
决策树

Outline

决策树简介

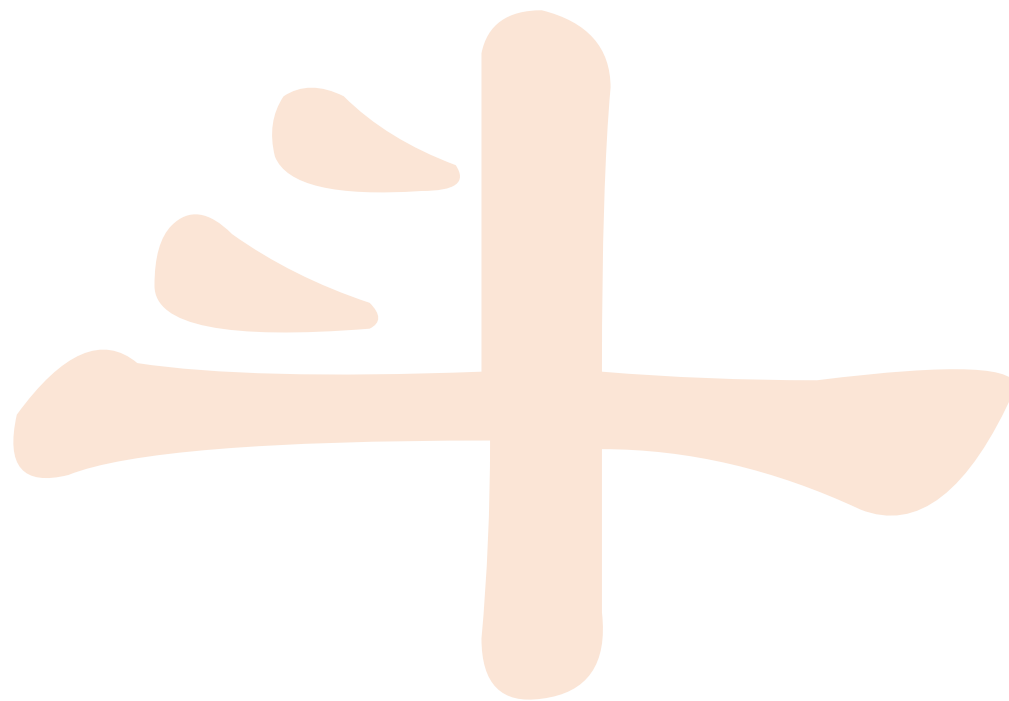
ID3算法

C4.5、CART算法

【实践】决策树

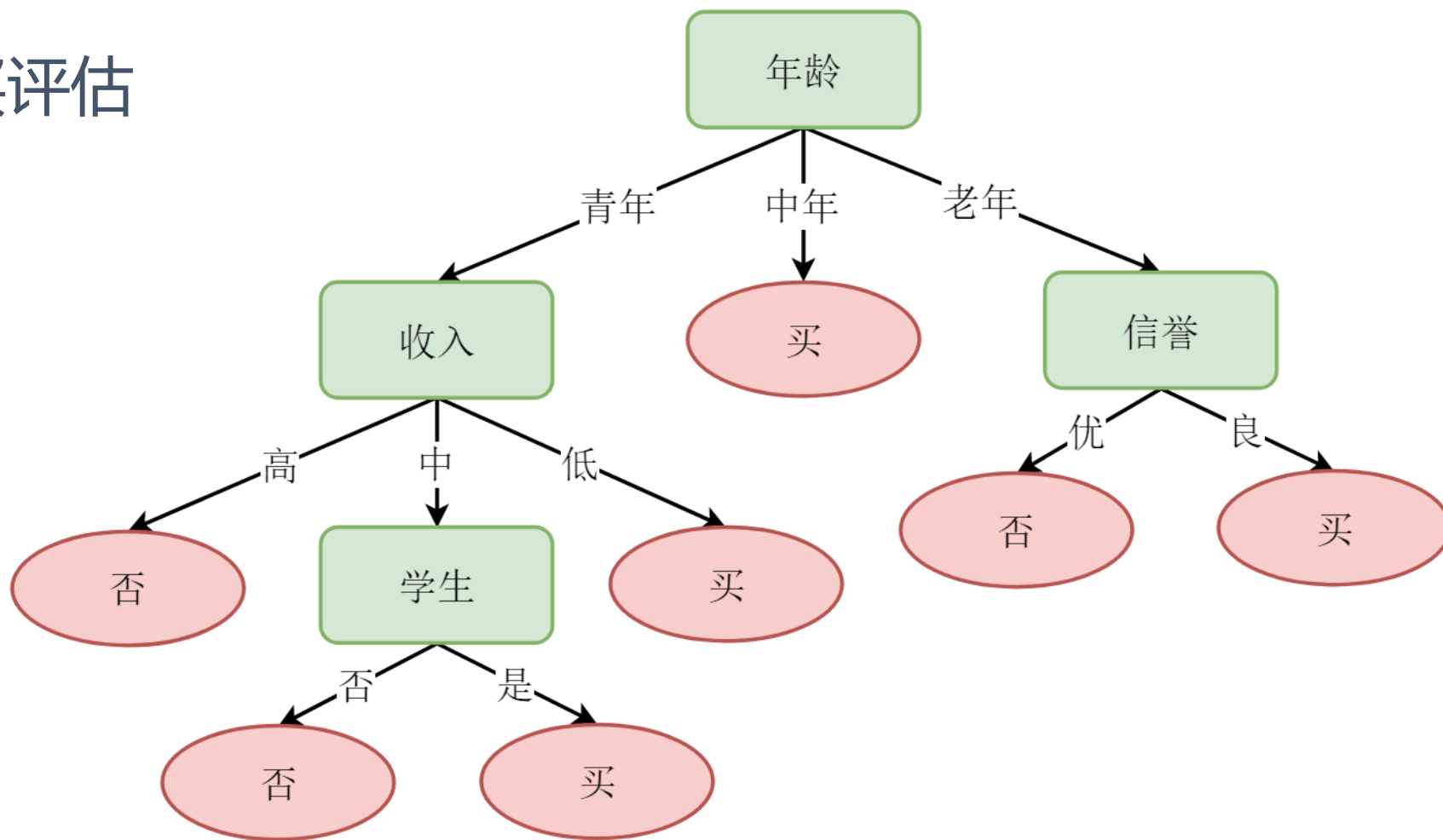
决策树

- 树形结构决策规则
- 分类问题，对样本：
 - 从根节点出发
 - 根据节点规则决定走哪个子节点
 - 一直走到叶子节点
 - 根据叶子节点的输出规则输出



决策树 - 例子

• 例子：购买评估



决策树 - 例子

• 如何做到？

计数	年龄	收入	学生	信誉	是否购买
64	青	高	否	良	否
64	青	高	否	优	否
128	中	高	否	良	买
60	老	中	否	良	买
64	老	低	是	良	买
64	老	低	是	优	否
64	中	低	是	优	买
128	青	中	否	良	否
64	青	低	是	良	买
132	老	中	是	良	买
64	青	中	是	优	买
32	中	中	否	优	买
32	中	高	是	良	买
64	老	中	否	优	否

决策树 - 例子

- 年龄作为根节点
 - 青年, {否, 买}
 - 中年, {买}
 - 老年, {否, 买}
- “中年” 停止分解, “青年”、 “老年” 继续分解

决策树 - 例子

• 对“青年”分解

计数	年龄	收入	学生	信誉	是否购买
64	青	高	否	良	否
64	青	高	否	优	否
128	青	中	否	良	否
64	青	低	是	良	买
64	青	中	是	优	买

- 经分析，高收入和低收入，只对应一个标签，停止分裂，直接将该标签作为叶子节点

决策树 - 例子

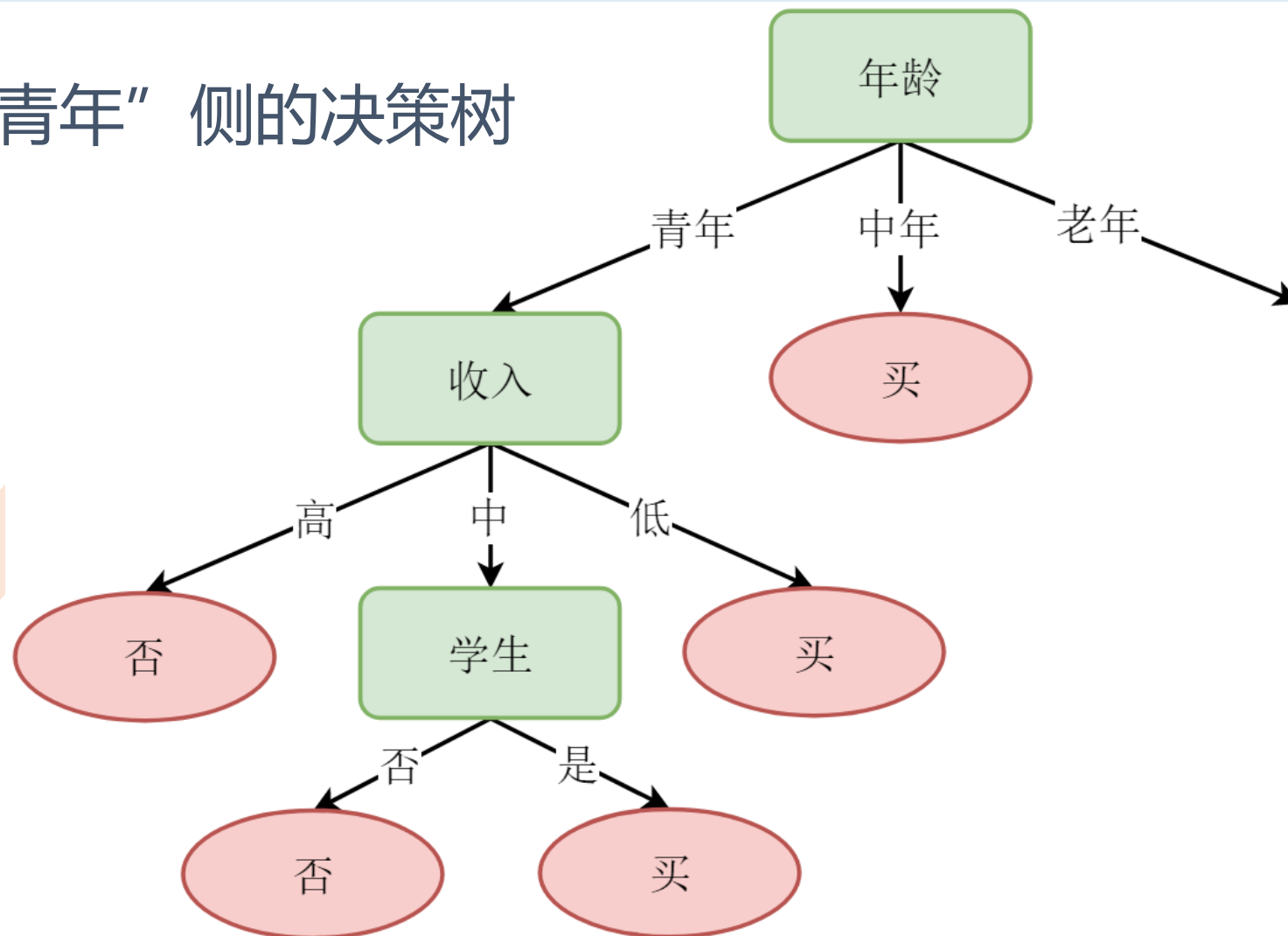
- 对“中收入人群”继续分解

计数	年龄	收入	学生	信誉	是否购买
128	青	中	否	良	否
64	青	中	是	优	买

- 经分析，学生特征可以完全区分标签，停止分裂

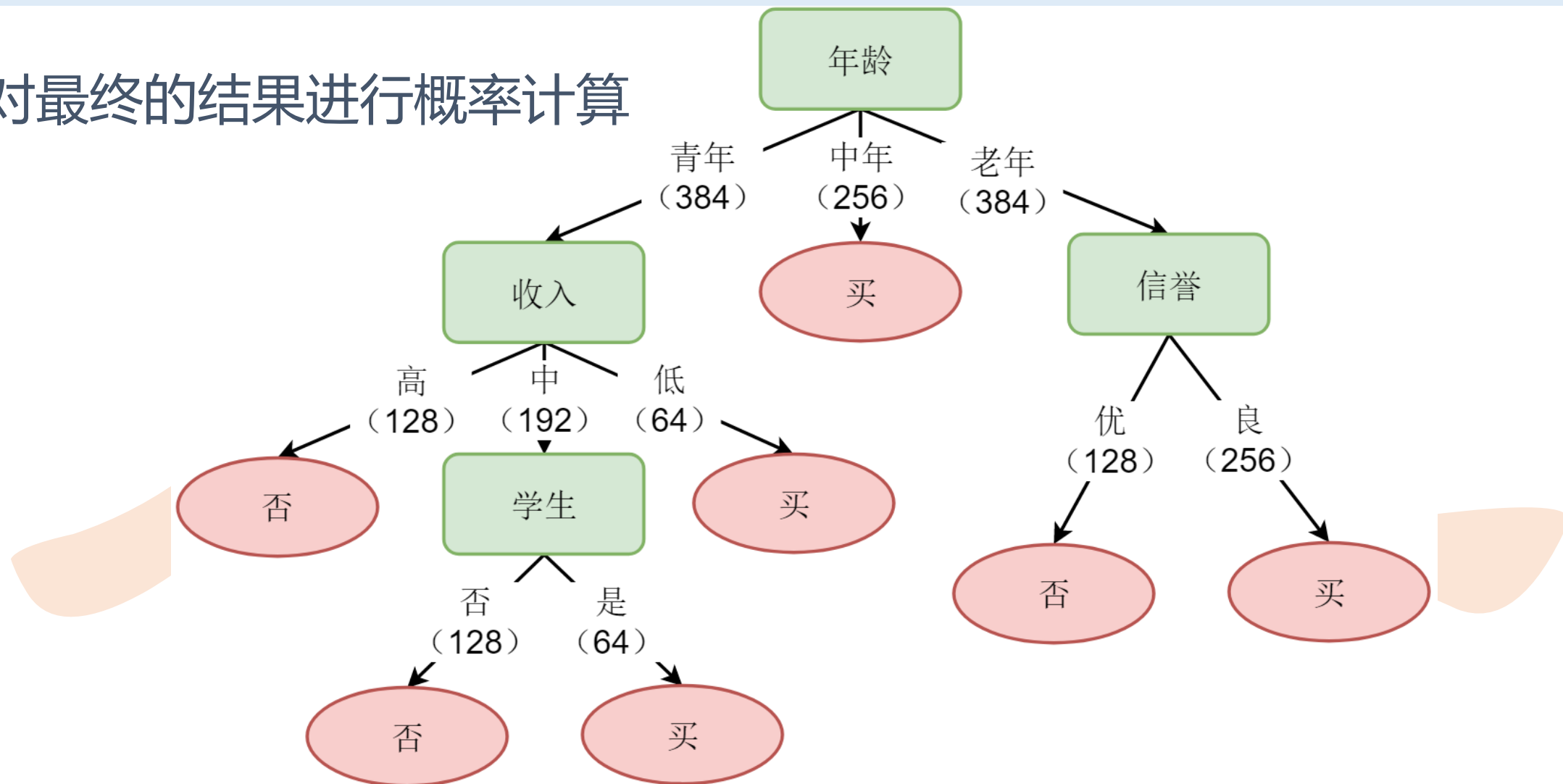
决策树 - 例子

- 于是生成“青年”侧的决策树



决策树 - 例子

- 对最终的结果进行概率计算



特征离散化

- 提前将各个特征值**离散化**

- 年龄：0（青年）,1（中年）,2（老年）
- 收入：0（高收入）, 1（中收入）, 2（低收入）
- 学生：0（是）, 1（否）
- 信誉：0（优）, 1（良）

信息熵，香农定理

- 不确定性函数 I 称为事件的信息量，事件 U 发生概率 p 的单调递减函数：

$$I(U) = \log\left(\frac{1}{p}\right) = -\log(p)$$

- 信息熵：一个信源中，不能仅考虑单一事件发生的不确定性，需要考虑所有可能情况的平均不确定性，为 $-\log(p)$ 的统计平均值 E

$$H(U) = E[-\log(p(u_i))] = - \sum_i p(u_i) \log(p(u_i))$$

- 信息熵是事物不确定性的度量标准
- 决策树中，不仅可用来度量类别不确定性，也可以度量包含不同特征的数据样本与类别的不确定性。
- 熵越大，不确定性越大，即混乱度越大

学习算法简介

• 学习算法

- 给定训练数据，决定树的结构

- 节点分裂规则
- 叶子结点输出规则

• 著名算法

- ID3

- C4.5

- 连续变量、缺省值、剪枝等

Outline

决策树简介

ID3算法

C4.5、CART算法

【实践】决策树

ID3 学习算法

• ID3

- 输入：离散值（属性）
- 使用信息增益来学习分裂规则

• 信息熵（Entropy）

- S是样例集合
- $p(u_i)$ 表示S中第i类样例的比例

$$H(S) = - \sum_i p(u_i) \log(p(u_i))$$

• 信息增益：

- 用规则r将样例集合S分为k个子集S1、.....、Sk
 - $|\cdot|$ 表示集合元素个数

ID3 学习算法

- 节点分裂规则：
 - 按属性：k种可能的属性值->k个子集
- 学习问题：挑那种属性？
 - ID3算法：信息增益最大的！

信息增益如何计算？

- 设S是s个数据样本的集合，假定类别标签具有m个不同值，定义m个不同类 $C_i (i = 1, 2, \dots, m)$ ，设 S_i 是 C_i 的样本数，对于一个给定的样本分类所需要的信息熵由下式给出：

$$I(s_1, s_2, \dots, s_m) = - \sum_i p_i \log(p_i)$$

- p_i 是任意样本属于 C_i 的概率，并用 $p_i = \frac{S_i}{|S|}$ 估计

信息增益如何计算？

- 用信息熵来度量每种特征不同取值的不确定性
- 设A具有v个不同的值 $\{a_1, a_2, \dots, a_v\}$
- 某一特征A将S划分为v个不同的子集 $\{S_1, S_2, \dots, S_v\}$,
- 其中 S_j 包含S中这样一些样本：他们在A上具有值 a_j ，若选A作测试特征，即最优划分特征，那么这些子集就是S节点中生长出来的决策树分支。设 s_{ij} 是子集 S_j 中类 C_i 的样本数。

信息增益如何计算？

- 由A划分成子集的熵或期望信息由下式给出：

$$E(A) = \sum_{j=1}^v \frac{s_{1j} + s_{2j} + \dots + s_{mj}}{s} I(s_{1j}, s_{2j}, \dots, s_{mj})$$

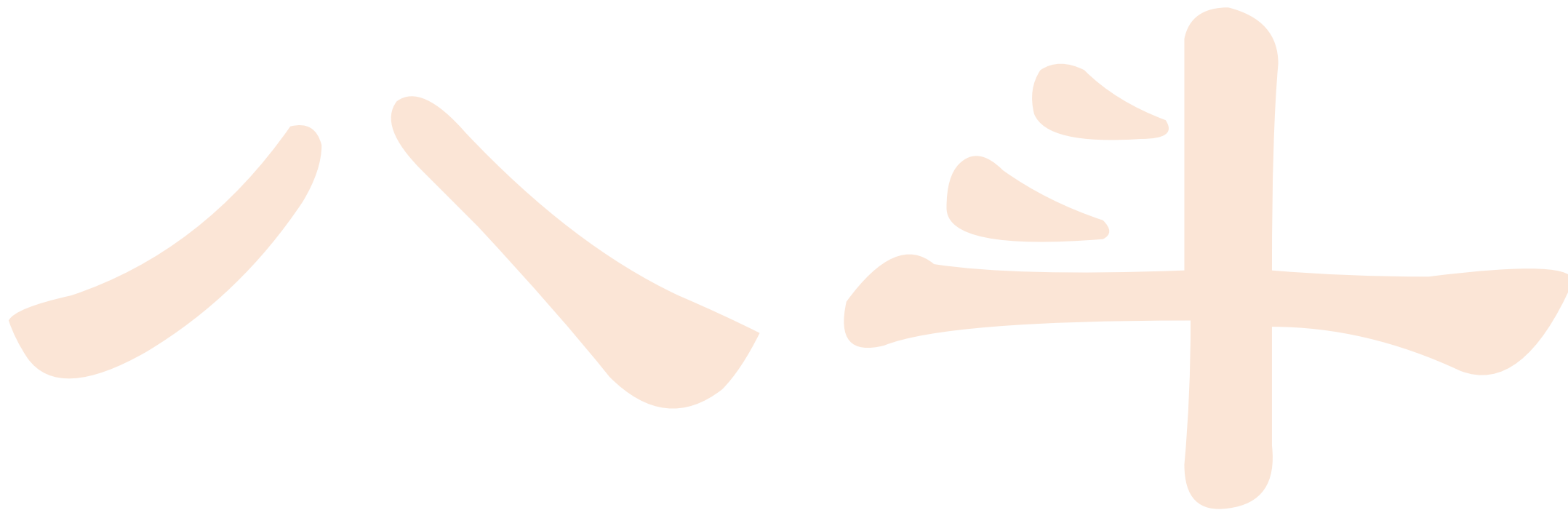
- 其中， $\frac{s_{1j} + s_{2j} + \dots + s_{mj}}{s}$ 是第j个子集的权，并且等于子集（即A值为 a_j ）中的样本个数除以S中的样本总数，其信息熵值越小，子集划分的纯度越高

- 其中，
$$I(s_{1j}, s_{2j}, \dots, s_{mj}) = - \sum_{i=1}^m p_{ij} \log(p_{ij}) \quad p_{ij} = \frac{s_{ij}}{|s_j|}$$

信息增益如何计算？

- 最后，实用信息增益确定决策树分支的划分依据

$$\text{Gain}(A) = I(s_1, s_2, \dots, s_m) - E(A)$$



信息增益计算举例

- 接前面例子:
- 类别标签S划分为两类: 买或不买

$$S_1(\text{买}) = 640$$

$$S_1(\text{不买}) = 384$$

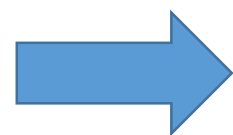
- 总体 $S = S_1 + S_2 = 1024$

$$p_1 = \frac{640}{1024} = 0.625$$

$$p_2 = \frac{384}{1024} = 0.375$$

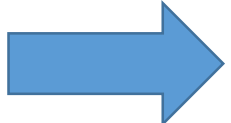
- 根据公式:

$$I(s_1, s_2, \dots, s_m) = - \sum_{i=1}^m p_i \log(p_i)$$



$$I(s_1, s_2) = 0.9544$$

信息增益计算举例

- 接下来计算每个特征的信息熵
- 1、先计算“年龄”特征的熵，共分3组，青年（0），中年（1），老年（2）
- 其中青年占总样本的概率为 $p(0)=384/1024=0.375$
- 青年中的买/不买=128/256
- 所以：S1(买)=128, $p1=128/384$
- S2(不买)=256, $p2=256/384$
- $S=S1+S2=384$
- 根据公式 $I(s_1, s_2, \dots, s_m) = - \sum_{i=1}^m p_i \log(p_i)$  $I(s_1, s_2) = 0.9183$

信息增益计算举例

- 其中中年占总样本的概率为 $p(1)=256/1024=0.25$
- 中年中的买/不买=256/0
- 所以: $S1(\text{买})=256$, $p1=1$
- $S2(\text{不买})=0$, $p2=0$
- $S=S1+S2=256$
- 根据公式

$$I(s_1, s_2, \dots, s_m) = - \sum_{i=1}^m p_i \log(p_i) \Rightarrow I(s_1, s_2) = 0$$

信息增益计算举例

- 其中老年占总样本的概率为 $p(1)=384/1024=0.375$
- 老年中的买/不买=256/128
- 所以: $S1(\text{买})=256$, $p1=256/384$
- $S2(\text{不买})=128$, $p2=128/384$
- $S=S1+S2=384$
- 根据公式

$$I(s_1, s_2, \dots, s_m) = - \sum_{i=1}^m p_i \log(p_i) \longrightarrow I(s_1, s_2) = 0.9157$$

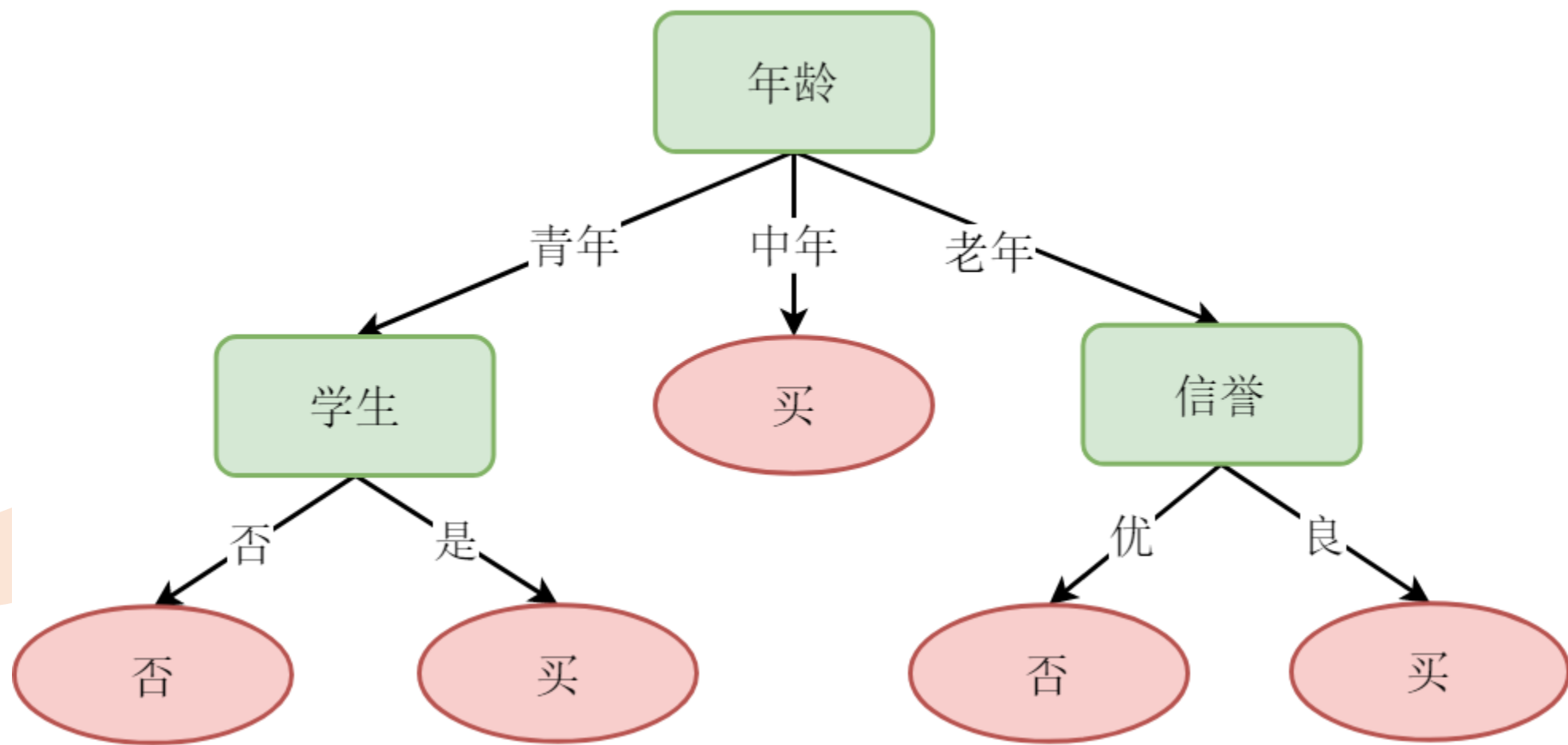
信息增益计算举例

- 所以汇总
- “年龄”的平均信息期望：
- $E(\text{年龄}) = 0.375 \times 0.9183 + 0.25 \times 0 + 0.375 \times 0.9157 = 0.6877$
- $G(\text{年龄}) = 0.9544 - 0.6877 = 0.2667$

- 同理：
- $E(\text{学生}) = 0.7811$ $G(\text{学生}) = 0.1733$
- $E(\text{收入}) = 0.9361$ $G(\text{收入}) = 0.0183$
- $E(\text{信誉}) = 0.9048$ $G(\text{信誉}) = 0.0496$

从所有特征中选择信息增益最大的作为根节点或者内部节点，根据计算，首次选取“年龄”来划分

信息增益计算举例



ID3 学习算法

- ID3(Dataset, attrList)

- 1、创建根节点R

- 2、如果当前Dataset中数据是“纯”的：将R的节点类型标记为当前类型

- 3、如果当attrList为空：将R的节点类型标记为当前Dataset中，样例个数最多的类型

- 4、其余情况：

- 1) 从attrList中选择属性A

- 2) 按照属性A所有的不同值 V_i ，将Dataset分为不同的子集 D_i ，对于每个 D_i

- a) 创建节点C

- b) 如果 D_i 为空：节点C标记为Dataset中，样例个数最多的类型

- c) D_i 不为空：节点C=ID3(D_i , attrList-A)

- d) 将节点C添加为R的子节点

- 5、返回R

终止条件

ID3 学习算法

- ID3(Dataset, attrList)

- 1、创建根节点R

- 2、如果当前Dataset中数据是“纯”的：将R的节点类型标记为当前类型

- 3、如果当attrList为空：将R的节点类型标记为当前Dataset中，样例个数最多的类型

- 4、其余情况：

- 1) 从attrList中选择属性A

- 2) 按照属性A所有的不同值 V_i ，将Dataset分为不同的子集 D_i ，对于每个 D_i

- a) 创建节点C

- b) 如果 D_i 为空：节点C标记为Dataset中，样例个数最多的类型

- c) D_i 不为空：节点C=ID3(D_i , attrList-A)

- d) 将节点C添加为R的子节点

- 5、返回R

终止条件

递归条件

ID3 学习算法

- ID3(Dataset, attrList)

1、创建根节点R

2、如果当前Dataset中数据是“纯”的：将R的节点类型标记为当前类型

3、如果当attrList为空：将R的节点类型标记为当前Dataset中，样例个数最多的类型

4、其余情况：

1) 从attrList中选择属性A

2) 按照属性A所有的不同值 V_i ，将Dataset分为不同的子集 D_i ，对于每个 D_i

a) 创建节点C

b) 如果 D_i 为空：节点C标记为Dataset中，样例个数最多的类型

c) D_i 不为空：节点C=ID3(D_i , attrList-A)

d) 将节点C添加为R的子节点

5、返回R

ID3 算法

• 缺点?

- 倾向于挑选属性值较多的属性，有些情况可能不会提供太多有价值的信息
 - 贪婪性
 - 奥卡姆剃刀原理：尽量用较少的东西做更多的事
- 不适用于连续变量

Outline

决策树简介

ID3算法

C4.5、CART算法

【实践】决策树

C4.5 算法

- 相对于ID3：
 - 克服了用信息增益选择属性时偏向选择取值多的属性的不足
 - 支持连续变量



信息增益率

- 信息增益率:

$$H(S) = - \sum_{i=1}^m p(u_i) \log(p(u_i))$$

$$Gain(S, A) = Entropy(S) - \sum_{v \in Value(A)} \frac{|S_v|}{|S|} Entropy(S_v)$$

$$GainRatio(A) = \frac{Gain(A)}{Entropy(A)}$$

属性A的分布情况，混乱度越大，
ratio越小，越纯净，ratio越大

C4.5 算法

- C4.5(Dataset, attrList)

1、创建根节点R

2、如果当前Dataset中数据是“纯”的或其他终止条件：将R的节点类型标记为当前类型

3、如果当attrList为空：将R的节点类型标记为当前Dataset中，样例个数最多的类型

4、其余情况：

1) 从attrList中选择gainratio最大的属性A

2) 按照属性A所有的不同值 V_i ，将Dataset分为不同的子集 D_i ，对于每个 D_i

a) 创建节点C

b) 如果 D_i 为空：节点C标记为Dataset中，属性值个数最多的类型

c) D_i 不为空：节点C=C4.5(D_i , attrList-A)

d) 将节点C添加为R的子节点

5、返回R

停止条件

- 信息增益（比例）增长低于阈值，则停止
 - 阈值需要调校
- 将数据分为训练集和测试集
 - 测试集上错误率增长，则停止
 - 没有用全部样本训练

剪枝

- 先较为充分地生长——过拟合
- 剪枝：
 - 相邻的叶子节点，如果合并后不纯度增加在允许范围内，则合并
 - 测试集错误率下降，则合并
 - 等等其他条件
- 反复尝试，较耗时间

叶子输出规则

- 叶子节点输出大多数样例所属的类别



CART 算法

• CART:

- 二叉回归树
- Gini系数

$$Gini(D) = \sum_{k=1}^{|y|} \sum_{k' \neq k} p_k p_{k'} = 1 - \sum_{k=1}^{|y|} p_k^2$$

$$Gini_Index(D, a) = \sum_{v=1}^V \frac{D^v}{D} Gini(D^v)$$

在候选属性集合中，选取使得划分后基尼系数最小的属性

CART 算法

有房者	婚姻状况	年收入	拖欠贷款者
是	单身	125K	否
否	已婚	100K	否
否	单身	70K	否
是	已婚	120K	否
否	离异	95K	是
否	已婚	60K	否
是	离异	220K	否
否	单身	85K	是
否	已婚	75K	否
否	单身	90K	是

- 按“有房”情况分析：

	有房	无房
否	3	4
是	0	3

$$\text{Gini}(t_1) = 1 - (3/3)^2 - (0/3)^2 = 0$$

$$\text{Gini}(t_2) = 1 - (4/7)^2 - (3/7)^2 = 0.4849$$

$$\text{Gini} = 0.3 \times 0 + 0.7 \times 0.4898 = 0.343$$

CART 算法

有房者	婚姻状况	年收入	拖欠贷款者
是	单身	125K	否
否	已婚	100K	否
否	单身	70K	否
是	已婚	120K	否
否	离异	95K	是
否	已婚	60K	否
是	离异	220K	否
否	单身	85K	是
否	已婚	75K	否
否	单身	90K	是

按“婚姻”情况分析：

	单身或已婚	离异
否	6	1
是	2	1

$Gini(t_1)=1-(6/8)^2-(2/8)^2=0.375$
 $Gini(t_2)=1-(1/2)^2-(1/2)^2=0.5$
 $Gini=8/10\times0.375+2/10\times0.5=0.4$

	单身或离异	已婚
否	3	4
是	3	0

$Gini(t_1)=1-(3/6)^2-(3/6)^2=0.5$
 $Gini(t_2)=1-(4/4)^2-(0/4)^2=0$
 $Gini=6/10\times0.5+4/10\times0=0.3$

	离异或已婚	单身
否	5	2
是	1	2

$Gini(t_1)=1-(5/6)^2-(1/6)^2=0.2778$
 $Gini(t_2)=1-(2/4)^2-(2/4)^2=0.5$
 $Gini=6/10\times0.2778+4/10\times0.5=0.3667$

CART 算法

有房者	婚姻状况	年收入	拖欠贷款者
是	单身	125K	否
否	已婚	100K	否
否	单身	70K	否
是	已婚	120K	否
否	离异	95K	是
否	已婚	60K	否
是	离异	220K	否
否	单身	85K	是
否	已婚	75K	否
否	单身	90K	是

- 按“收入”情况分析：

	60	70	75	85	90	95	100	120	125	220
	65	72	80	87	92	97	110	122	172	
	≤	>	≤	>	≤	>	≤	>	≤	>
是	0	3	0	3	0	3	1	2	2	1
否	1	6	2	5	3	4	3	4	3	4
Gini	0.400	0.375	0.343	0.417	0.400	0.300	0.343	0.375	0.400	

CART 算法逻辑

输入样本: $D = \{(x_1, y_1), (x_2, y_2), (x_3, y_3), \dots, (x_n, y_n)\}$

选择最优切变量j和切分点s:
$$\min_{j,s} [\min_{c_1} \sum_{x_i \in R_1(j,s)} (y_i - c_1)^2 + \min_{c_2} \sum_{x_i \in R_2(j,s)} (y_i - c_2)^2]$$

用选定的(j,s), 划分二区域, 并决定输出值: $R_1(j,s) = \{x | x^{(j)} \leq s\}, R_2(j,s) = \{x | x^{(j)} > s\}$

$$\hat{c}_m = \frac{1}{N_m} \sum_{x_i \in R_m(j,s)} y_i$$

$$x \in R_m, m = 1, 2$$

对两个子区域调用上述步骤, 将输入空间划分为M个区域: R_1, R_2, \dots, R_m , 生成决策树

当空间划分确定, 用平方误差表示预测方法, 用平方误差最小的准则, 求解每个单元上的最优输出值:

$$f(x) = \sum_{m=1}^M \hat{c}_m I(x \in R_m) \quad \sum_{x_i \in R_m} (y_i - f(x_i))^2$$

CART 算法逻辑

x_i	1	2	3	4	5	6	7	8	9	10
y_i	5.56	5.70	5.91	6.40	6.80	7.05	8.90	8.70	9.00	9.05

切分点: 1.5,2.5,3.5,4.5,5.5,6.5,7.5,8.5,9.5

对各切分点依次求出 $R1, R2, c1, c2$ 及 $m(s)$ 当: $s=1.5$ $R1=\{1\}$ $R2=\{2,3,4,5,6,7,8,9,10\}$

$$c_1 = \frac{1}{N_m} \sum_{x_i \in R_m(j,s)} y_i = \frac{1}{1} \sum_{x_i \in R_1(1,1.5)} 5.56 = 5.56$$

$$c_2 = \frac{1}{N_m} \sum_{x_i \in R_m(j,s)} y_i = \frac{1}{9} \sum_{x_i \in R_2(1,1.5)} (5.70 + 5.91 + \dots + 9.05) = 7.50$$

$$m(s) = \min_{j,s} [\min_{c_1} \sum_{x_i \in R_i(j,s)} (y_i - c_1)^2 + \min_{c_2} \sum_{x_i \in R_i(j,s)} (y_i - c_1)^2] = 0 + 15.72 = 15.72$$

依次改变 j, s :

s	1.5	2.5	3.5	4.5	5.5	6.5	7.5	8.5	9.5
$m(s)$	15.72	12.07	8.36	5.78	3.91	1.93	8.01	11.73	15.74

CART 算法逻辑

依次改变j,s:

x_i	1	2	3	4	5	6	7	8	9	10
y_i	5.56	5.70	5.91	6.40	6.80	7.05	8.90	8.70	9.00	9.05

s	1.5	2.5	3.5	4.5	5.5	6.5	7.5	8.5	9.5
$m(s)$	15.72	12.07	8.36	5.78	3.91	1.93	8.01	11.73	15.74

当: $s=6.5$ $R1=\{1,2,3,4,5,6\}$ $R2=\{7,8,9,10\}$

$$T_1(x) = \begin{cases} 6.24, & x < 6.5 \\ 8.91, & x \geq 6.5 \end{cases}$$

x_i	1	2	3	4	5	6	7	8	9	10
y_i	-0.68	-0.54	-0.33	0.16	0.56	0.81	-0.01	-0.21	0.09	0.14

$$f_1(x) = T_1(x)$$

此时 $f_1(x)$ 拟合数据得到平方误差: $L(y, f_1(x)) = \sum_{i=1}^{10} (y_i - f_1(x_i))^2 = 1.93$

——八斗大数据内部资料，盗版必究——

CART 算法逻辑

x_i	1	2	3	4	5	6	7	8	9	10
y_i	-0.68	-0.54	-0.33	0.16	0.56	0.81	-0.01	-0.21	0.09	0.14

第二步：求 $T_2(x)$ 参考 $T_1(x)$

$$T_1(x) = \begin{cases} 6.24, x < 6.5 \\ 8.91, x \geq 6.5 \end{cases}$$

$$f_1(x) = T_1(x)$$

$$T_2(x) = \begin{cases} -0.52, x < 3.5 \\ 0.22, x \geq 3.5 \end{cases}$$

$$f_2(x) = f_1(x) + T_2(x) = \begin{cases} 5.72, x < 3.5 \\ 6.46, 3.5 \leq x \leq 6.5 \\ 9.13, x \geq 6.5 \end{cases}$$

以此类推：

$$T_3(x) = \begin{cases} 0.15, x < 6.5 \\ -0.22, x \geq 6.5 \end{cases} L(y, f_3(x)) = 0.47$$

$$T_4(x) = \begin{cases} -0.16, x < 4.5 \\ 0.11, x \geq 4.5 \end{cases} L(y, f_4(x)) = 0.30$$

$$T_5(x) = \begin{cases} 0.07, x < 6.5 \\ -0.11, x \geq 6.5 \end{cases} L(y, f_5(x)) = 0.23$$

$$T_6(x) = \begin{cases} -0.15, x < 2.5 \\ 0.04, x \geq 2.5 \end{cases}$$

$$f_6(x) = f_5(x) + T_6(x) = T_1(x) + \dots + T_6(x) = \begin{cases} 5.63, x < 2.5 \\ 5.82, 2.5 \leq x \leq 3.5 \\ 6.56, 3.5 \leq x \leq 4.5 \\ 6.83, 4.5 \leq x \leq 6.5 \\ 8.95, x \geq 6.5 \end{cases}$$

Outline

决策树简介

ID3算法

C4.5、CART算法

【实践】决策树

Q & A

@八斗学院
