第五课　决策树分类

主讲人：丁兆云

dingzhaoyun1983@163.com　17607310865　https://github.com/zyding1983/datamining

数据挖掘

Data Mining

01 回顾

02 深入

03 实践

1 决策树回顾

训练集如右图所示：

根据训练集数据建立决策树，并判断顾客：

(青年，低收入，无游戏爱好，中等信用度)

是否有购买电脑的倾向

id	年龄	收入	爱好	信用	购买
1	青	高	否	中	否
2	青	高	否	优	否
3	中	高	否	中	是
4	老	中	否	中	是
5	老	低	是	中	是
6	老	低	是	优	否
7	中	低	是	优	是
8	青	中	否	中	否
9	青	低	是	中	是
10	老	中	是	中	是
11	青	中	是	优	是
12	中	中	否	优	是
13	中	高	是	中	是
14	老	中	否	优	否

1 决策树回顾

训练集如右图所示：

根据训练集数据建立决策树，并判断顾客：

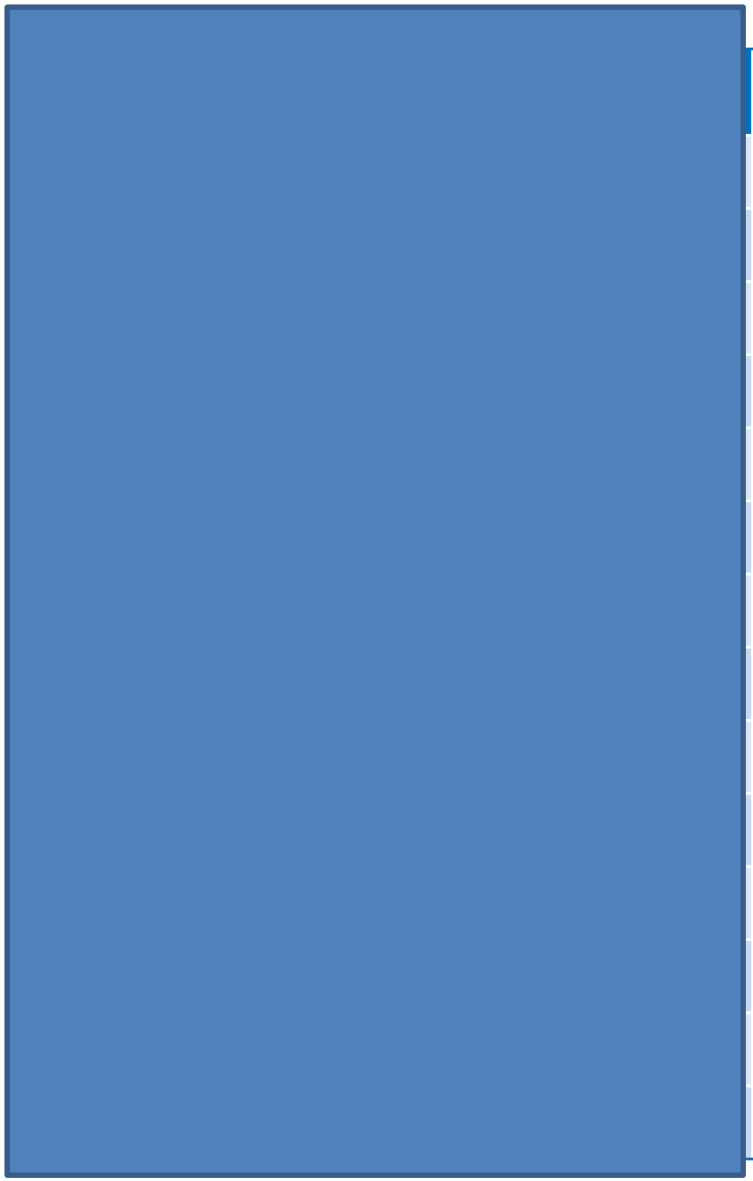
(青年，低收入，无游戏爱好，中等信用度)

是否有购买电脑的倾向

C0:9
C1:4

纯性小

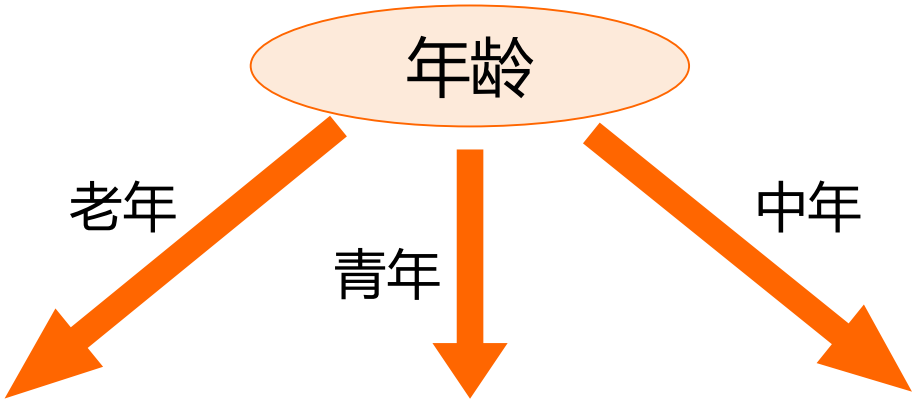
(不确定性大)



购买
否
否
是
是
是
否
是
否
是
是
是
是
是
否

1、假设以年龄为树的根节点

id	年龄	收入	爱好	信用	购买
1	青	高	否	中	否
2	青	高	否	优	否
3	中	高	否	中	是
4	老	中	否	中	是
5	老	低	是	中	是
6	老	低	是	优	否
7	中	低	是	优	是
8	青	中	否	中	否
9	青	低	是	中	是
10	老	中	是	中	是
11	青	中	是	优	是
12	中	中	否	优	是
13	中	高	是	中	是
14	老	中	否	优	否



id	收入	爱好	信用	购买
4	中	否	中	是
5	低	是	中	是
6	低	是	优	否
10	中	是	中	是
14	中	否	优	否

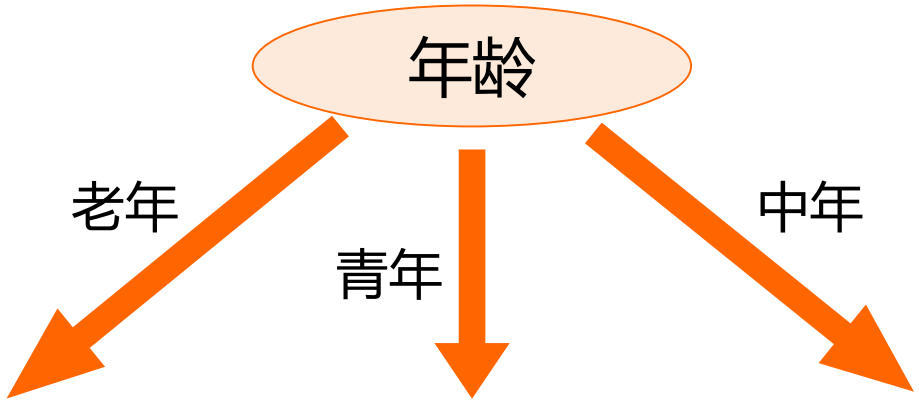
id	收入	爱好	信用	购买
1	高	否	中	否
2	高	否	优	否
8	中	否	中	否
9	低	是	中	是
11	中	是	优	是

id	收入	爱好	信用	购买
3	高	否	中	是
7	低	是	优	是
12	中	否	优	是
13	高	是	中	是

1、假设以年龄为树的根节点

C0:4
C1:0

纯性大



id	年龄	收入	爱好	信用	购买
1	青	高	否	中	否
2	青	高	否	优	否
3	中	高	否	中	是
4	老	中	否	中	是
5	老	低	是	中	是
6	老	低	是	优	否
7	中	低	是	优	是
8	青	中	否	中	否
9	青	低	是	中	是
10	老	中	是	中	是
11	青	中	是	优	是
12	中	中	否	优	是
13	中	高	是	中	是
14	老	中	否	优	否

id	收入	爱好	信用	购买
4	中	否	中	是
5	低	是	中	是
6	低	是	优	否
10	中	是	中	是
14	中	否	优	否

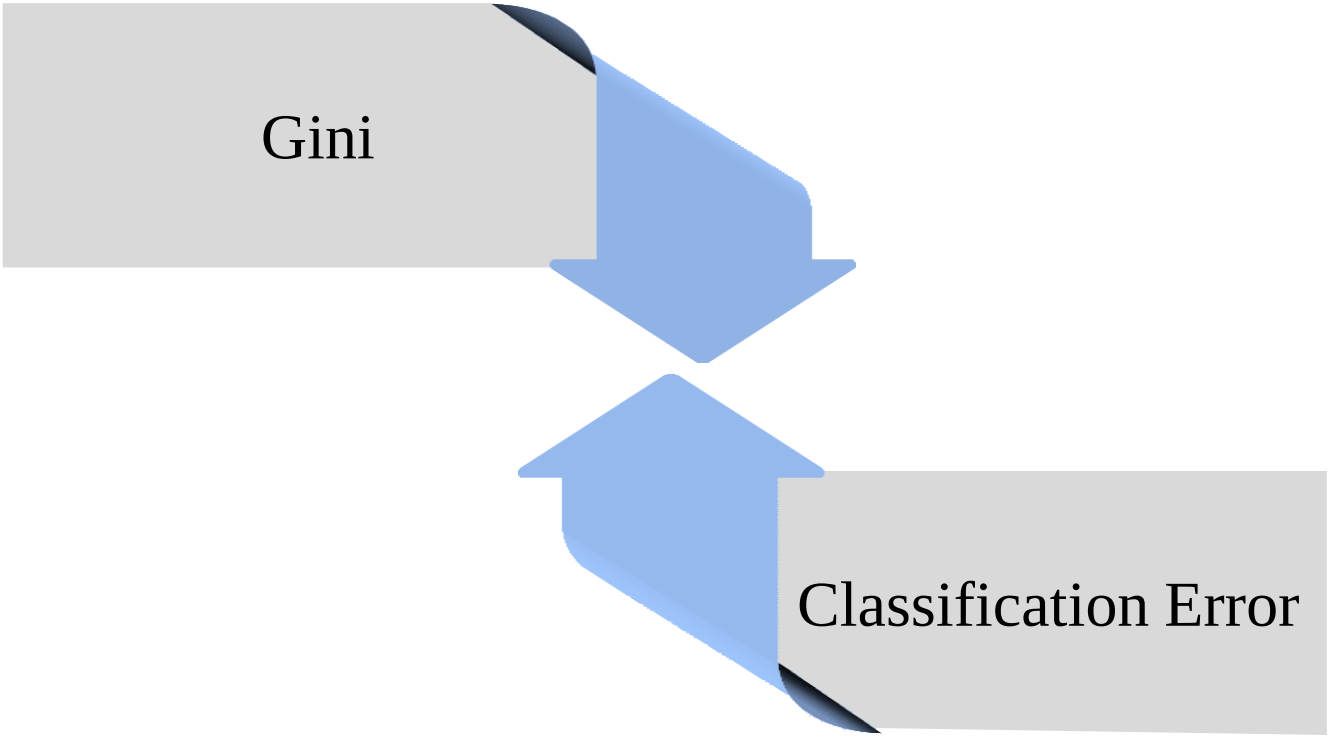
id	收入	爱好	信用	购买
1	高	否	中	否
2	高	否	优	否
8	中	否	中	否
9	低	是	中	是
11	中	是	优	是

id	收入	爱好	信用	购买
3	高	否	中	是
7	低	是	优	是
12	中	否	优	是
13	高	是	中	是

$$Entropy(S) = -\sum_{i=1}^{C} p_i \log(p_i)$$

不确定性

- 熵值越高，数据越混乱
- 熵值越低，数据越纯



数据挖掘

Data Mining

01 回顾

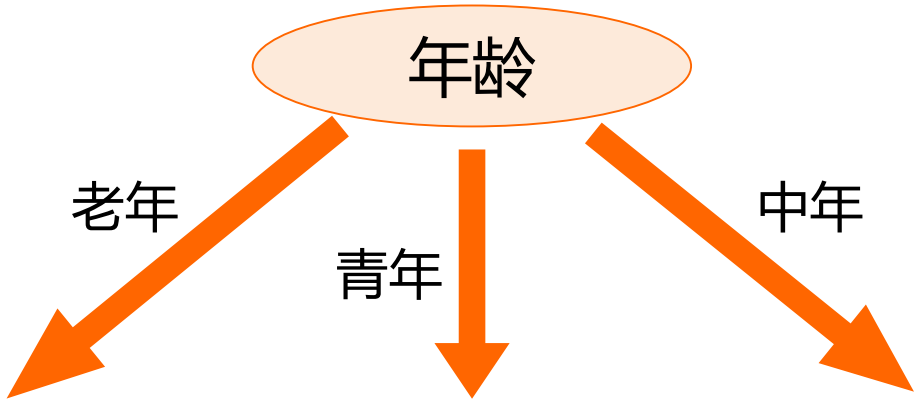
02 深入

03 实践

2.1 讨论一：属性分裂对ID3算法的影响

1、假设以年龄为树的根节点

$$Entropy(S) = -\sum_{i=1}^{C} p_i \log(p_i)$$



id	年龄	收入	爱好	信用	购买
1	青	高	否	中	否
2	青	高	否	优	否
3	中	高	否	中	是
4	老	中	否	中	是
5	老	低	是	中	是
6	老	低	是	优	否
7	中	低	是	优	是
8	青	中	否	中	否
9	青	低	是	中	是
10	老	中	是	中	是
11	青	中	是	优	是
12	中	中	否	优	是
13	中	高	是	中	是
14	老	中	否	优	否

id	收入	爱好	信用	购买
4	中	否	中	是
5	低	是	中	是
6	低	是	优	否
10	中	是	中	是
14	中	否	优	否

id	收入	爱好	信用	购买
1	高	否	中	否
2	高	否	优	否
8	中	否	中	否
9	低	是	中	是
11	中	是	优	是

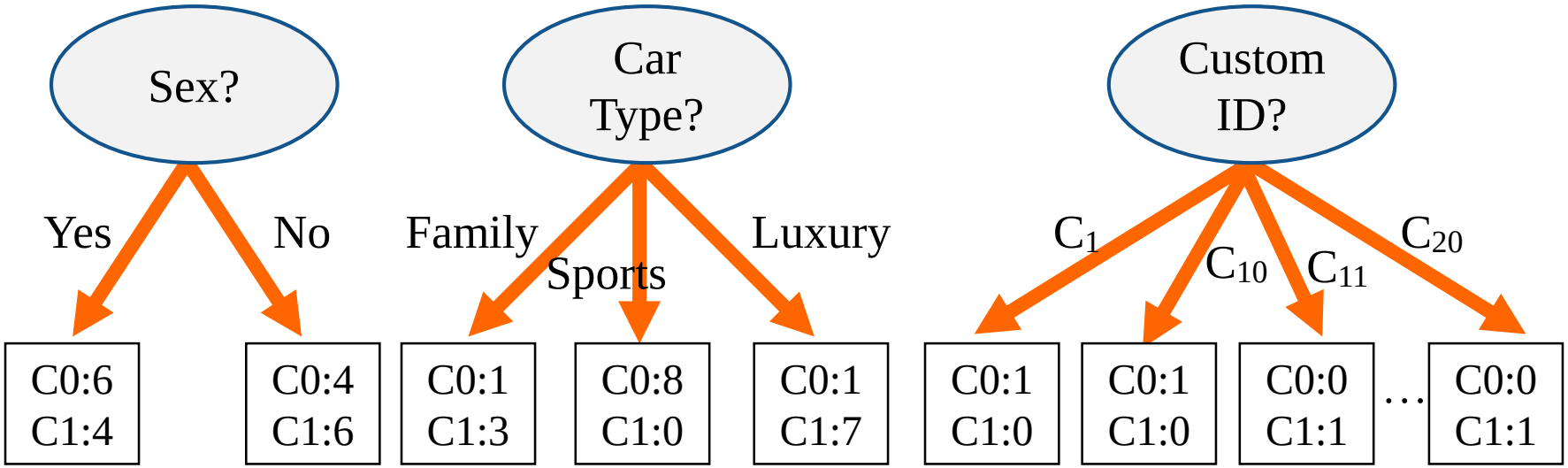
id	收入	爱好	信用	购买
3	高	否	中	是
7	低	是	优	是
12	中	否	优	是
13	高	是	中	是

2.1.1 思考，哪棵树子节点纯性最高?

在划分前：10 个记录 class 0,

10 个记录 class 1

$$Entropy(S) = -\sum_{i=1}^{C} p_i \log(p_i)$$



Entropy Bias

基于熵，会趋向于具有大量不同值的划分如：利用雇员id产生更纯的划分，但它却毫无用处。

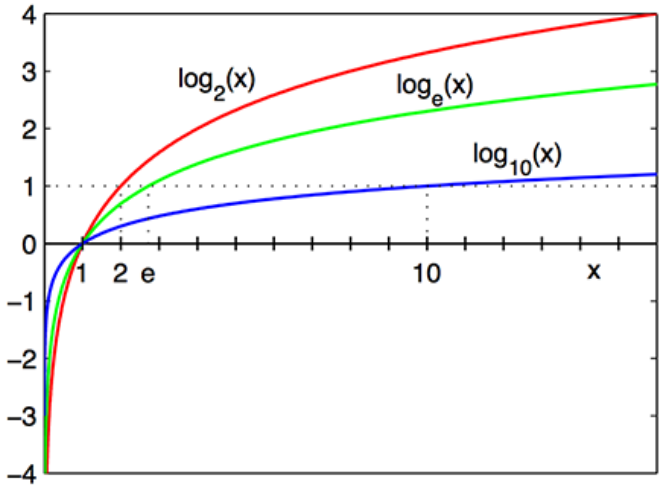
2.1.2 考虑增益率（Gain Ratio）C4.5算法

解决该问题的策略有两种：

- 限制测试条件只能是二元划分
- 使用增益率，K越大，SplitINFO越大，增益率被平衡。

id	年龄	收入	爱好	信用	购买
1	青	高	否	中	否
2	青	高	否	优	否
3	中	高	否	中	是
4	老	中	否	中	是
5	老	低	是	中	是
6	老	低	是	优	否
7	中	低	是	优	是
8	青	中	否	中	否
9	青	低	是	中	是
10	老	中	是	中	是
11	青	中	是	优	是
12	中	中	否	优	是
13	中	高	是	中	是
14	老	中	否	优	否

$$GainRATIO_{split} = \frac{GAIN_{Split}}{SplitINFO}$$



$$SplitINFO = -\sum_{i=1}^{k} \frac{n_i}{n} \log \frac{n_i}{n}$$

2.1.2 考虑增益率（Gain Ratio）C4.5算法

$$Gain(age) = 0.246$$

$$Gain(income) = 0.029$$

$$Gain(fancy) = 0.151$$

$$Gain(credit\_rating) = 0.048$$

$$GainRATIO_{split} = \frac{GAIN_{Split}}{SplitINFO}$$

$$SplitINFO = -\sum_{i=1}^{k} \frac{n_i}{n} \log \frac{n_i}{n}$$

id	年龄	收入	爱好	信用	购买
1	青	高	否	中	否
2	青	高	否	优	否
3	中	高	否	中	是
4	老	中	否	中	是
5	老	低	是	中	是
6	老	低	是	优	否
7	中	低	是	优	是
8	青	中	否	中	否
9	青	低	是	中	是
10	老	中	是	中	是
11	青	中	是	优	是
12	中	中	否	优	是
13	中	高	是	中	是
14	老	中	否	优	否

$$SplitInfo_{income}(D) = -\frac{4}{14} \times \log_2\left(\frac{4}{14}\right) - \frac{6}{14} \times \log_2\left(\frac{6}{14}\right) - \frac{4}{14} \times \log_2\left(\frac{4}{14}\right) = 1.557$$

gain_ratio(income) = 0.029/1.557 = 0.019

2.2 特征构造 单选题 1分

设置

决策树特征构造适合采用如下哪种方法

A 单调变换

B 线性组合

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

$$Entropy(S) = -\sum_{i=1}^{C} p_i \log(p_i)$$

作答

1	正			
2	正			
3	正			
4	正			
6	正			
5	负			
7	负			
8	负			
9	负			
10	负			

1	正	1		
2	正	2		
3	正	3		
4	正	6		
6	正	5		
5	负	7		
7	负	8		
8	负	9		
9	负	10		
10	负	11		

2.2 特征构造

1	正	1	2	
2	正	2	4	
3	正	3	6	
4	正	6	10	
6	正	5	11	
5	负	7	12	
7	负	8	15	
8	负	9	17	
9	负	10	19	
10	负	11	21	

2.3 总结

- 特点:
  - 决策树是一种构建分类模型的非参数方法
  - 不需要昂贵的的计算代价
  - 决策树相对容易解释
  - 决策树是学习离散值函数的典型代表
  - 决策数对于噪声的干扰具有相当好的鲁棒性
  - 冗余属性不会对决策树的准确率造成不利影响
  - 数据碎片问题：随着数的生长，可能导致叶结点记录数太少，**对于叶结点代表的类，不能做出具有统计意义的判决**
  - **子树可能在决策树中重复多次**，使决策树过于复杂
  - 决策树无法学习特征之间的线性关系：**特征构造**

数据挖掘

Data Mining

01 回顾

02 深入

03 实践

3 决策树分类编程实践

- from sklearn import tree
- from sklearn import svm
- from sklearn.naive_bayes import GaussianNB
- import pandas as pd
- data_url = "iris.csv"
- df = pd.read_csv(data_url)
- x = df.ix[:, 1:5]
- y = df.ix[:, 5]

- clf = GaussianNB()　【同学们自己思考修改该函数】
- clf = clf.fit(x, y)

- data_urltest = "iristest.csv"
- dftest = pd.read_csv(data_urltest)
- xtest = dftest.ix[:, 1:5]

- print(clf.predict(xtest))

https://scikit-learn.org/stable/modules/tree.html

https://scikit-learn.org/stable/modules/ensemble.html#forests-of-randomized-trees

Any Questions?

谢　谢！