机器学习基础

决策树

黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

- 决策树模型与建立
- 特征选择
- 决策树的生成
- 决策树的剪枝
- 分类回归树（CART）
- 总结

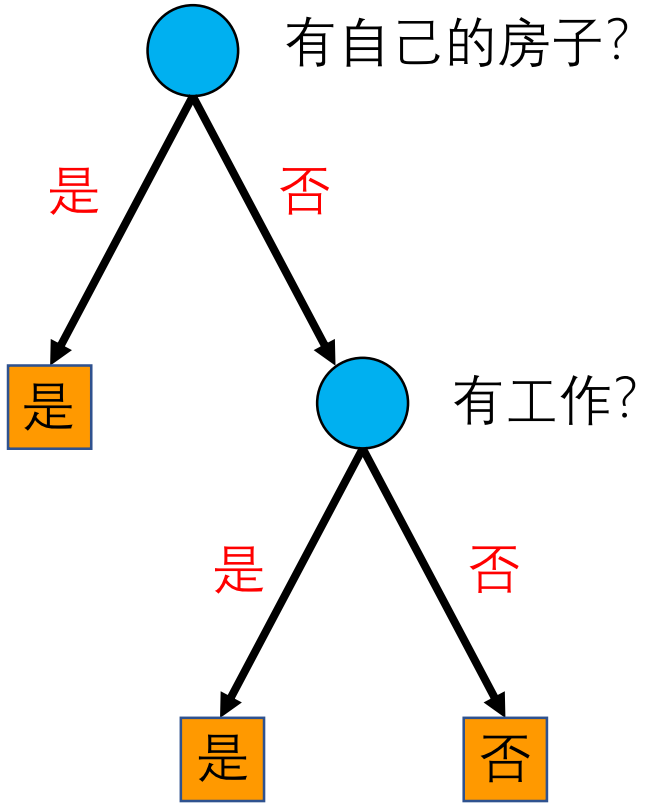
黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

- 决策树模型与建立
- 特征选择
- 决策树的生成
- 决策树的剪枝
- 分类回归树（CART）
- 总结

黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

决策树模型

- 分类决策树模型是一种描述对实例进行分类的树形结构
  - 节点 (node)
    - 内部节点表示样本的一个特征或属性
    - 叶节点表示一个分类结果
  - 有向边 (directed edge)：决策过程



黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

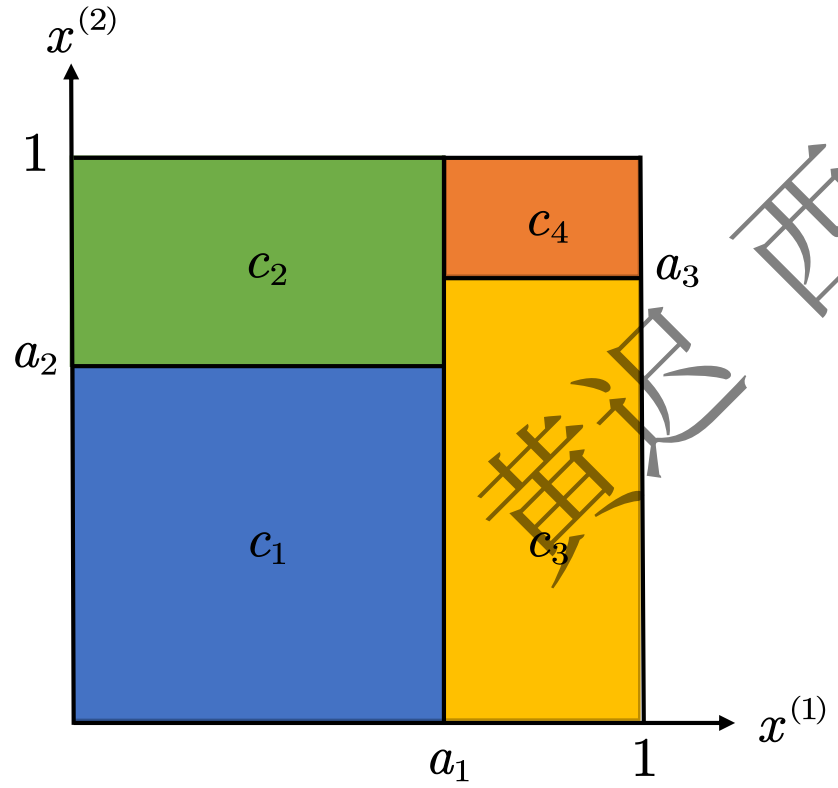
决策过程

- 从根节点开始将样本的某一特征进行测试，并根据测试结果将其分配到某个子节点
- 递归的对实例进行测试分配直至到达叶节点
- 将样本分到叶节点的类中
- 决策树可以看成 if-then 规则的集合
  - 内部节点对应规则的条件；叶节点对应规则的结论

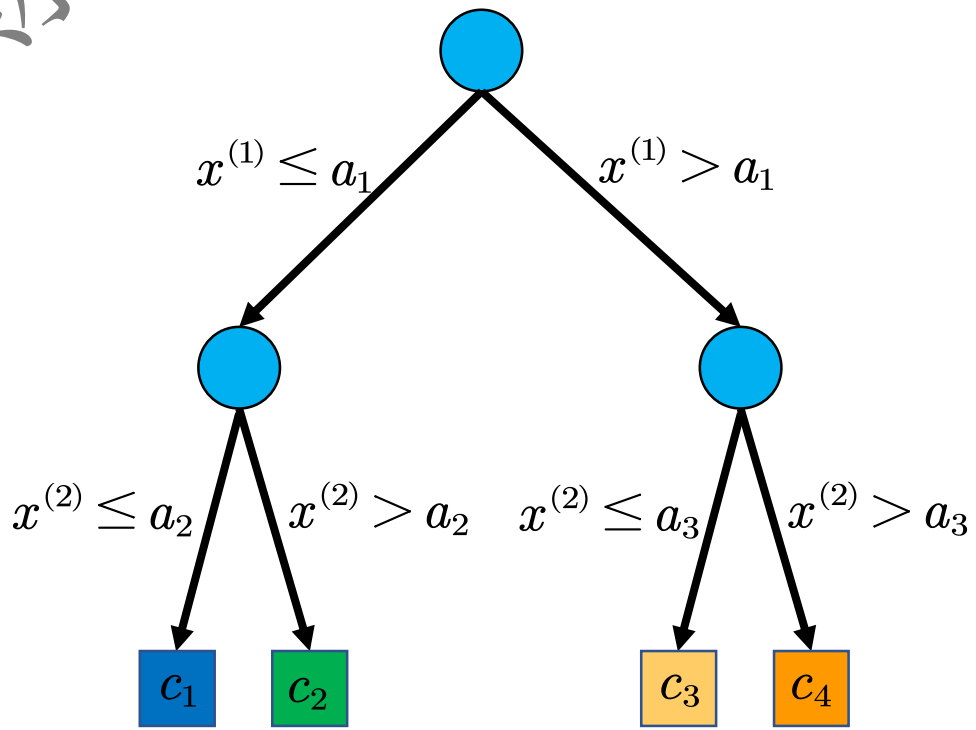
黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

决策树与特征空间划分

- 特征空间划分



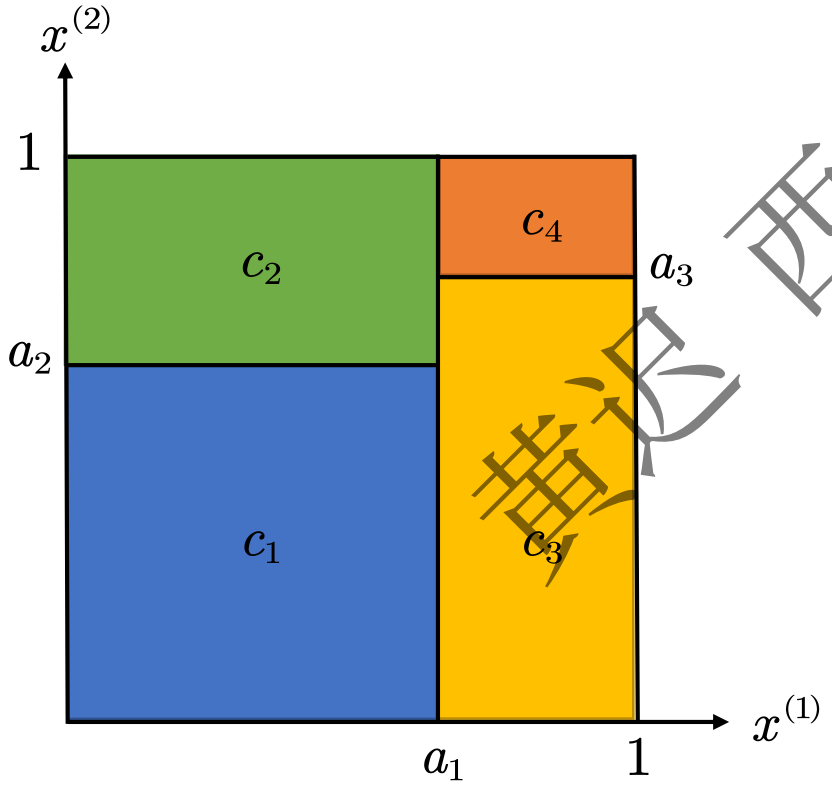
- 对应决策树

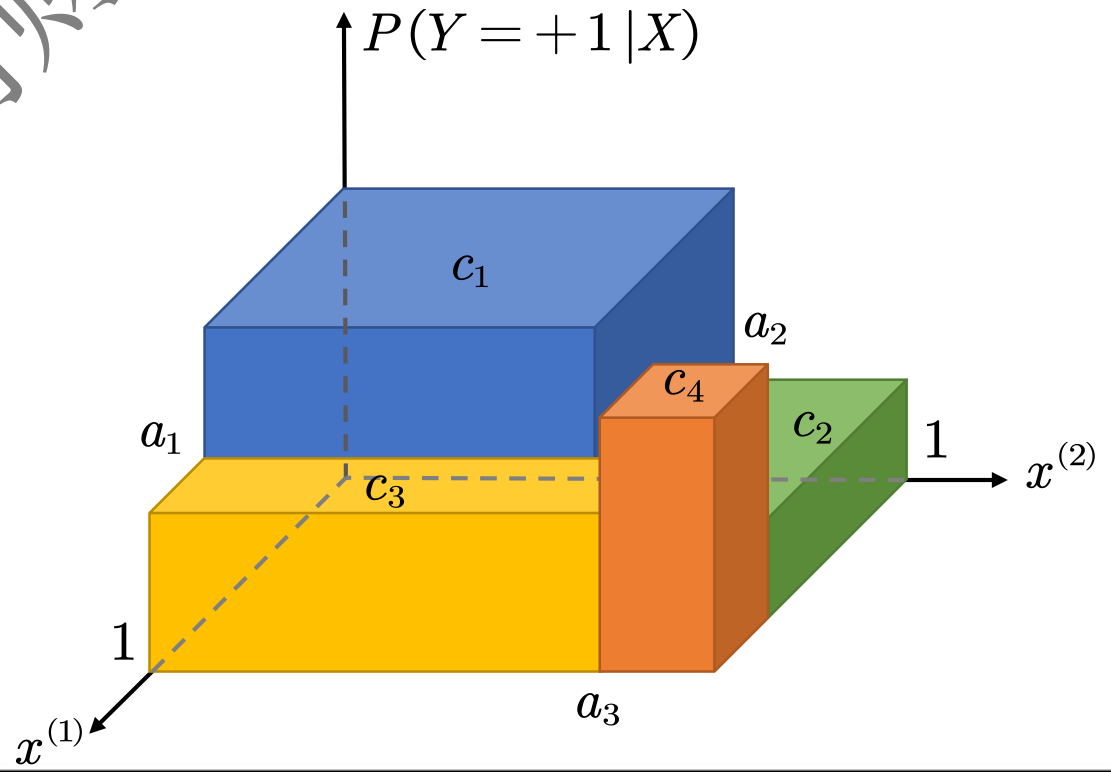


黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

决策树与特征空间划分

- 划分单元的条件概率分布

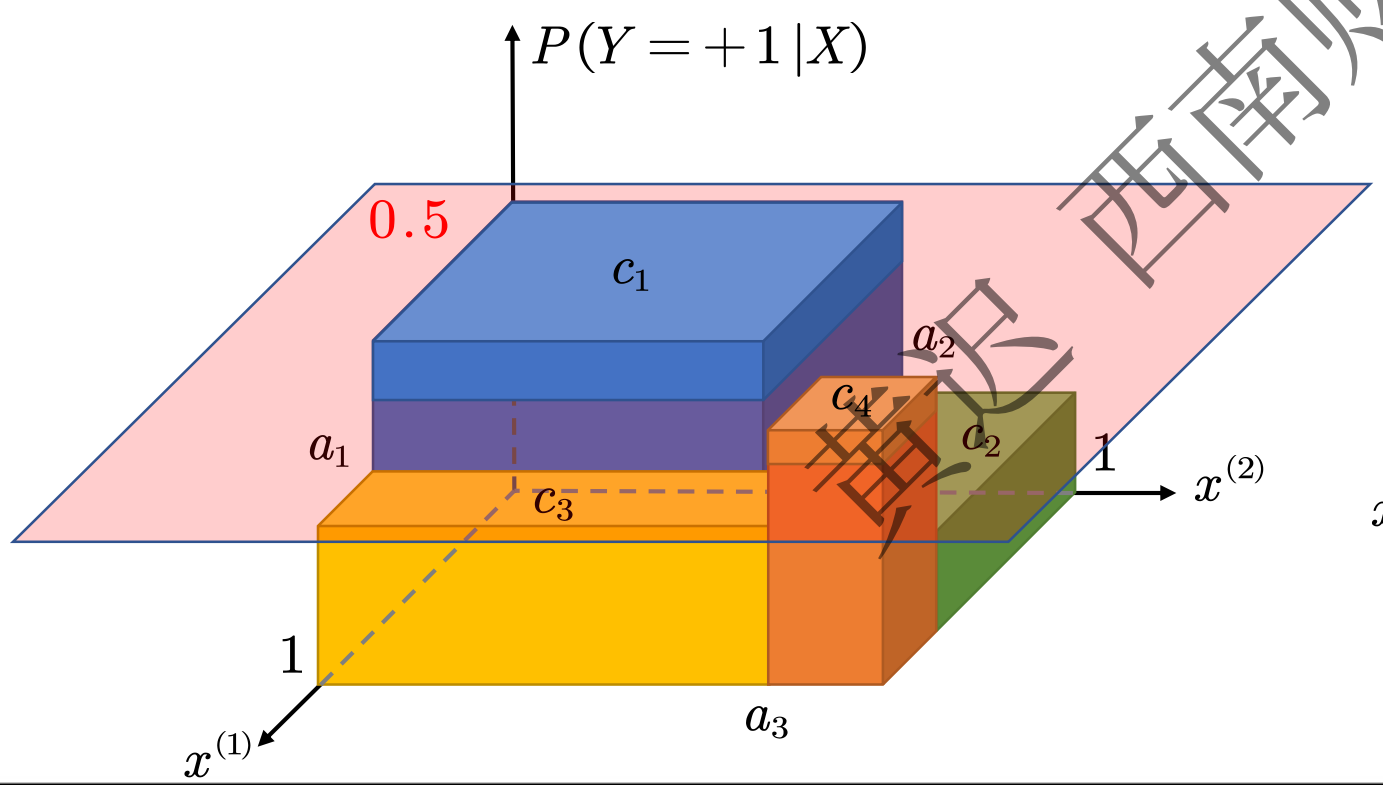


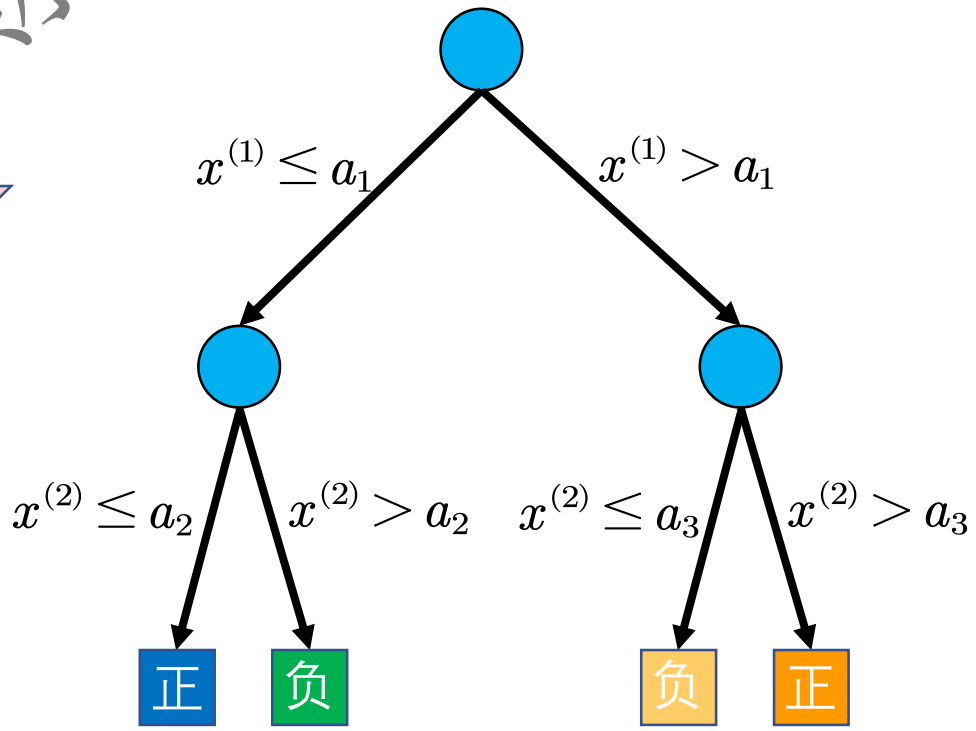


黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

决策树与特征空间划分

- 条件概率 $P(Y=+1|X=c)>0.5$ 为正类，否则为负类





黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

决策树的建立

- 对特征空间进行划分，并估计条件概率进行分类
- 决策树算法递归的选择特征，并根据该特征划分空间
  - 将所有训练样本放在根节点
  - 选择一个最优特征，按照这一特征将样本分割成子集
    - 若这些子集中的样本能基本正确分类，则构建叶节点
    - 若不能正确分类，对该子集选择新的最优特征，继续分割
    - 如此递归，直至所有子集中的样本基本正确分类

黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

- 决策树模型与建立
- 特征选择
- 决策树的生成
- 决策树的剪枝
- 分类回归树（CART）
- 总结

黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

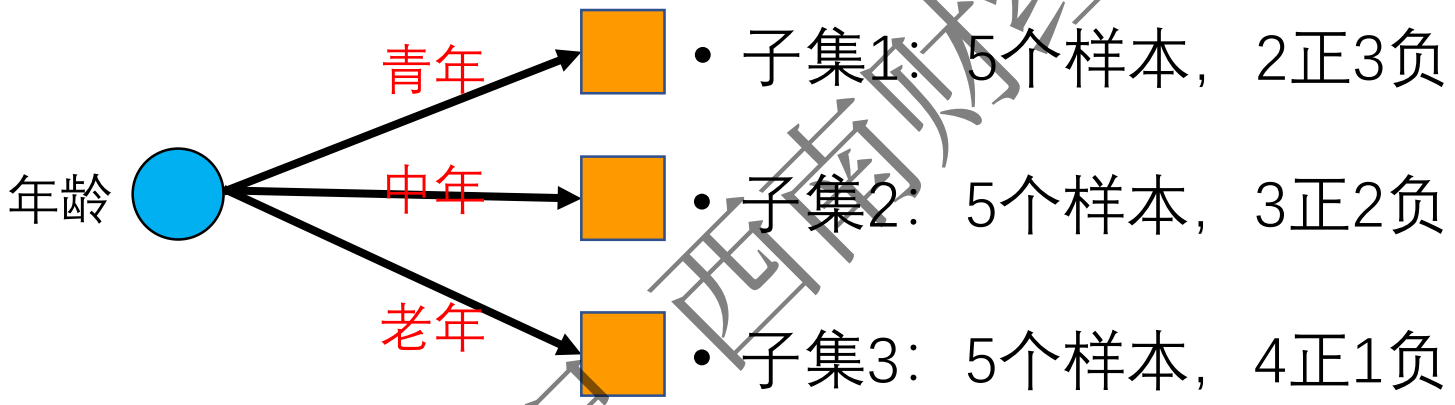
特征选择：

年龄	有工作	有自己房子	信贷情况	类别
青年	否	否	一般	否
青年	否	否	好	否
青年	是	否	好	是
青年	是	是	一般	是
青年	否	否	一般	否
中年	否	否	一般	否
中年	否	否	好	否
中年	是	是	好	是
中年	否	是	非常好	是
中年	否	是	非常好	是
老年	否	是	非常好	是
老年	否	是	好	是
老年	是	否	好	是
老年	是	否	非常好	是
老年	否	否	一般	否

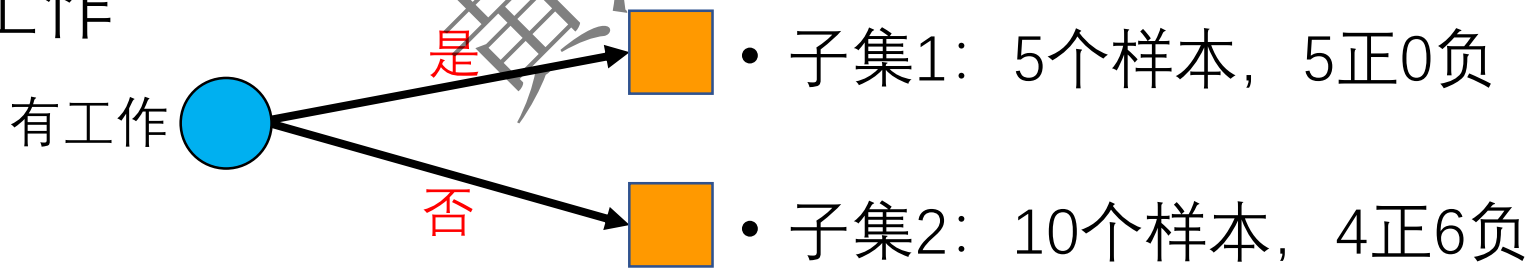
黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

特征选择

- 年龄



- 有工作



黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

熵 (entropy)

- 熵表示随机变量不确定性的度量
  - 熵越大，随机变量不确定性越大
  - 假设随机变量 X 服从分布 $P(X=x_i)=p_i,\ i=1,2,\cdots,n$
  - 熵 $H(X)=\mathbb{E}_{p_x}\log\frac{1}{p_i}=\sum_{i=1}^{n}p_i\log\frac{1}{p_i}$
  - 可以验证 $0\le H(X)\le\log n$

黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

条件熵 (conditional entropy)

- 设随机变量 X 和 Y 的联合分布为

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

- X 和 Y 的联合熵为

$$H(X,Y)=\mathbb{E}_{p_{xy}}\log\frac{1}{p_{ij}}=\sum_{i=1}^{n}\sum_{j=1}^{m}p_{ij}\log\frac{1}{p_{ij}}$$

黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

条件熵 (conditional entropy)

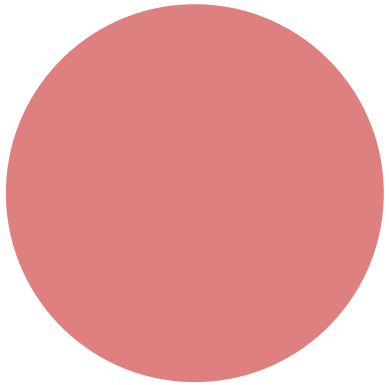
- 设随机变量 X 和 Y 的联合分布为

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

- X 和 Y 的联合熵为

$$H(X,Y)=\mathbb{E}_{p_{xy}}\log\frac{1}{p_{ij}}=\sum_{i=1}^{n}\sum_{j=1}^{m}p_{ij}\log\frac{1}{p_{ij}}$$

$H(X)$



黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

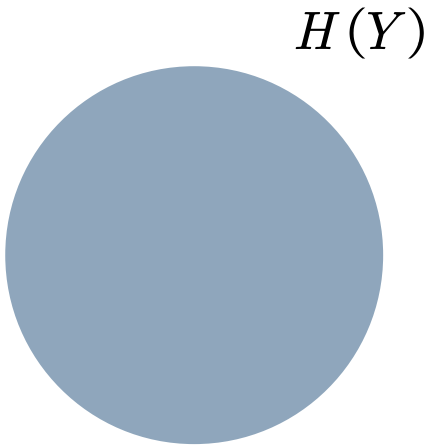
条件熵 (conditional entropy)

- 设随机变量 X 和 Y 的联合分布为

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

- X 和 Y 的联合熵为

$$H(X,Y)=\mathbb{E}_{p_{xy}}\log\frac{1}{p_{ij}}=\sum_{i=1}^{n}\sum_{j=1}^{m}p_{ij}\log\frac{1}{p_{ij}}$$



黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

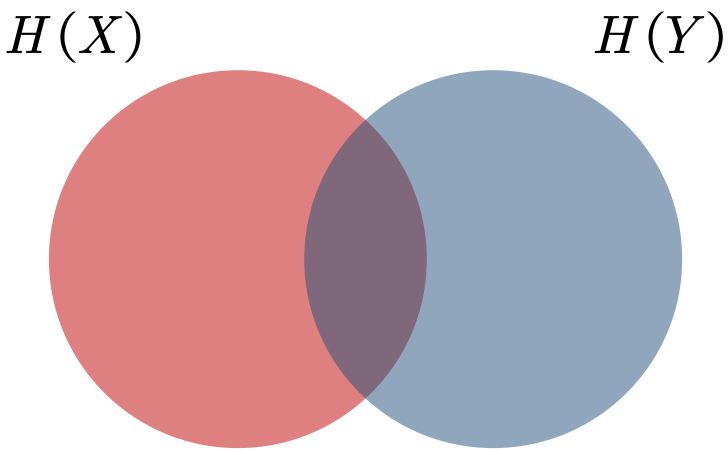
条件熵 (conditional entropy)

- 设随机变量 X 和 Y 的联合分布为

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

- X 和 Y 的联合熵为

$$H(X,Y)=\mathbb{E}_{p_{xy}}\log\frac{1}{p_{ij}}=\sum_{i=1}^{n}\sum_{j=1}^{m}p_{ij}\log\frac{1}{p_{ij}}$$



黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

条件熵 (conditional entropy)

- 设随机变量 X 和 Y 的联合分布为

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

- X 和 Y 的联合熵为

$$H(X,Y)=\mathbb{E}_{p_{xy}}\log\frac{1}{p_{ij}}=\sum_{i=1}^{n}\sum_{j=1}^{m}p_{ij}\log\frac{1}{p_{ij}}$$

- X 给定条件下 Y 的条件熵

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

$H(X,Y)$

黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

条件熵 (conditional entropy)

- 设随机变量 X 和 Y 的联合分布为

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

- X 和 Y 的联合熵为

$$H(X,Y)=\mathbb{E}_{p_{xy}}\log\frac{1}{p_{ij}}=\sum_{i=1}^{n}\sum_{j=1}^{m}p_{ij}\log\frac{1}{p_{ij}}$$

- X 给定条件下 Y 的条件熵

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

  - 已知 X 的信息后，Y 仍然存在的不确定性

$$H(Y|X)=H(X,Y)-H(X)$$

黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

信息增益 (information gain)

- 概率由数据统计（极大似然估计）得到，所对应的熵和条件熵称为经验熵和经验条件熵
- 特征 A 对数据集 D 的信息增益 $g(D,A)$ 定义为

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

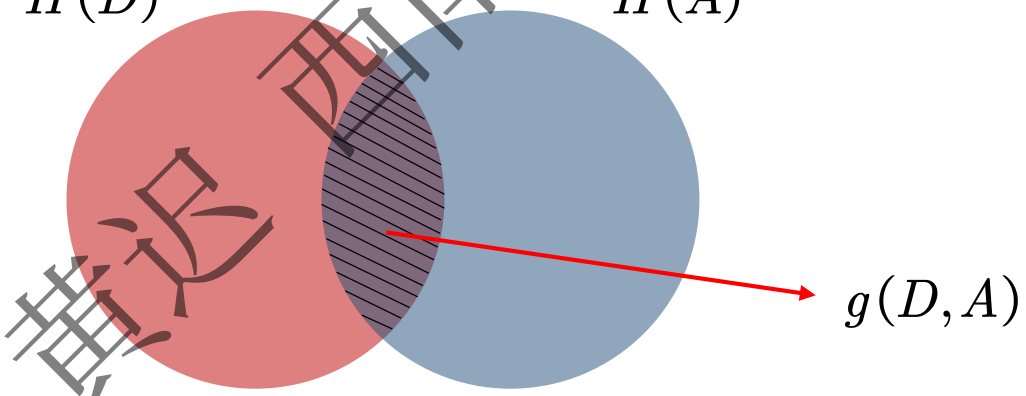
- 信息增益表示得知特征 A 的信息而使得数据集 D 的分类不确定性减少的程度

黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

信息增益 (information gain)

- 在信息论中，信息增益就是互信息 (mutual information)

$$I(D;A) = g(D,A) = H(D) + H(A) - H(D,A)$$



- 互信息是 D 和 A 的共同信息，即已知 A 后有助于了解 D 的程度

黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

信息增益 (information gain)

- $H(D)$表示对数据集 D 进行分类的不确定性（纯度）
  - 熵越大，不确定性越大，纯度越小
- $H(D|A)$ 表示特征 A 给后，对数据集 D 进行分类的不确定性
  - 条件熵越大，不确定性越大，特征对分类的帮助越小
- $g(D,A)$ 由于特征 A 使得对数据集 D 的分类不确定性减小程度
  - 信息增益越大，特征对减少分类不确定性的效果越好
  - 从信息论角度看，能从特征获得数据集更多的信息

黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

信息增益算法

输入：训练数据集 D 和特征 A；

输出：特征 A 对训练数据集 D 的信息增益 $g(D,A)$。

（1）计算数据集 D 的经验熵 $H(D)$

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

（2）计算特征 A 对数据集 D 的经验条件熵 $H(D|A)$

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

（3）计算信息增益

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

黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

信息增益比 (information gain ratio)

- 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
- 信息增益比

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

  - 其中 $H_A(D)=-\sum_{i=1}^{n}\frac{|D_i|}{|D|}\log_2\frac{|D_i|}{|D|}$，$n$ 表示特征 A 的取值个数

黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

基尼指数 (Gini index)

- 有K个类，样本属于第k类的概率为p_k，则概率分布$\mathbf{p}=(p_1,\cdots,p_K)$的基尼指数为 $\mathrm{Gini}(\mathbf{p})=\sum_{k=1}^{K}p_k(1-p_k)=1-\sum_{k=1}^{K}p_k^2$
- 当$K=2$且第一个分类的概率为p，则基尼指数为 $\mathrm{Gini}(\mathbf{p})=2p(1-p)$
- 对样本集D其基尼指数为

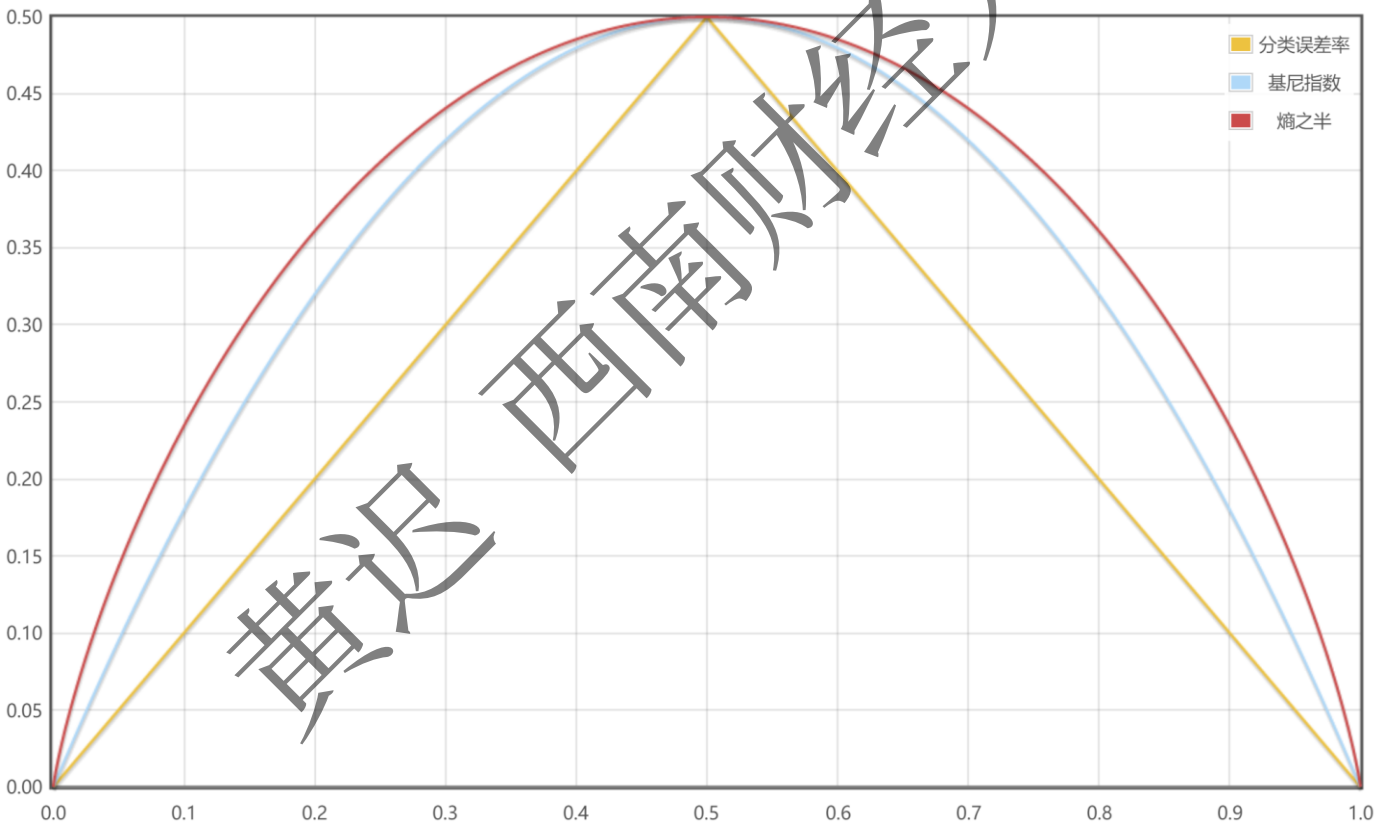
$$\mathrm{Gini}(D)=1-\sum_{k=1}^{K}\left(\frac{|C_k|}{|D|}\right)^2$$

其中C_k是第k类样本的集合

- 基尼指数表示D的不确定性，基尼指数越大不确定性越大

黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

基尼指数、熵之半、分类误差率



黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

- 决策树模型与建立
- 特征选择
- 决策树的生成
- 决策树的剪枝
- 分类回归树（CART）
- 总结

黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

ID3算法

- 从根节点开始，计算所有可能特征的信息增益（特征不重复）
- 选择信息增益最大的特征，并根据特征取值建立子节点
- 再对子节点递归调用以上方法，构建决策树
- 递归终止条件
  - 节点所有样本属于同一类，将该类作为节点的类标记
  - 无可选特征，节点中样本最多的类作为类标记
  - 信息增益小于给定阈值，节点中样本最多的类作为类标记

黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

C4.5算法

- 与ID3算法类似，C4.5用信息增益比选择特征
  - 选择信息增益比最大的特征，并根据特征取值建立子节点
- 递归终止条件
  - 节点所有样本属于同一类，将该类作为节点的类标记
  - 无可选特征，节点中样本最多的类作为类标记
  - 信息增益比小于给定阈值，节点中样本最多的类作为类标记

黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

- 决策树模型与建立
- 特征选择
- 决策树的生成
- 决策树的剪枝
- 分类回归树（CART）
- 总结

黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

剪枝的必要性

- 递归终止条件表明，决策树对训练数据的分类很准确
- 但往往对未知数据分类不准确（过拟合）
- 原因在于特征空间划分过于细致，决策树太复杂
- 需要在决策树建立过程中或已生成的决策树进行简化
  - 前者被称为预剪枝：提前终止某些分支的生长
  - 后者被称为后剪枝：生成一棵完全树，再“回头”剪枝
- 决策树的生成考虑局部最优，决策树的剪枝考虑全局最优

黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

划分选择 vs. 剪枝

- 研究表明: 特征选择的各种准则虽然对决策树的尺寸有较大影响，但对泛化性能的影响很有限
  - 例如信息增益与基尼指数产生决策树的分类结果，仅在约2% 的情况下不同
- 剪枝方法和程度对决策树泛化性能的影响更为显著
  - 在数据带噪时甚至可能将泛化性能提升25%
- 剪枝是决策树处理过拟合的主要手段

黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

预剪枝 vs. 后剪枝

- 时间开销
  - 预剪枝：训练时间开销降低，测试时间开销降低
  - 后剪枝：训练时间开销增加，测试时间开销降低
- 过/欠拟合风险
  - 预剪枝：过拟合风险降低，欠拟合风险增加
  - 后剪枝：过拟合风险降低，欠拟合风险基本不变
- 泛化性能：后剪枝通常优于预剪枝

黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

损失函数

- 构造决策树的损失函数

$$C_\alpha(T)=\sum_{t=1}^{|T|}N_tH_t(T)+\alpha|T|$$

经验风险　正则化项

其中 $H_t(T)=-\sum_k\frac{N_{tk}}{N_t}\log\frac{N_{tk}}{N_t}$ 表示第 t 个叶节点的分类效果

- 令 $C(T)=\sum_{t=1}^{|T|}N_tH_t(T)=-\sum_{t=1}^{|T|}\sum_{k=1}^{K}N_{tk}\log\frac{N_{tk}}{N_t}$

有 $C_\alpha(T)=C(T)+\alpha|T|$

黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

损失函数

- 由于 $C(T)=-\sum_{t=1}^{|T|}\sum_{k=1}^{K}N_{tk}\log\frac{N_{tk}}{N_t}=-\log\prod_{t=1}^{|T|}\prod_{k=1}^{K}\left(\frac{N_{tk}}{N_t}\right)^{N_{tk}}$
- 所以 $\min C(T)$ 等价于

$$\max\log\prod_{t=1}^{|T|}\prod_{k=1}^{K}\left(\frac{N_{tk}}{N_t}\right)^{N_{tk}}$$

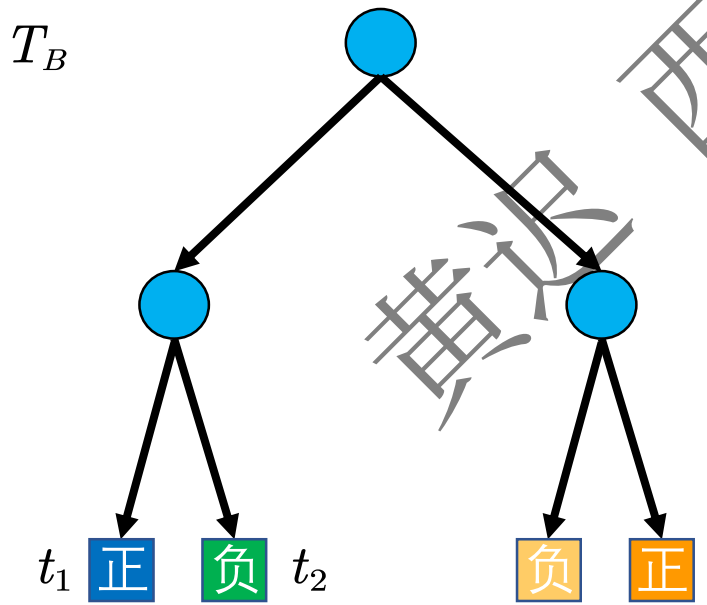
对数极大似然函数

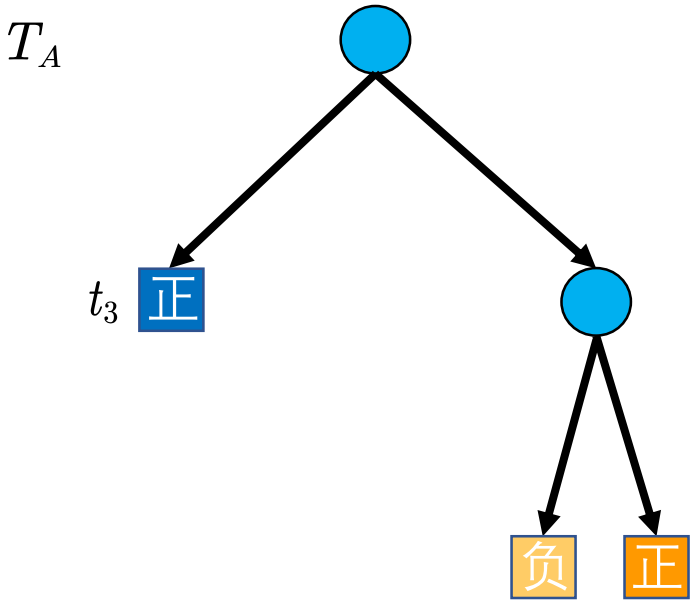
- 决策树的损失函数的极小化等价于正则化的极大似然估计

黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

剪枝算法

- 计算每个节点的经验熵
- 递归的从树的叶节点向上回缩





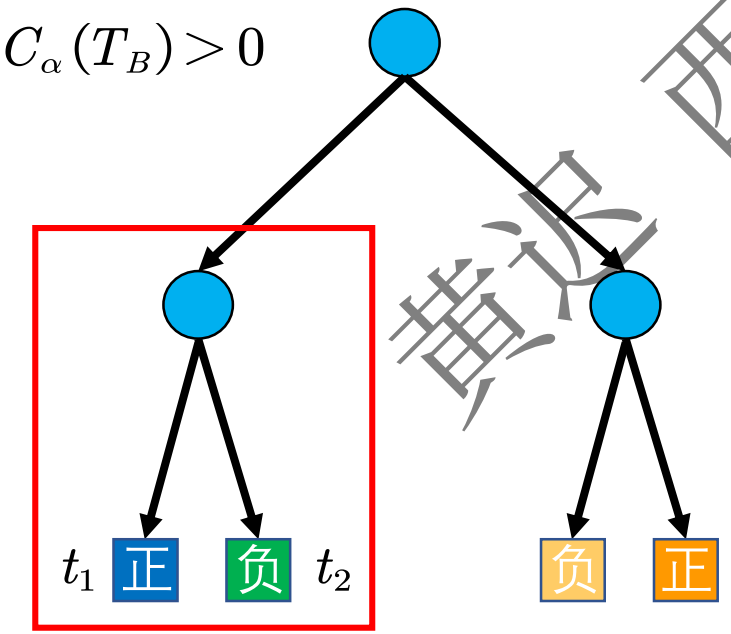
黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

剪枝算法

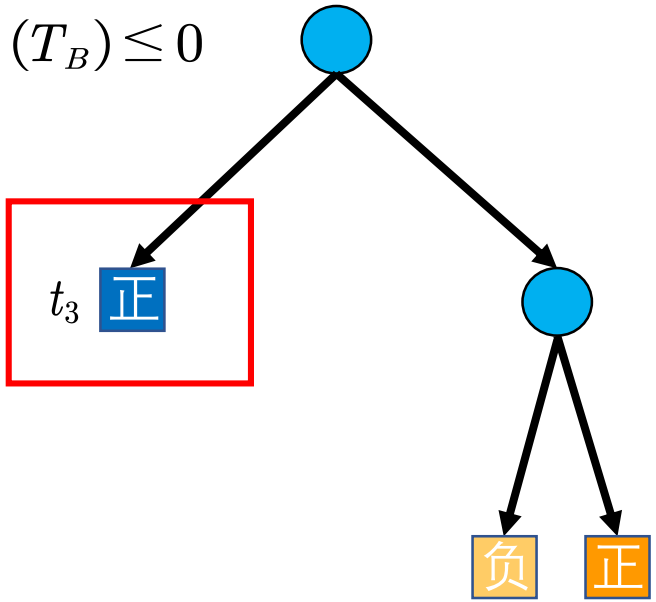
$$C_\alpha(T_A)-C_\alpha(T_B)=\sum_{t=1}^{|T_A|}N_tH_t(T_A)+\alpha|T_A|-\sum_{t=1}^{|T_B|}N_tH_t(T_B)-\alpha|T_B|$$

$$=N_{t_3}H_{t_3}(T_A)-N_{t_1}H_{t_1}(T_B)-N_{t_2}H_{t_2}(T_B)-\alpha$$

$C_\alpha(T_A)-C_\alpha(T_B)>0$



$C_\alpha(T_A)-C_\alpha(T_B)\leq 0$



黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

- 决策树模型与建立
- 特征选择
- 决策树的生成
- 决策树的剪枝
- 分类回归树（CART）
- 总结

黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

CART算法

- CART是二叉树，既可以用来分类也可以用于回归
- CART算法
  - 决策树的生成：基于训练数据集生成决策树，生成的决策树要尽量大
  - 决策树剪枝：用验证数据集对已生成的树进行剪枝并选择最优子树，这时用损失函数最小作为剪枝的标准

黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

回归树的生成

- 回归树将特征空间划分为M个单元$R_1, R_2, \cdots, R_M$
- 在每个单元R_m，有固定输出c_m
- 回归树模型为$f(x) = \sum_{m=1}^{M} c_m I(x \in R_m)$
- 当单元划分确定时，R_m上的c_m的最优值$\hat{c}_m$是使平方误差$\sum_{x_i \in R_m} (y_i - f(x_i))^2$最小，即$\hat{c}_m = \text{ave}(y_i | x_i \in R_m)$

黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

单元划分

- 对第 j 个特征和它的一个取值 s，定义区域

$$R_1(j,s)=\{x|x^{(j)}\le s\},\ R_2(j,s)=\{x|x^{(j)}>s\}$$

- 通过以下优化问题寻找最优特征 j 和切分点 s

$$\min_{j,s}\left[\sum_{x_i\in R_1(j,s)}(y_i-\hat{c}_1)^2+\sum_{x_i\in R_2(j,s)}(y_i-\hat{c}_2)^2\right]$$

其中 $\hat{c}_1=\text{ave}(y_i|x_i\in R_1(j,s)),\ \hat{c}_2=\text{ave}(y_i|x_i\in R_2(j,s))$

黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

建立回归树

- 对解得的最优值(j,s)，得到划分

$$R_1(j,s)=\{x|x^{(j)}\leq s\},\ R_2(j,s)=\{x|x^{(j)}>s\}$$

和输出值 $\hat{c}_1=\text{ave}(y_i|x_i\in R_1(j,s)),\ \hat{c}_2=\text{ave}(y_i|x_i\in R_2(j,s))$

- 继续对区域$R_1(j,s),\ R_2(j,s)$寻找最优划分和输出值，直至满足停止条件
- 生成回归树 $f(x)=\sum_{m=1}^{M}\hat{c}_m I(x\in R_m)$

黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

分类树的生成

- 对每一个特征 A 的每个值 a，将训练集 D 分成两个子集

$$D_1=\{x_i|x_i^{(A)}=a\},\ D_2=\{x_i|x_i^{(A)}\neq a\}$$

- 计算基尼指数 $\mathrm{Gini}(D,A)=\frac{|D_1|}{D}\mathrm{Gini}(D_1)+\frac{|D_2|}{D}\mathrm{Gini}(D_2)$
  - $\mathrm{Gini}(D,A)$ 表示经 $A=a$ 分割后 D 的不确定性（类似条件熵）
- 对所有特征的每个值计算 $\mathrm{Gini}(D,A)$，选择基尼指数最小的 (A,a)，作为切分点将训练数据集分为 D_1,D_2，并作为现节点的子节点
- 对两个子节点递归调用算法，直至满足停止条件

黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

CART的停止条件

- 节点中的样本个数小于预定阈值
- 样本集的平方误差小于预定阈值
- 节点的基尼指数小于给定阈值
- 没有特征可用
  - ID3，C4.5是多叉树，每次选择的特征在样本分裂时已经用了所有信息，所以特征不会重复选用
  - CART每次只用了某个特征的一个值，因此可以重复用

黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

CART剪枝

- 从生成算法产生的决策树T_0底端开始不断剪枝，直到T_0的根节点，形成一个子树序列$\{T_0, T_1, \cdots, T_n\}$
- 通过交叉验证法在独立的验证数据集上对子树序列进行测试，从中选择最优子树

黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

生成子树序列

- 构造损失函数 $C_\alpha(T)=C(T)+\alpha|T|$
- 对给定的α，定义 $T_\alpha=\arg\min_{T\subseteq T_0}C_\alpha(T)$，即损失函数最小的子树
  - 这样的最优子树在结构最简意义下是唯一
    - 假设有两棵最优子树，则它们一定不会有包含关系
    - 那么取两棵树的交集是更小的最优子树，矛盾
- 当 α 大时，T_α偏小；当 α 小时，T_α偏大

黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

生成子树序列

- 将α从小到大增大，产生一系列的区间$[\alpha_i, \alpha_{i+1})$, $i=0,1,2,\cdots,n$，其中$0=\alpha_0<\alpha_1<\cdots<\alpha_n<+\infty$
- 在每个区间$[\alpha_i, \alpha_{i+1})$有唯一的最优子树T_i
- 可以证明最优子树序列$\{T_0, T_1, \cdots, T_n\}$中的子树是嵌套的

黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

生成子树序列

- 以 T_0 的任意内部节点 t 为单节点树的损失函数为 $C_\alpha(t)=C(t)+\alpha$
- 以 t 为根节点的子树 T_t 的损失函数为 $C_\alpha(T_t)=C(T_t)+\alpha|T_t|$
- 当 α 较小时，有 $C_\alpha(T_t)<C_\alpha(t)$（不剪枝损失函数更小）
- 当 α 增大到 $\frac{C(t)-C(T_t)}{|T_t|-1}$ 时，有 $C_\alpha(T_t)=C_\alpha(t)$，对 T_t 进行剪枝
- 令 $t_1=\arg\min_{t\in T_0}\frac{C(t)-C(T_t)}{|T_t|-1}$，则 T_1 是 T_0 在 t_1 处剪枝的子树，且 $\alpha_1=\frac{C(t_1)-C(T_{t_1})}{|T_{t_1}|-1}$
- 在 T_2 上重复上一步过程，直至得到根节点

黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

交叉验证

- 用独立数据集测试子树序列$\{T_0, T_1, \cdots, T_n\}$中各棵子树的平方误差或基尼指数
- 平方误差或基尼指数最小的子树就是最优决策树
- 该决策树对应的α也确定为损失函数的α

黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

- 决策树模型与建立
- 特征选择
- 决策树的生成
- 决策树的剪枝
- 分类回归树（CART）
- 总结

黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

ID3, C4.5和CART比较

算法	支持模型	树结构	特征选择	是否支持连续数据处理	是否支持缺失值处理	剪枝
ID3	分类	多叉树	信息增益	不支持	不支持	不支持
C4.5	分类	多叉树	信息增益比 信息增益	支持	支持	支持
CART	分类、回归	二叉树	基尼指数 平方误差	支持	支持	支持

黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

决策树优点

- 决策树算法容易可视化，易理解，机理解释起来简单
- 决策树算法的时间复杂度较小
- 对缺失值不敏感
- 算法完全不受数据缩放的影响，不需要特征预处理，例如归一化或标准化

黄迟 计算机与人工智能学院
huangchi@swufe.edu.cn

决策树缺点

- 对连续性的字段比较难预测
- 即使做了剪枝，也经常会过拟合，泛化性能很差
- 忽略属性之间的相关性，在处理特征关联性比较强的数据时表现得不是太好