

人工智能之机器学习

决策树

上海育创网络科技有限公司

主讲人：刘老师(GerryLiu)

课程要求

- 课上课下 “九字” 真言
 - 认真听，**善摘录，勤思考**
 - **多温故，乐实践**，再发散
- 四不原则
 - **不懒散惰性，不迟到早退**
 - **不请假旷课，不拖延作业**
- 一点注意事项
 - 违反 “四不原则”，不推荐就业

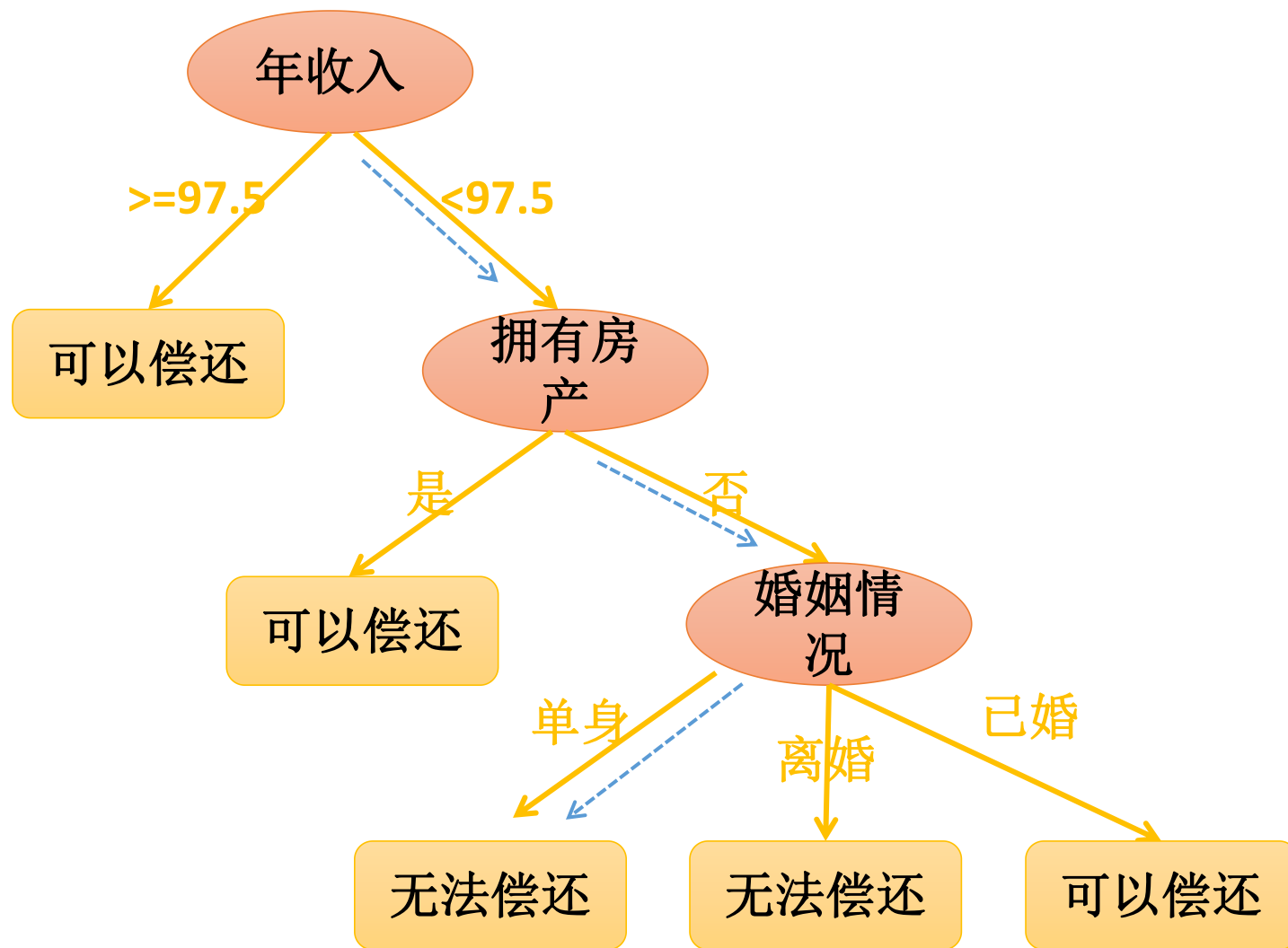
课程内容

- 信息熵
- 决策树
- 决策树优化
- 剪枝
- 决策树可视化

决策树直观理解

ID	拥有房产(是/否)	婚姻状态(单身\已婚\离婚)	年收入(单位:千元)	无法偿还债务(是/否)
1	是	单身	125	否
2	否	已婚	100	否
3	否	单身	100	否
4	是	已婚	110	否
5	是	离婚	60	否
6	否	离婚	95	是
7	否	单身	85	是
8	否	已婚	75	否
9	否	单身	90	是
10	是	离婚	220	否

决策树直观理解



- 当构建好一个判断模型后，新来一个用户后，可以根据构建好的模型直接进行判断，比如新用户特性为：**无房产**、**单身**、**年收入55K**，那么根据判断得出该用户无法进行债务偿还。这种决策对于借贷业务有比较好的指导意义。

比特化(Bits)

- 假设存在一组随机变量X，各个值出现的概率关系如图；
- 现在有一组由X变量组成的序列: BACADDCBAC.....; 如果现在希望将这个序列转换为二进制来进行网络传输，那么我们得到一个得到一个这样的序列:
01001000111110010010.....
- 结论: 在这种情况下，我们可以使用两个比特位来表示一个随机变量。

$P(X=A)=1/4$	$P(X=B)=1/4$	$P(X=C)=1/4$	$P(X=D)=1/4$
--------------	--------------	--------------	--------------

A	B	C	D
00	01	10	11

比特化(Bits)

- 而当X变量出现的概率值不一样的时候，对于一组序列信息来讲，每个变量平均需要多少个比特位来描述呢??

$P(X=A)=1/2$	$P(X=B)=1/4$	$P(X=C)=1/8$	$P(X=D)=1/8$
--------------	--------------	--------------	--------------

A	B	C	D
0	10	110	111

$$E = 1 * \frac{1}{2} + 2 * \frac{1}{4} + 3 * \frac{1}{8} + 3 * \frac{1}{8} = 1.75$$

$$E = -\log_2\left(\frac{1}{2}\right) * \frac{1}{2} - \log_2\left(\frac{1}{4}\right) * \frac{1}{4} - \log_2\left(\frac{1}{8}\right) * \frac{1}{8} - \log_2\left(\frac{1}{8}\right) * \frac{1}{8} = 1.75$$

一般化的比特化(Bits)

- 假设现在随机变量 X 具有 m 个值，分别为： V_1, V_2, \dots, V_m ；并且各个值出现的概率如下表所示；那么对于一组序列信息来讲，每个变量平均需要多少个比特位来描述呢？

$$P(X=V_1)=p_1$$

$$P(X=V_2)=p_2$$

$$P(X=V_3)=p_3$$

.....

$$P(X=V_m)=p_m$$

- 可以使用这些变量的期望来表示每个变量需要多少个比特位来描述信息：

$$E(X) = -p_1 \log_2(p_1) - p_2 \log_2(p_2) - \dots - p_m \log_2(p_m)$$

$$= -\sum_{i=1}^m p_i \log_2(p_i)$$

信息熵(Entropy)

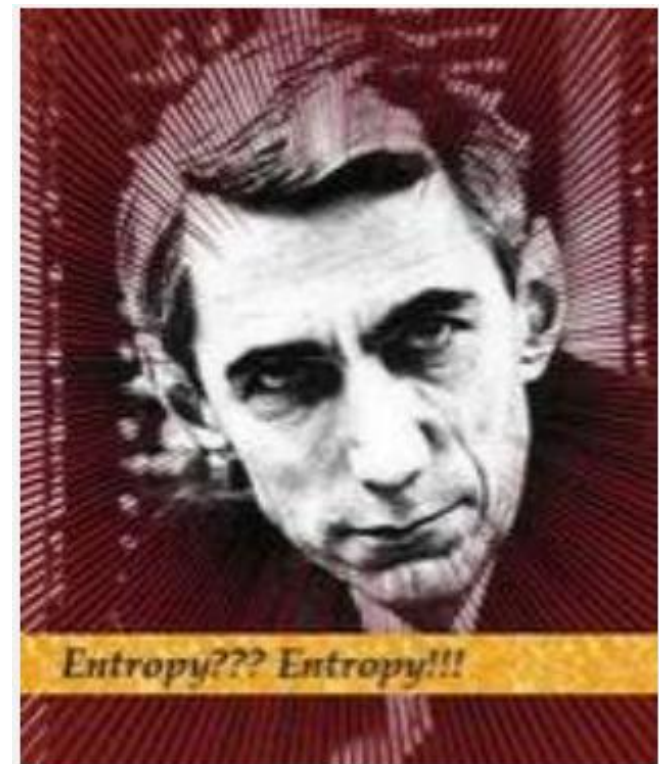
- $H(X)$ 就叫做随机变量 X 的信息熵;

$$H(X) = -\sum_{i=1}^m p_i \log_2(p_i)$$

信息熵(Entropy)

$$H(X) = - \sum_{i=1}^m p_i \log_2(p_i)$$

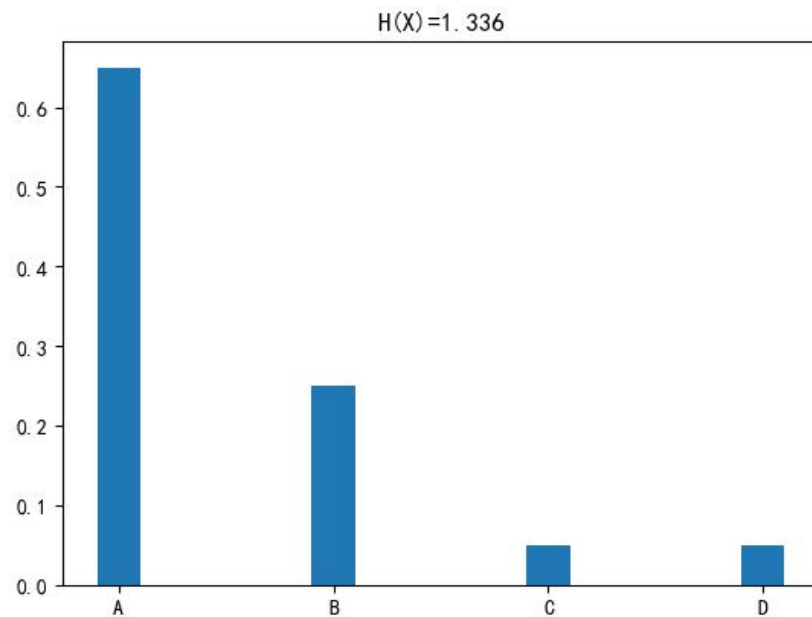
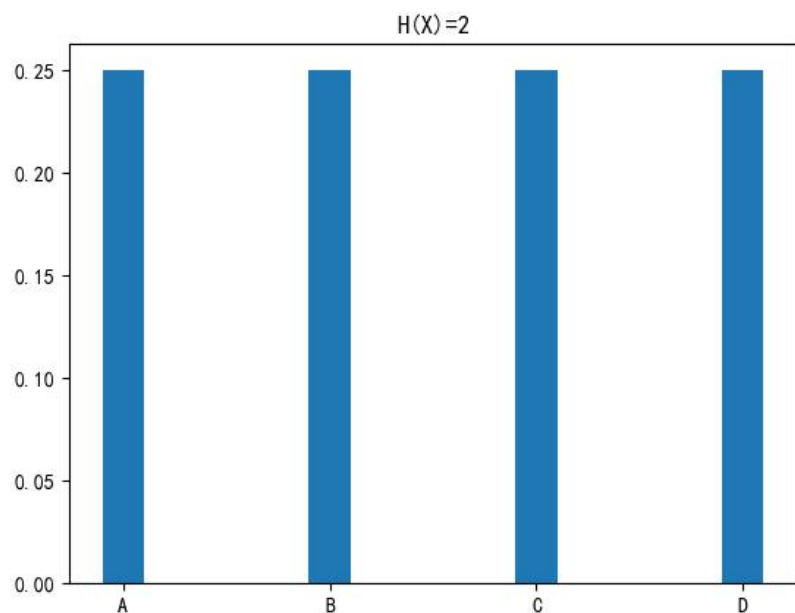
- 信息量：指的是一个样本/事件所蕴含的信息，如果一个事件的概率越大，那么就可以认为该事件所蕴含的信息越少。极端情况下，比如：“太阳从东方升起”，因为是确定事件，所以不携带任何信息量。
- 信息熵：1948年，香农引入信息熵；一个系统越是有序，信息熵就越低，一个系统越是混乱，信息熵就越高，所以信息熵被认为是一个系统有序程度的度量。
- **信息熵就是用来描述系统信息量的不确定度。**



信息熵(Entropy)

$$H(X) = -\sum_{i=1}^m p_i \log_2(p_i)$$

- High Entropy(高信息熵): 表示随机变量X是均匀分布的, 各种取值情况是等概率出现的。
- Low Entropy(低信息熵): 表示随机变量X各种取值不是等概率出现。可能出现有的事件概率很大, 有的事件概率很小。



信息熵(Entropy)案例

$$H(X) = -\sum_{i=1}^m p_i \log_2(p_i)$$

- 赌马比赛中，有两组赛马共八匹，获胜的概率如下：

	第一组	第二组
P(X=A)	1/4	13/20
P(X=B)	1/4	5/20
P(X=C)	1/4	1/20
P(X=D)	1/4	1/20

- 在比赛前，对于第一组而言，我们只知道A/B/C/D获胜的概率是一样的，我们是判断不出来任何偏向的；但是对于第二组而言，我们很清楚的就能够判断A会获胜。

条件熵 $H(Y|X)$

- 给定条件 X 的情况下，随机变量 Y 的信息熵就叫做条件熵。

专业(X)	性别(Y)
数学	M
IT	M
英语	F
数学	F
数学	M
IT	M
英语	F
数学	F

$$P(X = \text{数学}) = 0.5$$

$$P(Y = M) = 0.5$$

$$P(X = \text{数学}, Y = F) = 0.25$$

$$P(Y = M | X = \text{英语}) = 0$$

$$H(X) = 1.5$$

$$H(Y) = 1$$

条件熵 $H(Y|X)$

- 当专业(X)为数学的时候，Y的信息熵的值为： $H(Y|X=\text{数学})$

专业(X)	性别(Y)
数学	M
IT	M
英语	F
数学	F
数学	M
IT	M
英语	F
数学	F

专业(X)	性别(Y)
数学	M
数学	F
数学	M
数学	F

$$H(Y | X = \text{数学}) = 1$$

条件熵 $H(Y|X)$

- 给定条件 X 的情况下，所有不同 x 值情况下 Y 的信息熵的平均值叫做条件熵。

$$H(Y|X) = \sum_{j=1} P(X = v_j) H(Y|X = v_j)$$

专业(X)	性别(Y)
数学	M
IT	M
英语	F
数学	F
数学	M
IT	M
英语	F
数学	F

v_j	$P(X=v_j)$	$H(Y X=v_j)$
数学	0.5	1
IT	0.25	0
英语	0.25	0

$$H(Y|X) = 0.5 * 1 + 0.25 * 0 + 0.25 * 0 = 0.5$$

条件熵 $H(Y|X)$

- 给定条件 X 的情况下，所有不同 x 值情况下 Y 的信息熵的平均值叫做条件熵。另外一个公式如下所示：

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

- 事件 (X, Y) 发生所包含的熵，减去事件 X 单独发生的熵，即为在事件 X 发生的前提下， Y 发生“新”带来的熵，这个也就是条件熵本身的概念。

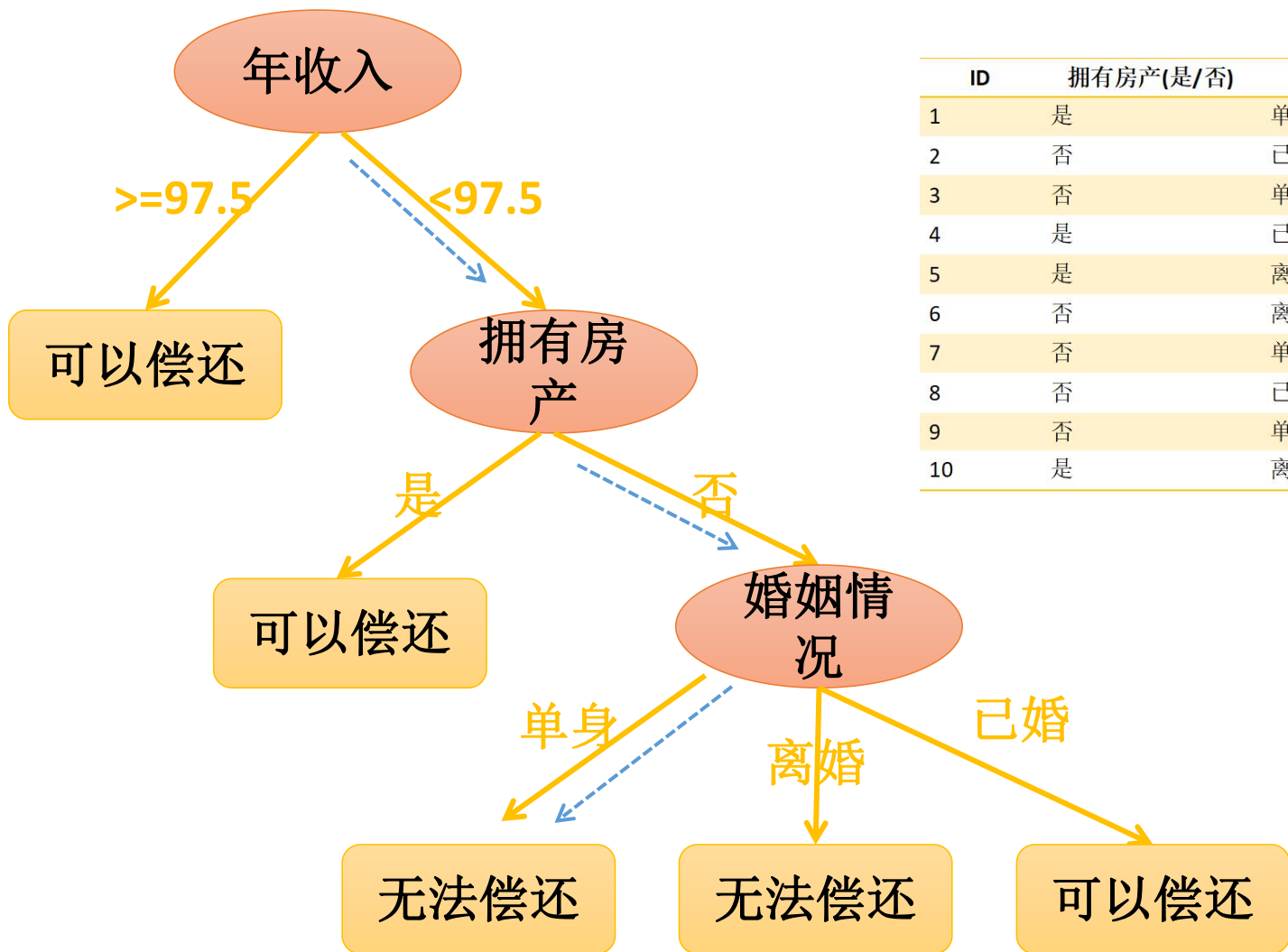
条件熵 $H(Y|X)$

$$\begin{aligned} H(Y|X) &= \sum_{j=1} P(X = v_j) H(Y|X = v_j) = \sum_x P(x) H(Y|x) \\ &= \sum_x p(x) \left(- \sum_y p(y|x) \log(p(y|x)) \right) = - \sum_x \sum_y p(x) p(y|x) \log(p(y|x)) \\ &= - \sum_x \sum_y p(x, y) \log \left(\frac{p(x, y)}{p(x)} \right) \\ &= - \sum_x \sum_y p(x, y) \log(p(x, y)) - \left[- \sum_x \left(\sum_y p(x, y) \right) \log(p(x)) \right] \\ &= H(X, Y) - \left[- \sum_x p(x) \log(p(x)) \right] = H(X, Y) - H(X) \end{aligned}$$

什么是决策树

- 决策树(Decision Tree)是在已知各种情况发生概率的基础上, 通过构建决策树来进行分析的一种方式, 是一种直观应用概率分析的一种图解法; 决策树是一种预测模型, 代表的是对象属性与对象值之间的映射关系; 决策树是一种树形结构, 其中每个内部节点表示一个属性的测试, 每个分支表示一个测试输出, 每个叶节点代表一种预测类别; 决策树是一种非常常用的有监督的分类算法。
- 决策树的决策过程就是从根节点开始, 测试待分类项中对应的特征属性, 并按照其值选择输出分支, 直到叶子节点, 将叶子节点的存放的类别作为决策结果。
- 决策树分为两大类: 分类树和回归树, 前者用于分类标签值, 后者用于预测连续值, 常用算法有ID3、C4.5、CART等

决策树构建



ID	拥有房产(是/否)	婚姻状态(单身\已婚\离婚)	年收入(单位:千元)	无法偿还债务(是/否)
1	是	单身	125	否
2	否	已婚	100	否
3	否	单身	100	否
4	是	已婚	110	否
5	是	离婚	60	否
6	否	离婚	95	是
7	否	单身	85	是
8	否	已婚	75	否
9	否	单身	90	是
10	是	离婚	220	否

决策树构建过程

- 决策树算法的重点就是决策树的构造；决策树的构造就是进行属性选择度量，确定各个特征属性之间的拓扑结构(树结构)；构建决策树的关键步骤就是分裂属性，分裂属性是指在某个节点按照某一类特征属性的不同划分构建不同的分支，其目标就是让各个分裂子集尽可能的'纯'(让一个分裂子数据集中待分类的项尽可能的属于同一个类别)。
- 构建步骤如下：
 - 1. 将所有特征看成一个一个的节点；
 - 2. 遍历当前特征的每一种分割方式，找到最好的分割点；将数据划分为不同的子节点，eg： N_1 、 N_2 N_m ；计算划分之后所有子节点的'纯度'信息；
 - 3. 使用第二步遍历所有特征，选择出最优的特征以及该特征的最优的划分方式；得出最终的子节点： N_1 、 N_2 N_m
 - 4. 对子节点 N_1 、 N_2 N_m 分别继续执行2-3步，直到每个最终的子节点都足够'纯'。

决策树特征属性类型

- 根据特征属性的类型不同，在构建决策树的时候，采用不同的方式，具体如下：
 - 属性是离散值，而且不要求生成的是二叉决策树，此时一个属性就是一个分支
 - 属性是离散值，而且要求生成的是二叉决策树，此时使用属性划分的子集进行测试，按照“属于此子集”和“不属于此子集”分成两个分支
 - 属性是连续值，可以确定一个值作为分裂点split_point，按照 $> \text{split_point}$ 和 $\leq \text{split_point}$ 生成两个分支

决策树分割属性选择

- 决策树算法是一种“贪心”算法策略，只考虑在当前数据特征情况下的最好分割方式，不能进行回溯操作。
- 对于整体的数据集而言，按照所有的特征属性进行划分操作，对所有划分操作的结果集的“纯度”进行比较，选择“纯度”越高的特征属性作为当前需要分割的数据集进行分割操作，持续迭代，直到得到最终结果。决策树是通过“纯度”来选择分割特征属性点的。

决策树量化纯度

- 决策树的构建是基于样本概率和纯度进行构建操作的，那么进行判断数据集是否“纯”可以通过三个公式进行判断，分别是Gini系数、熵(Entropy)、错误率，这三个公式值越大，表示数据越“不纯”；越小表示越“纯”；实践证明这三种公式效果差不多，一般情况使用熵公式

$P(1) = 7/10 = 0.7$; 可以偿还概率

$P(2) = 3/10 = 0.3$; 无法偿还概率

$$Gini = 1 - \sum_{i=1}^n P(i)^2 \quad H(Entropy) = - \sum_{i=1}^n P(i) \log_2(P(i)) \quad Error = 1 - \max_{i=1}^n \{P(i)\}$$

决策树量化纯度

- 当计算出各个特征属性的量化纯度值后使用**信息增益度**来选择出当前数据集的分割特征属性；如果信息增益度的值越大，表示在该特征属性上会损失的纯度越大，那么该属性就越应该在决策树的上层，计算公式为：

$$Gain = \Delta = H(D) - H(D|A)$$

- D目标属性，A为某一个待划分的特征属性；Gain为A为特征对训练数据集D的信息增益，它为集合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

决策树算法的停止条件

- 决策树构建的过程是一个递归的过程，所以必须给定停止条件，否则过程将不会进行停止，一般情况有两种停止条件：
 - 当每个子节点只有一种类型的时候停止构建
 - 当前节点中样本数小于某个阈值，同时迭代次数达到给定值时，停止构建过程，此时使用 $\max(p(i))$ 作为节点的对应类型
- NOTE：方式一可能会使树的节点过多，导致过拟合(Overfitting)等问题；比较常用的方式是使用方式二作为停止条件。

决策树算法效果评估

- 决策树的效果评估和一般的分类算法一样，采用混淆矩阵来进行计算准确率、召回率、精确率等指标
- 也可以采用叶子节点的不纯度值总和来评估算法的效果，值越小，效果越好

		predicted condition			
total population		prediction positive	prediction negative	Prevalence = $\frac{\sum \text{condition positive}}{\sum \text{total population}}$	
true condition	condition positive	True Positive (TP)	False Negative (FN) (type II error)	True Positive Rate (TPR), Sensitivity, Recall, Probability of Detection $= \frac{\sum TP}{\sum \text{condition positive}}$	False Negative Rate (FNR), Miss Rate $= \frac{\sum FN}{\sum \text{condition positive}}$
	condition negative	False Positive (FP) (Type I error)	True Negative (TN)	False Positive Rate (FPR), Fall-out, Probability of False Alarm $= \frac{\sum FP}{\sum \text{condition negative}}$	True Negative Rate (TNR), Specificity (SPC) $= \frac{\sum TN}{\sum \text{condition negative}}$
Accuracy $= \frac{\sum TP + \sum TN}{\sum \text{total population}}$		Positive Predictive Value (PPV), Precision = $\frac{\sum TP}{\sum \text{prediction positive}}$	False Omission Rate (FOR) $= \frac{\sum FN}{\sum \text{prediction negative}}$	Positive Likelihood Ratio (LR+) $= \frac{TPR}{FPR}$	Diagnostic Odds Ratio (DOR) $= \frac{LR+}{LR-}$
		False Discovery Rate (FDR) $= \frac{\sum FP}{\sum \text{prediction positive}}$	Negative Predictive Value (NPV) $= \frac{\sum TN}{\sum \text{prediction negative}}$	Negative Likelihood Ratio (LR-) $= \frac{FNR}{TNR}$	

$$C(T) = \sum_{t=1}^{leaf} \frac{|D_t|}{|D|} H(t)$$

决策树算法效果评估

- 决策树的损失函数(该值越小, 算法效果越好)

$$loss = \sum_{t=1}^{leaf} \frac{|D_t|}{|D|} H(t)$$

决策树直观理解结果计算

ID	拥有房产(是/否)	婚姻状态(单身\已婚\离婚)	年收入(单位:千元)	无法偿还债务(是/否)
1	是	单身	125	否
2	否	已婚	100	否
3	否	单身	100	否
4	是	已婚	110	否
5	是	离婚	60	否
6	否	离婚	95	是
7	否	单身	85	是
8	否	已婚	75	否
9	否	单身	90	是
10	是	离婚	220	否

$$H(Y | X) = \sum_{j=1} P(X = v_j) H(Y | X = v_j)$$

$$Info(D) = - \sum_{i=1}^2 P(i) \log_2(P(i)) = -0.7 * \log_2(0.7) - 0.3 \log_2(0.3) = 0.88$$

$$Info(D_{有房产}) = -4/4 * \log_2(4/4) - 0 * \log_2(0) = 0 \quad Info(D_{无房产}) = -0.5 \log_2(0.5) - 0.5 \log_2(0.5) = 1$$

$$Gain(房产) = Info(D) - \sum_{j=1}^2 \frac{N(D_j)}{N(D)} Info(D_j) = 0.88 - 0.4 * 0 - 0.6 * 1 = 0.28$$

$$Gain(婚姻) = 0.205 \quad Gain(收入 = 97.5) = 0.395$$



决策树直观理解结果计算

ID	拥有房产(是/否)	婚姻状态(单身\已婚\离婚)	年收入(单位:千元)	无法偿还债务(是/否)
1	是	单身	125	否
2	否	已婚	100	否
3	否	单身	100	否
4	是	已婚	110	否
5	是	离婚	60	否
6	否	离婚	95	是
7	否	单身	85	是
8	否	已婚	75	否
9	否	单身	90	是
10	是	离婚	220	否

$$p(x = \text{是}) = 0.4 \quad p(x = \text{否}) = 0.6$$

$$p(y = \text{是} \mid x = \text{是}) = 0 \quad p(y = \text{否} \mid x = \text{是}) = 1$$

$$p(y = \text{是} \mid x = \text{否}) = 0.5 \quad p(y = \text{否} \mid x = \text{否}) = 0.5$$

$$Info(D_{\text{有房产}}) = H(D_{\text{有房产}}) = -1 * \log_2(1) - 0 * \log_2(0) = 0$$

$$Info(D_{\text{无房产}}) = H(D_{\text{无房产}}) = -0.5 \log_2(0.5) - 0.5 \log_2(0.5) = 1$$

$$Gain(\text{房产}) = Info(D) - \sum_{j=1}^2 \frac{N(D_j)}{N(D)} Info(D_j) = 0.88 - 0.4 * 0 - 0.6 * 1 = 0.28$$

x拥有房产(是/否) y无法偿还债务(是/否)

是 否

否 否

否 否

是 否

是 否

否 是

否 是

否 否

否 是

是 否

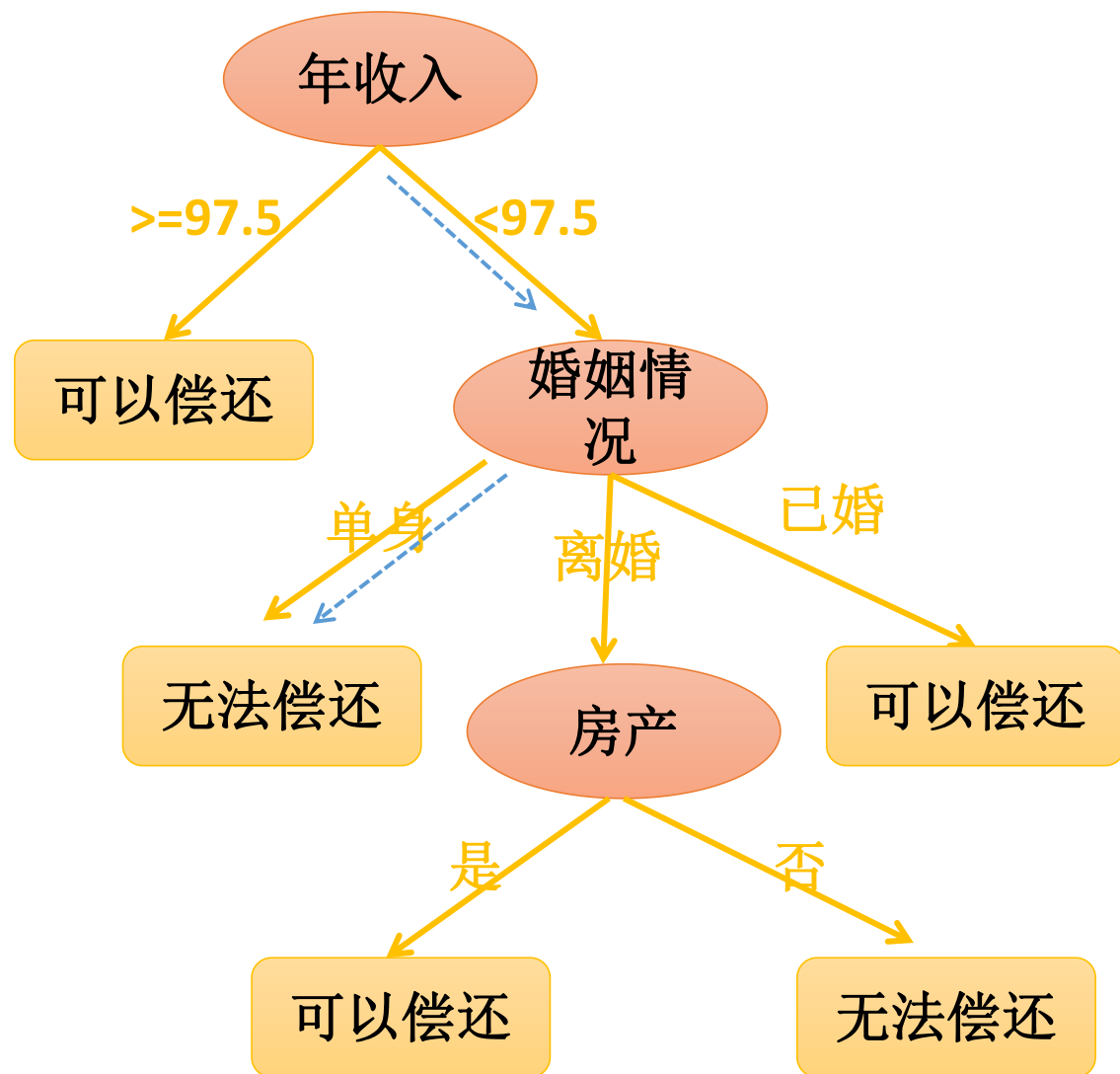
决策树直观理解结果计算

x(年收入)	60	75	85	90	95	100	100	110	125	220
y(无法偿还债务)	否	否	是	是	是	否	否	否	否	否
分割点	67.5	80	87.5	92.5	97.5	100	105	117.5	172.5	

$$Gain(\text{收入} = 97.5) = 0.395$$

$$Gain(\text{收入} = 80) = 0.116$$

决策树直观理解



- 重新构建决策树!!! (如果一个特征不允许重复选择)
- 当构建好一个判断模型后，新来一个用户后，可以根据构建好的模型直接进行判断，比如新用户特性为：**无房产**、**单身**、**年收入55K**，那么根据判断得出该用户无法进行债务偿还。这种决策对于借贷业务有比较好的指导意义。

