, A_4 = 1) = 0.36$$
$$\text{Gini}(D, A_4 = 2) = 0.47$$
$$\text{Gini}(D, A_4 = 3) = 0.32$$



解释与结论

1. $A_1 = 1$和$A_1 = 3$都可以作为最优切分点
2. 由于A_2和A_3只有一个切分点，所以他们就是最优切分点
3. $A_4 = 3$是最优切分点

4. 在四个特征中，$Gain(D, A_3 = 1) = 0.27$最小，所以选择A_3为最优特征，$A_3 = 1$为其最优切分点。于是根节点生成两个子节点，一个是叶结点。对于另一个结点继续使用以上方法在A_1，A_2，A_4中选择最优特征及其最优切分点，结果是$A_2 = 1$。依此计算，所有结点都是叶结点

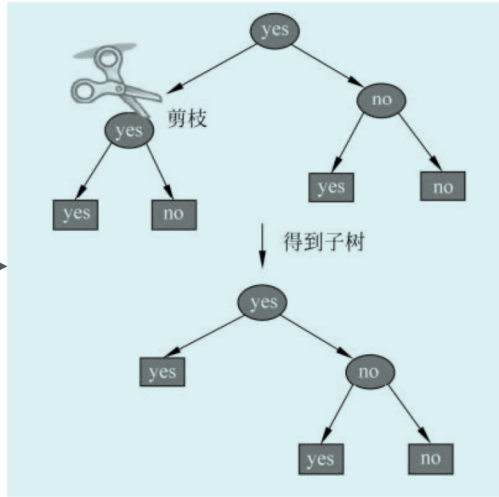
05

课　程　小　结

37 决策树学习

决策树学习过程

特征选择 → 树生成 → 剪枝



38 三 种 算 法

ID3

信息增益

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

C4.5

信息增益比

$$g_R(D,A) = \frac{g(D,A)}{H(D)}$$

CART

基尼指数

$$Gini(D,A) = \frac{D_1}{D}Gini(D_1) + \frac{D_2}{D}Gini(D_2)$$