决策树

陈飞宇
fchen@cqu.edu.cn
办公室：软件学院529

程序员的直觉（The intuition of Programmer）

- 数据集：

学号	数学	英语	语文	录取
1	A	C	B	Yes
2	A	B	B	Yes
3	A	B	C	No
4	B	B	B	Yes
5	B	C	B	No
6	C	C	B	No
7	C	A	A	Yes

程序员的直觉（The intuition of Programmer）

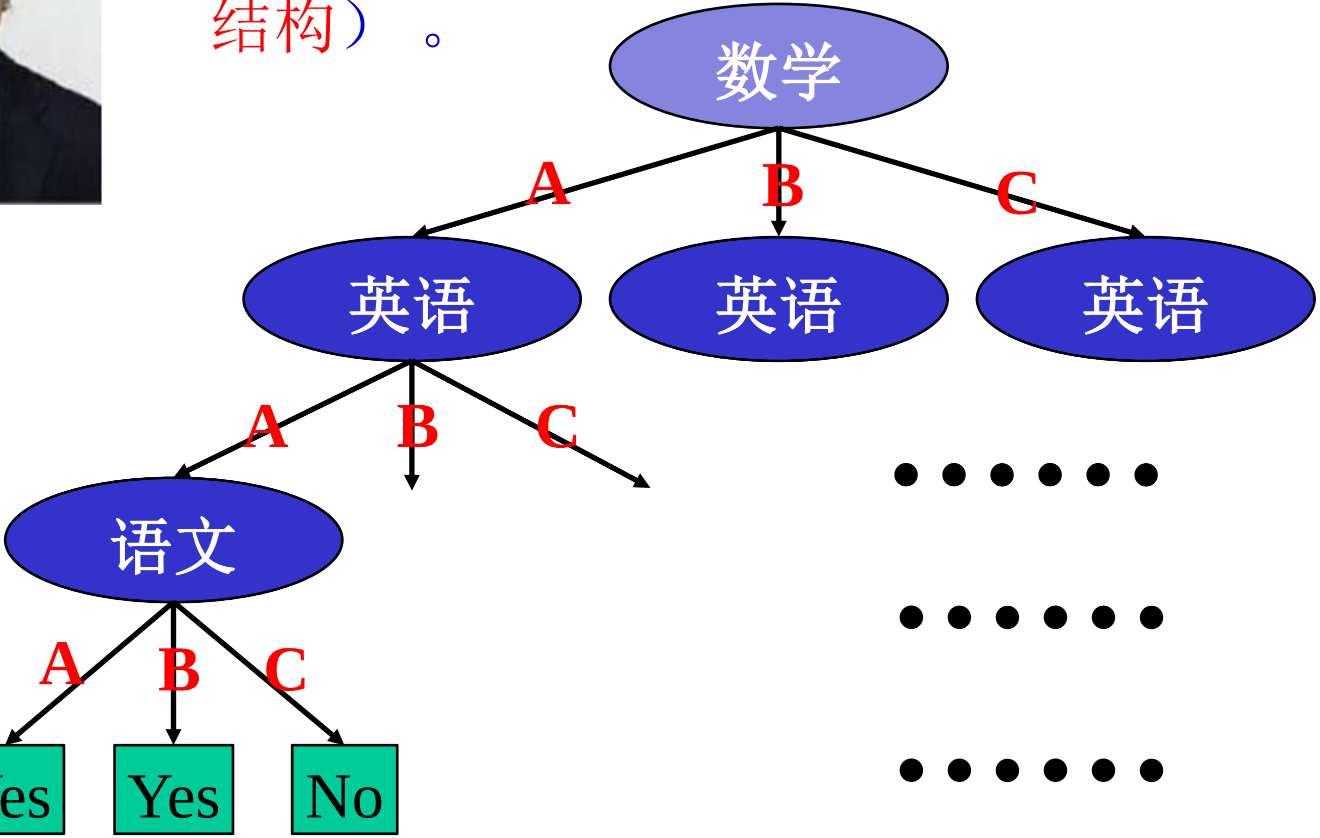
- 验证集：

学号	数学	英语	语文	录取
8	A	A	A	?
9	B	B	C	?
10	C	B	B	?
11	B	C	A	?
12	C	C	A	?
13	B	B	A	?

程序员的直觉

- 一般程序员：

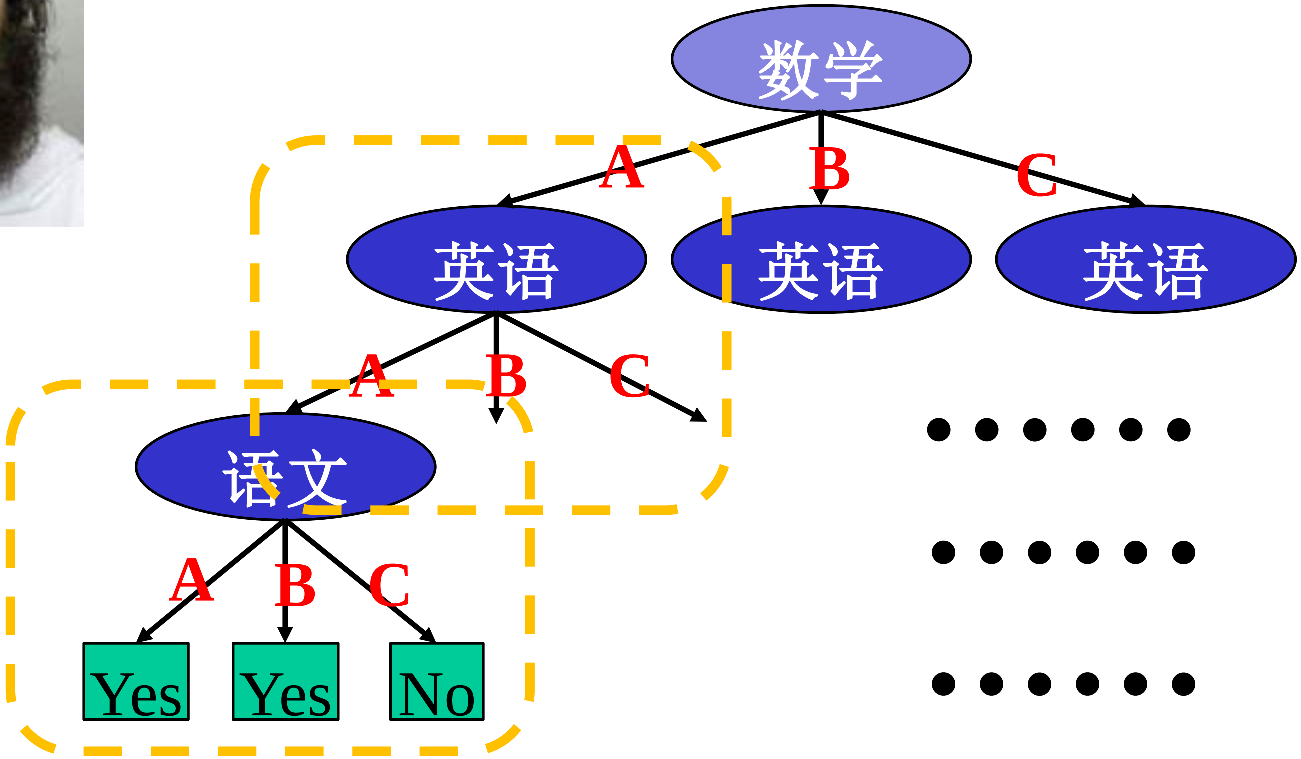
  - 绘制程序流程图，然后利用if else或switch case语句进行分支结构的程序实现（树形结构）。



程序员的直觉

- 资深程序员：

  – 发现规律，采用‘分而治之’思想，利用递归进行求解。

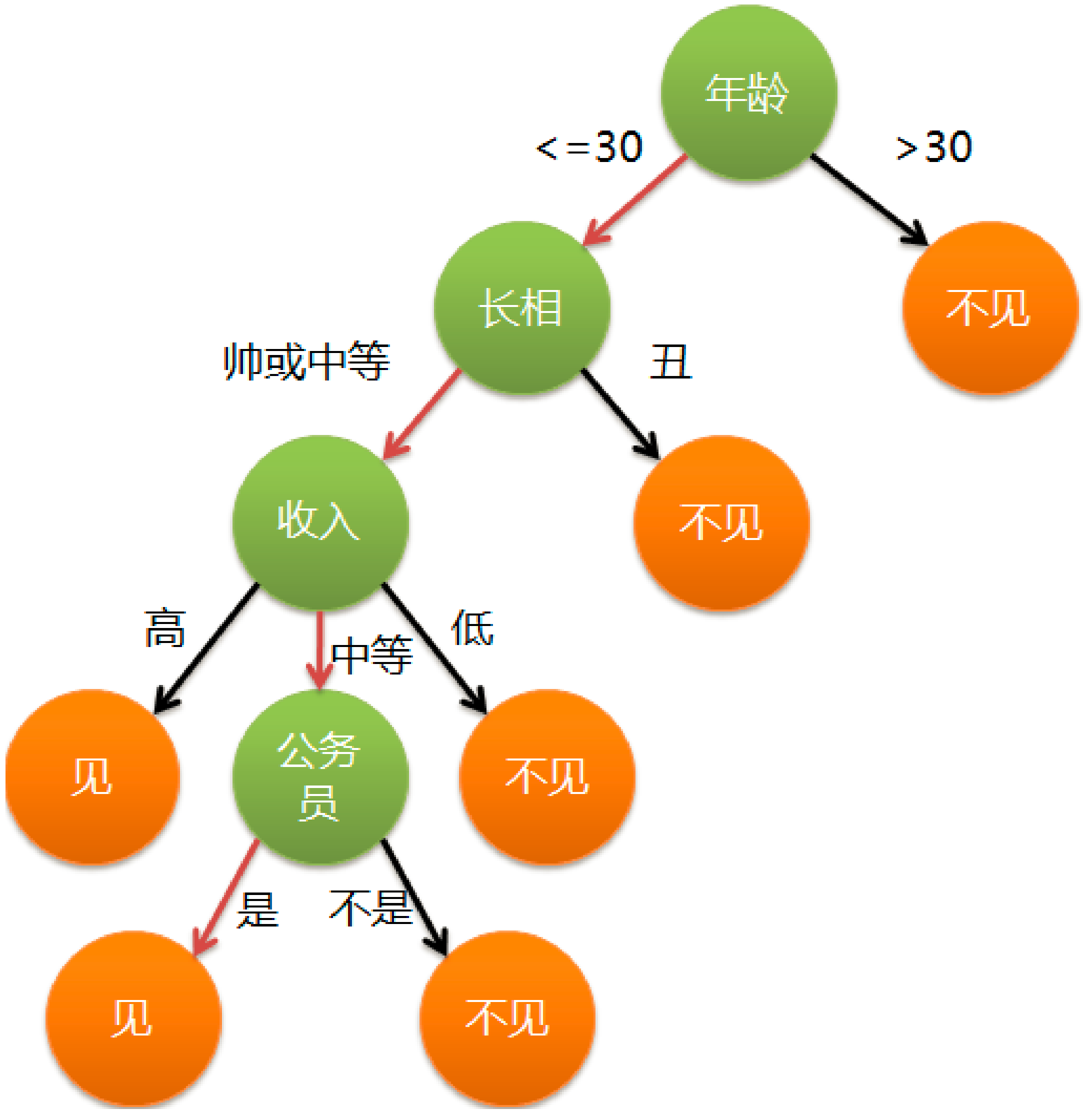


决策树-生活例子

- 相亲——母女对话：
  - 女儿：多大年纪了？
  - 母亲：26。
  - 女儿：长的帅不帅？
  - 母亲：挺帅的。
  - 女儿：收入高不？
  - 母亲：不算很高，中等情况。
  - 女儿：是公务员不？
  - 母亲：是，在税务局上班呢。
  - 女儿：那好，我去见见。

此例子纯属虚构，不代表广大女性同胞的择偶标准。如有雷同纯属巧合。

决策树



决策树（Decision Tree）

- 决策树（decision tree）：构建一个基于属性的树形分类器。
  - 每个非叶节点表示一个特征属性上的测试（分割），
  - 每个分支代表这个特征属性在某个值域上的输出，
  - 每个叶节点存放一个类别。
- 使用决策树进行决策的过程就是从根节点开始，测试待分类项中相应的特征属性，并按照其值选择输出分支，直到到达叶子节点，将叶子节点存放的类别作为决策结果。

决策树

- 决策树构建：分治法思想（递归）
  - 对于当前结点返回递归条件：
    ① 当前结点样本均属于同一类别，无需划分。
    ② 当前属性集为空。
    ③ 所有样本在当前属性集上取值相同，无法划分。
    ④ 当前结点包含的样本集合为空，不能划分。

决策树

- 递归结束条件
  1. 当前结点样本均属于同一类别，无需划分。
     - Example: 下一个要划分的属性为属性1

编号	属性1	类别
1	A	P
2	A	P
3	B	P
4	C	P

决策树

- 递归结束条件

  2. 当前属性集为空。

     - Example: 属性1(B)➔属性2(A)➔属性3(A) 走完该路径已经无属性往下分。

编号1	属性1	属性2	属性3	类别
1	A	C	A	P
2	B	A	A	P
3	B	B	B	N
4	C	C	B	N

决策树

- 递归结束条件

  3.所有样本在当前属性集上取值相同，无法划分。

  - Example: **属性1 B分支下**，样本子集中所有样本属性值完全一样，再往下划分就没有意义了。

编号1	属性1	属性2	属性3	类别
1	A	B	A	P
2	**B**	B	A	P
3	**B**	B	A	N
4	C	C	B	N

决策树

- 递归结束条件

  4.当前结点包含的样本集合为空，不能划分。

  - Example: **属性1 B分支中 属性2 A分支下**，唯一的属性——属性3，只有在值为A，其余情况样本集合为空。

编号1	属性1	属性2	属性3	类别
1	A	C	A	P
2	B	A	A	P
3	B	B	B	N
4	C	C	B	N

输入: 训练样本集$D=\{(x_1,y_1),(x_2,y_2),\cdots\cdots,(x_n,y_n)\}$;

属性集$A=\{a_1,a_2,\cdots\cdots,a_n\}$

函数 $TreeGenerate(D,A)$:

生成节点node

if D中样本全属于同一类别C:

  将node标记为C类叶节点; **return**;

end if

if 属性集A为空或者D的所有属性值均一样:

  将node标记为最多类; **return**;

end if

  从A中选取最佳划分属性a_*;

for a_*^v in a_*:

  为node生成一个分支，令D_v表示D中在a_*属性值为a_*^v的样本子集;

  if D_v为空:

   将分支结点标记为叶结点，其类别标签为D中样本最多的类; **return**;

  else:

   以$TreeGenerate(D_v,A\backslash\{a_*\})$为分支结点;

  end if

end for

决策树的核心

- 如何选取最佳划分属性：
  - 极端例子：

编号	属性1	属性2	属性3	标签
1	是	是	是	正
2	否	是	否	负
3	否	是	是	正
4	是	是	否	负
5	是	否	是	正
6	是	否	否	负
7	否	否	是	正

决策树的核心

- 定义最佳划分属性：
  – 经过属性划分后，不同类样本被更好的分离。
  – 理想情况：划分后样本被完美分类。即每个分支的样本都属性同一类。
  – 实际情况：不可能完美划分！尽量使得每个分支某一类样本比例尽量高！即尽量提高划分后子集的纯度（purity）。
- 最佳划分属性目标：
  – 提升划分后子集的纯度
  – 降低划分后子集的不纯度

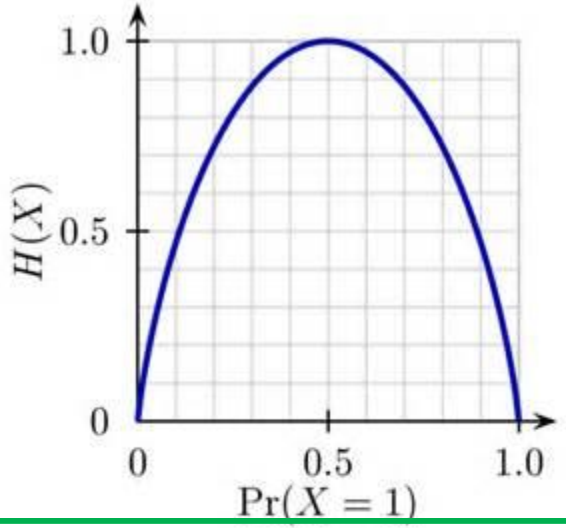
信息熵（Information Entropy）

信息熵（Information Entropy）：

$$\mathrm{Ent}(D) = -\sum_{k=1}^{m} p_k \log_2 p_k$$

其中 p_k 是集合 D 中第 k 类样本所占的比例。

信息熵越小，信息量越大，不确定性越小，样本纯度越高。



明天是星期二。

明天会下雨。

信息增益（Information Gain）

假设属性a有V可能取值$\{a^1, a^2, \cdots\cdots, a^V\}$, a^v对应划分后的数据子集为D^v.

信息增益（Information Gain）：

$$\mathrm{Gain}(D,a) = \mathrm{Ent}(D) - \sum_{v=1}^{V} \frac{|D^v|}{|D|} \mathrm{Ent}(D^v)$$

$\mathrm{Gain}(D,a)$越大，意味着使用属性来划分所获得的纯度提升越大。

ID3（Iterative Dichotomiser 3）

ID3算法（Quinlan, 1986）

基本思想：使用信息增益为准则来选择划分属性

$$a_* = \underset{a \in A}{\arg\max}\, \mathrm{Gain}(D, a)$$

$0 \log_2 0 = 0$	$\log_2 3 = 1.5850$	$\log_2 5 = 2.3219$
$\log_2 7 = 2.8074$	$\log_2 11 = 3.4594$	$\log_2 13 = 3.7004$
$\log_2 17 = 4.0875$	$\log_2 19 = 4.2479$	$\log_2 23 = 4.5236$

决策树（Decision Tree）

Computer Sale 实例

No.	age	income	student	credit	Buyer
1	<30	high	no	fair	no
2	<30	high	no	excellent	no
3	30~40	high	no	fair	yes
4	>40	medium	no	fair	yes
5	>40	low	yes	fair	yes
6	>40	low	yes	excellent	no
7	30~40	low	yes	excellent	yes
8	<30	medium	no	fair	no
9	<30	low	yes	fair	yes
10	>40	medium	yes	fair	yes
11	<30	medium	yes	excellent	yes
12	30~40	medium	no	excellent	yes
13	30~40	high	yes	fair	yes
14	>40	medium	no	excellent	no

ID3（计算信息熵）

No.	Buyer
1	no
2	no
3	yes
4	yes
5	yes
6	no
7	yes
8	no
9	yes
10	yes
11	yes
12	yes
13	yes
14	no

Class 1: Buyer = “yes” $\Longrightarrow$ $p_1 = \frac{9}{14}$

Class 2: Buyer = “no” $\Longrightarrow$ $p_2 = \frac{5}{14}$

信息熵（Information Entropy）：

$$\mathrm{Ent}(D) = -\sum_{k=1}^{m} p_k \log_2 p_k$$

$$\mathrm{Ent}(D) = -\left(\frac{9}{14}\log_2\frac{9}{14} + \frac{5}{14}\log_2\frac{5}{14}\right) = 0.9403$$

信息熵（属性 age）

No.	age	Buyer
1	<30	no
2	<30	no
3	30~40	yes
4	>40	yes
5	>40	yes
6	>40	no
7	30~40	yes
8	<30	no
9	<30	yes
10	>40	yes
11	<30	yes
12	30~40	yes
13	30~40	yes
14	>40	no

- **Subset 1: < 30.** $p_1 = \frac{2}{5}$ $p_2 = \frac{3}{5}$

$$\mathrm{Ent}(D^1) = -\left(\frac{2}{5}\log_2\frac{2}{5} + \frac{3}{5}\log_2\frac{3}{5}\right) = 0.9710$$

- **Subset 2: 30~40.** $p_1 = \frac{4}{4}$ $p_2 = \frac{0}{4}$

$$\mathrm{Ent}(D^2) = -\left(\frac{4}{4}\log_2\frac{4}{4} + \frac{0}{4}\log_2\frac{0}{4}\right) = 0$$

- **Subset 3: > 40.** $p_1 = \frac{2}{5}$ $p_2 = \frac{3}{5}$

$$\mathrm{Ent}(D^3) = -\left(\frac{2}{5}\log_2\frac{2}{5} + \frac{3}{5}\log_2\frac{3}{5}\right) = 0.9710$$

信息增益（属性age）

No.	age	Buyer
1	<30	no
2	<30	no
3	30~40	yes
4	>40	yes
5	>40	yes
6	>40	no
7	30~40	yes
8	<30	no
9	<30	yes
10	>40	yes
11	<30	yes
12	30~40	yes
13	30~40	yes
14	>40	no

- **Subset 1:** $\mathbf{Ent}(\boldsymbol{D}^{1}) = \mathbf{0.9710}$
- **Subset 2:** $\mathbf{Ent}(\boldsymbol{D}^{2}) = \mathbf{0}$
- **Subset 3:** $\mathbf{Ent}(\boldsymbol{D}^{3}) = \mathbf{0.9710}$

信息增益**(Information Gain):**

$$\mathrm{Gain}(D,a) = \mathrm{Ent}(D) - \sum_{v=1}^{V} \frac{|D^v|}{|D|}\mathrm{Ent}(D^v)$$

$$\begin{aligned} &\mathbf{Gain}(\boldsymbol{D},\mathbf{age}) \\ &= \mathbf{0.9403} - \left(\frac{\mathbf{5}}{\mathbf{14}} \times \mathbf{0.971} + \frac{\mathbf{4}}{\mathbf{14}} \times \mathbf{0} + \frac{\mathbf{5}}{\mathbf{14}} \times \mathbf{0.971}\right) \\ &= \mathbf{0.2467} \end{aligned}$$

信息熵（属性 income）

No.	income	Buyer
1	high	no
2	high	no
3	high	yes
4	medium	yes
5	low	yes
6	low	no
7	low	yes
8	medium	no
9	low	yes
10	medium	yes
11	medium	yes
12	medium	yes
13	high	yes
14	medium	no

- **Subset 1: high.** $p_1 = \frac{2}{4}$ $p_2 = \frac{2}{4}$

$$\text{Ent}(D^1) = -\left(\frac{2}{4}\log_2\frac{2}{4} + \frac{2}{4}\log_2\frac{2}{4}\right) = 1$$

- **Subset 2: medium.** $p_1 = \frac{4}{6}$ $p_2 = \frac{2}{6}$

$$\text{Ent}(D^2) = -\left(\frac{4}{6}\log_2\frac{4}{6} + \frac{2}{6}\log_2\frac{2}{6}\right) = 0.9183$$

- **Subset 3: low.** $p_1 = \frac{3}{4}$ $p_2 = \frac{1}{4}$

$$\text{Ent}(D^3) = -\left(\frac{3}{4}\log_2\frac{3}{4} + \frac{1}{4}\log_2\frac{1}{4}\right) = 0.8113$$

信息增益（属性income）

No.	income	Buyer
1	high	no
2	high	no
3	high	yes
4	medium	yes
5	low	yes
6	low	no
7	low	yes
8	medium	no
9	low	yes
10	medium	yes
11	medium	yes
12	medium	yes
13	high	yes
14	medium	no

- **Subset 1:** $\mathrm{Ent}(D^1) = 1$
- **Subset 2:** $\mathrm{Ent}(D^2) = 0.9183$
- **Subset 3:** $\mathrm{Ent}(D^3) = 0.8113$

信息增益(Information Gain):

$$\mathrm{Gain}(D, \mathrm{income})$$

$$= 0.9403 - \left(\frac{4}{14} \times 1 + \frac{6}{14} \times 0.9183 + \frac{4}{14} \times 0.8113\right)$$

$$= 0.0291$$

信息增益（属性student）

No.	student	Buyer
1	no	no
2	no	no
3	no	yes
4	no	yes
5	yes	yes
6	yes	no
7	yes	yes
8	no	no
9	yes	yes
10	yes	yes
11	yes	yes
12	no	yes
13	yes	yes
14	no	no

- **Subset 1: yes.** $p_1 = \frac{6}{7}$ $p_2 = \frac{1}{7}$

$$\mathrm{Ent}(D^1) = -\left(\frac{1}{7}\log_2\frac{1}{7} + \frac{6}{7}\log_2\frac{6}{7}\right) = 0.5917$$

- **Subset 2: no.** $p_1 = \frac{3}{7}$ $p_2 = \frac{4}{7}$

$$\mathrm{Ent}(D^2) = -\left(\frac{3}{7}\log_2\frac{3}{7} + \frac{4}{7}\log_2\frac{4}{7}\right) = 0.9852$$

信息增益(Information Gain):

$$\begin{aligned}&\mathrm{Gain}(D, \mathrm{student})\\&= 0.9403 - \left(\frac{7}{14} \times 0.5917 + \frac{7}{14} \times 0.9852\right)\\&= 0.1519\end{aligned}$$

信息增益（属性credit）

No.	credit	Buyer
1	fair	no
2	excellent	no
3	fair	yes
4	fair	yes
5	fair	yes
6	excellent	no
7	excellent	yes
8	fair	no
9	fair	yes
10	fair	yes
11	excellent	yes
12	excellent	yes
13	fair	yes
14	excellent	no

- **Subset 1: fair.** $p_1 = \frac{6}{8}$ $p_2 = \frac{2}{8}$

$$\text{Ent}(D^1) = -\left(\frac{6}{8}\log_2\frac{6}{8} + \frac{2}{8}\log_2\frac{2}{8}\right) = 0.8113$$

- **Subset 2: excellent.** $p_1 = \frac{3}{6}$ $p_2 = \frac{3}{6}$

$$\text{Ent}(D^2) = -\left(\frac{3}{6}\log_2\frac{3}{6} + \frac{3}{6}\log_2\frac{3}{6}\right) = 1$$

信息增益(Information Gain):

$$\begin{aligned}&\text{Gain}(D, \text{credit}) \\ &= 0.9403 - \left(\frac{8}{14} \times 0.8113 + \frac{6}{14} \times 1\right) \\ &= 0.0481\end{aligned}$$

最佳划分属性

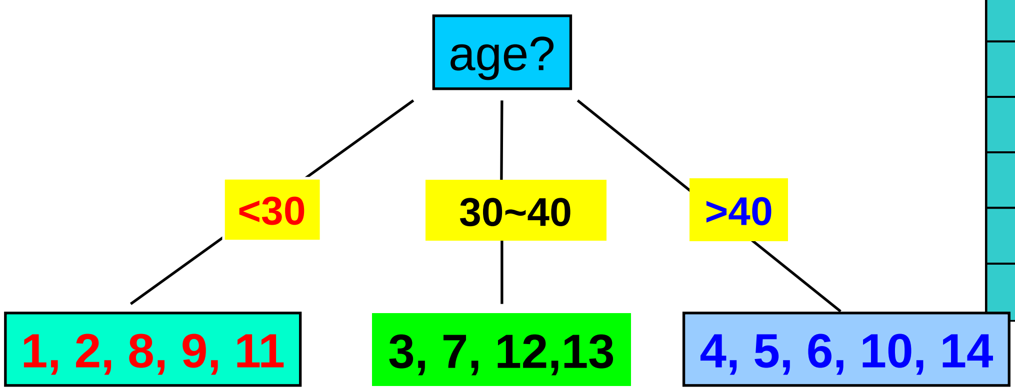
$\text{Gain}(D, \text{age}) = 0.2467$

$\text{Gain}(D, \text{income}) = 0.0291$

$\text{Gain}(D, \text{student}) = 0.1519$

$\text{Gain}(D, \text{credit}) = 0.0481$

No.	age	Buyer
1	<30	no
2	<30	no
3	30~40	yes
4	>40	yes
5	>40	yes
6	>40	no
7	30~40	yes
8	<30	no
9	<30	yes
10	>40	yes
11	<30	yes
12	30~40	yes
13	30~40	yes
14	>40	no



直观理解：纯度

age	Buyer
<30	no
<30	no
30~40	yes
>40	yes
>40	yes
>40	no
30~40	yes
<30	no
<30	yes
>40	yes
<30	yes
30~40	yes
30~40	yes
>40	no

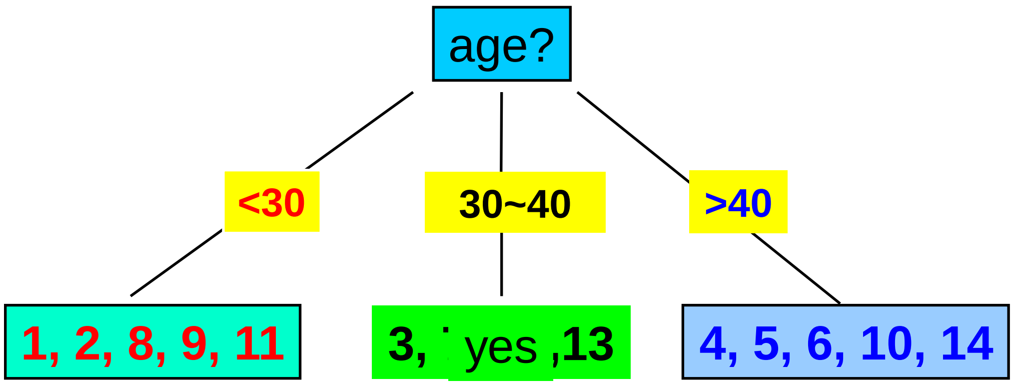
income	Buyer
high	no
high	no
high	yes
medium	yes
low	yes
low	no
low	yes
medium	no
low	yes
medium	yes
medium	yes
medium	yes
high	yes
medium	no

student	Buyer
no	no
no	no
no	yes
no	yes
yes	yes
yes	no
yes	yes
no	no
yes	yes
yes	yes
yes	yes
no	yes
yes	yes
no	no

credit	Buyer
fair	no
excellent	no
fair	yes
fair	yes
fair	yes
excellent	no
excellent	yes
fair	no
fair	yes
fair	yes
excellent	yes
excellent	yes
fair	yes
excellent	no

纯度最高

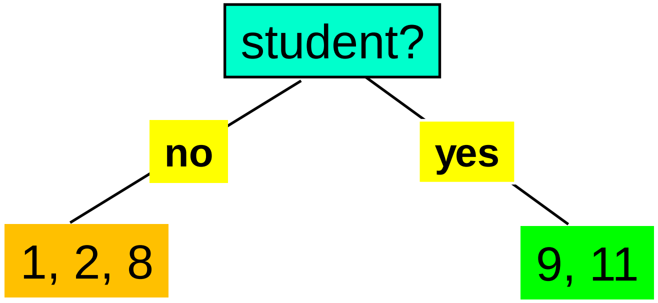
判断递归条件



- <30,样本仍然有两类，不符合所有递归返回条件，仍然可分，递归继续。
- 30~40， 样本类别均为Yes，满足递归返回条件1， 设为标签为yes的叶节点。
- >40, 样本仍然有两类，不符合所有递归返回条件, 仍然可分，递归继续。

子集（1, 2, 8, 9, 11）

No.	income	student	credit	Buyer
1	high	no	fair	no
2	high	no	excellent	no
8	medium	no	fair	no
9	low	yes	fair	yes
11	medium	yes	excellent	yes



Ent(D) = 0.9710

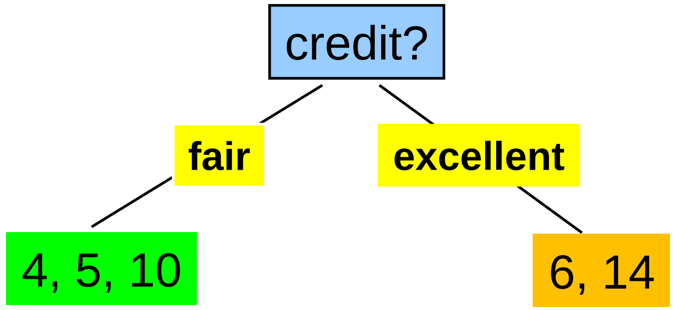
Gain(D, income) = 0.9710 - 0.4 = 0.5710

Gain(D, student) = 0.9710 – 0 = 0.9710

Gain(D, credit) = 0.9710 - 0.9510 = 0.02

子集（4, 5, 6, 10, 14）

No.	income	student	credit	Buyer
4	**medium**	**no**	**fair**	**yes**
5	**low**	**yes**	**fair**	**yes**
6	**low**	**yes**	**excellent**	**no**
10	**medium**	**yes**	**fair**	**yes**
14	**medium**	**no**	**excellent**	**no**

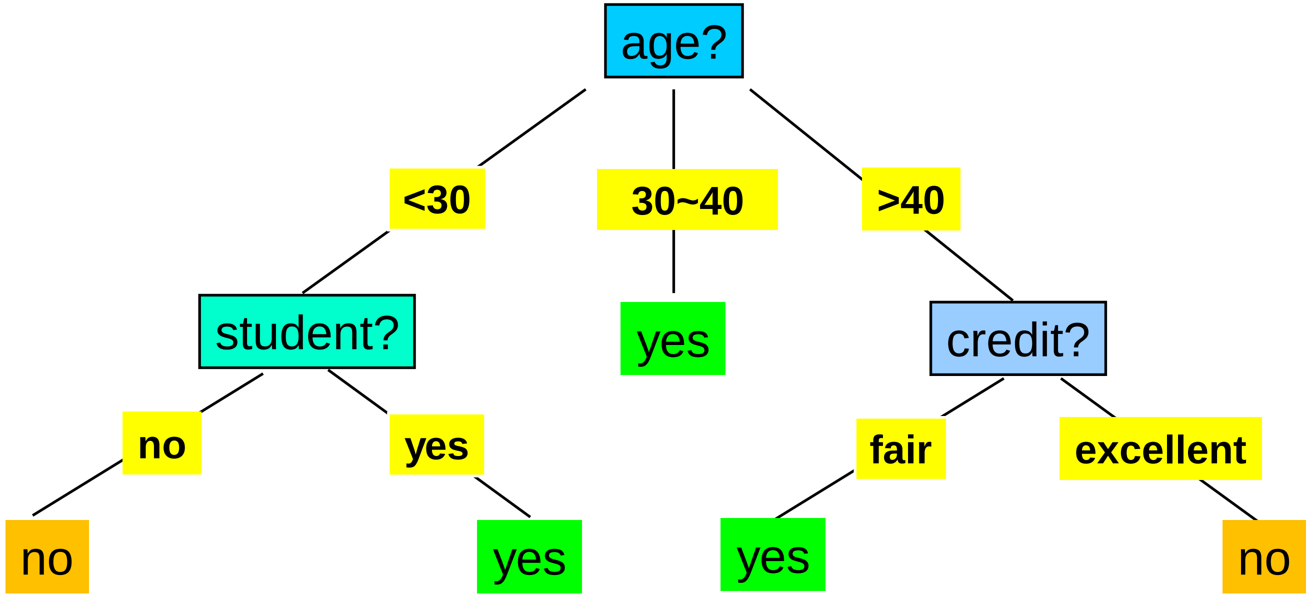


Gain(D, income) = 0.9710 – 0.9510 = 0.02

Gain(D, student) = 0.9710 – 0.9510 = 0.02

Gain(D, credit) = 0.9710 – 0 = 0.9710

信息熵（Information Entropy）



此决策树的最终形式是由数据决定，可能并非完美。比如10岁的小学生，35岁的盲人，50岁的软件学院教授。

例子1

回归开头的例子，动手绘制它的决策树。

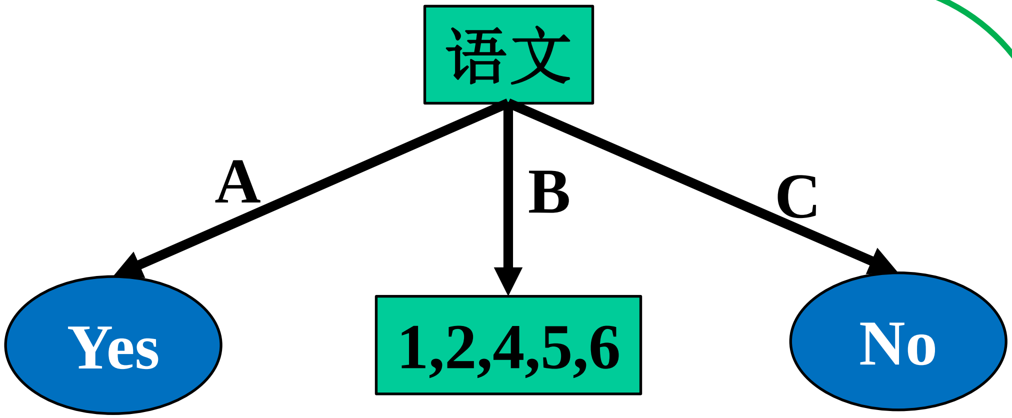
学号	数学	英语	语文	录取
1	A	C	B	Yes
2	A	B	B	Yes
3	A	B	C	No
4	B	B	B	Yes
5	B	C	B	No
6	C	C	B	No
7	C	A	A	Yes

第一层

Gain(数学) = 0.0202

Gain(英语) = 0.1981

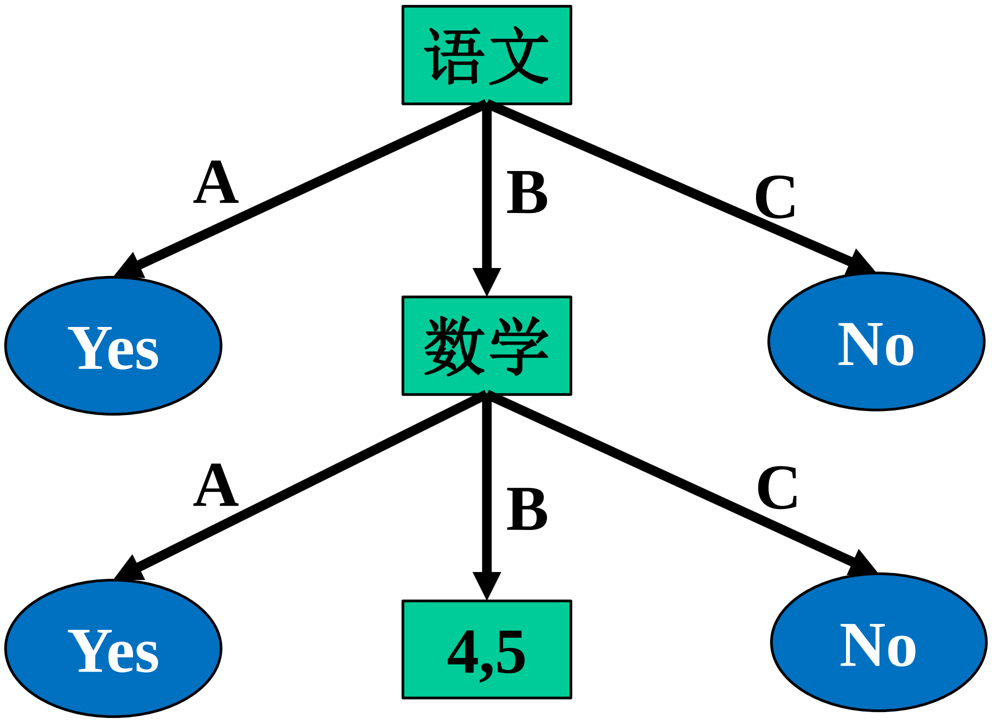
Gain(语文) = 0.2917



学号	数学	英语	语文	录取
1	A	C	B	Yes
2	A	B	B	Yes
4	B	B	B	Yes
5	B	C	B	No
6	C	C	B	No

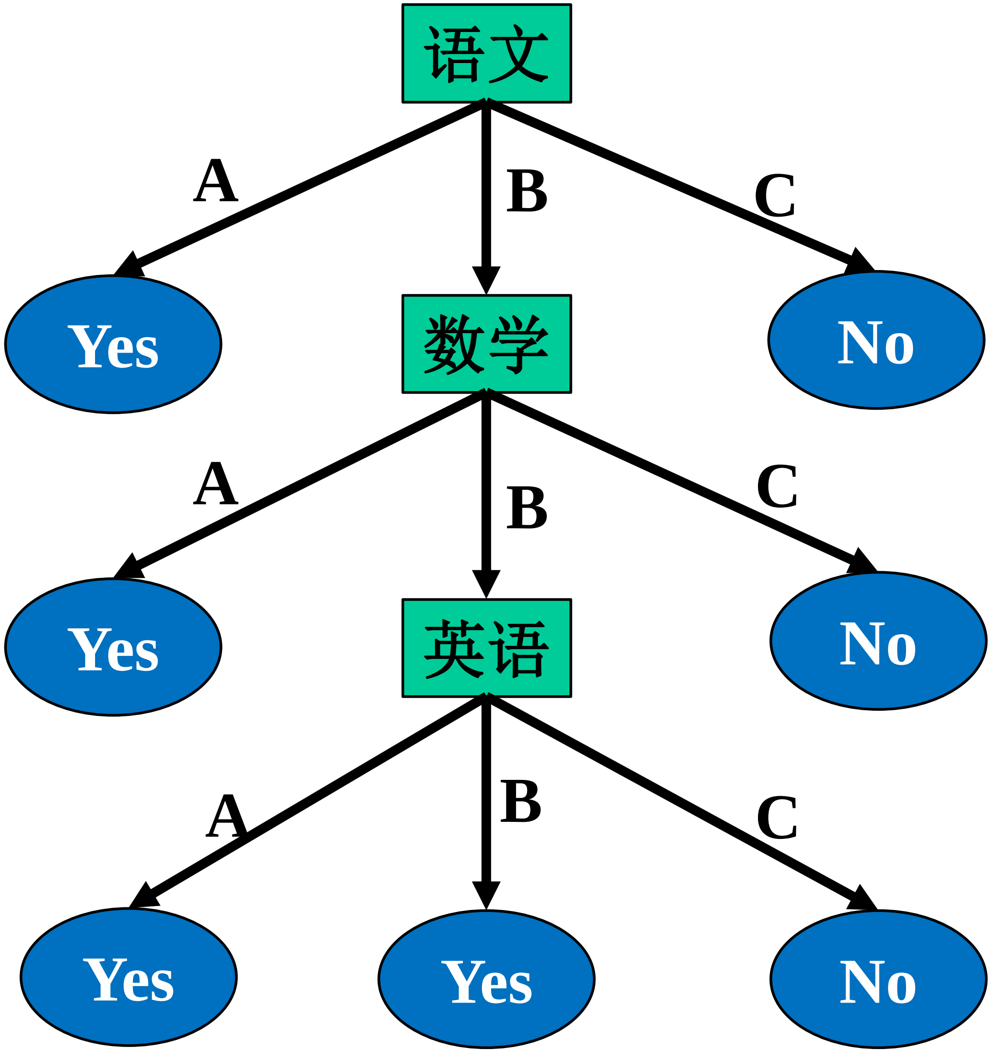
第二层

Gain(数学)=0.5710　　Gain(英语)=0.4200



学号	数学	英语	语文	录取
4	B	B	B	Yes
5	B	C	B	No

第三层



预测？

学号	数学	英语	语文	录取
8	A	A	A	Yes
9	B	B	C	No
10	C	B	B	No
11	B	C	A	Yes
12	C	C	A	Yes
13	B	B	A	Yes

信息增益（属性No.?）

No.	Buyer
1	no
2	no
3	yes
4	yes
5	yes
6	no
7	yes
8	no
9	yes
10	yes
11	yes
12	yes
13	yes
14	no

学号	录取
1	Yes
2	Yes
3	No
4	Yes
5	No
6	No
7	Yes

$\text{Gain}(D, \text{No}) = 0.9403$

$\text{Gain}(D, \text{学号}) = 0.9852$

ID3算法的缺陷

信息增量准则对可取值数目较多的属性有所偏好
。

- 考虑学号为一个属性
  ① **Gain(数学)=0.0202**
  ② **Gain(英语)=0.1981**
  ③ **Gain(语文)=0.2917**
  ④ **Gain(学号)=0.9852**

- **每个学号因为只有一个样本，纯度都很高！**

C4.5算法

判断准则：增益率（Gain Ratio）

$$\text{Gain\_ratio}(D,a)=\frac{\text{Gain}(D,a)}{IV(a)}$$

其中$IV(a)$称为属性a的“固有值”（Intrinsic Value）

$$IV(a)=-\sum_{v=1}^{V}\frac{|D^v|}{|D|}\log_2\frac{|D^v|}{|D|}$$

Gain_ratio(数学) = 0.0130　　**Gain_ratio(英语) = 0.1367**

Gain_ratio(语文) = 0.2539　　**Gain_ratio(学号) = 0.3509**

CART算法

CART（Classification And Regression Tree）

判断准则：基尼指数（Gini Index）：

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

基尼值反映了从数据集中随机抽取两个样本，其类别标记不一致的概率。

$$\mathrm{Gini\_index(D)} = \sum_{v=1}^{|V|} \frac{|D^v|}{|D|} \mathrm{Gini}(D)$$

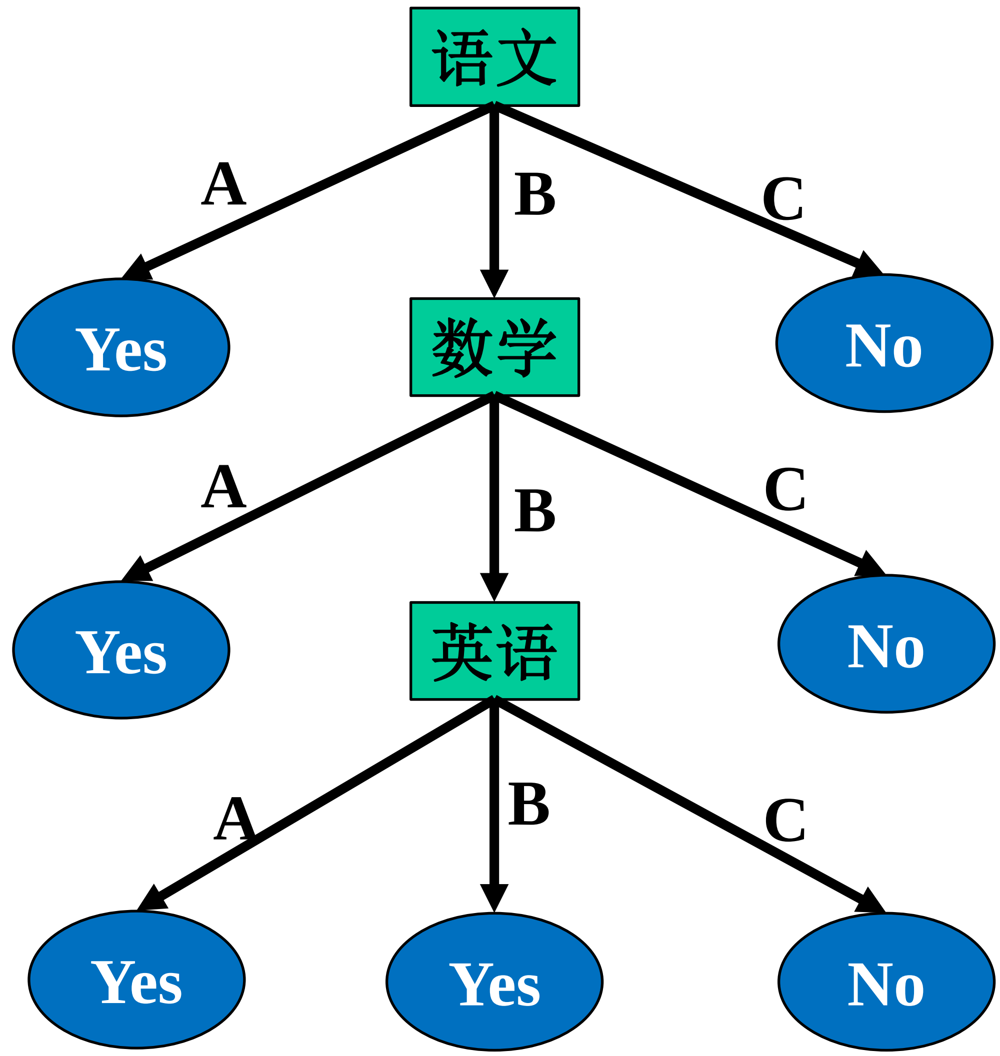
剪枝（Pruning）

- 剪枝(Pruning)处理——避免训练过拟合。
  - 预剪枝(prepruning)
    - 预剪枝是指在决策树生成过程中，对每个结点在划分前后进行估计，若当前结点划分不能提升决策树泛化性能，则进行裁剪，把结点标记为叶结点。
  - 后剪枝(postpruning)
    - 后剪枝是在生成一颗完整的决策树后，对非叶结点自底向上地对非叶结点进行考察，若将该结点对应的子树被替换为叶节点能提升决策树泛化能力，则进行裁剪。

例子1:数据集

学号	数学	英语	语文	录取
1	A	C	B	Yes
2	A	B	B	Yes
3	A	B	C	No
4	B	B	B	Yes
5	B	C	B	No
6	C	C	B	No
7	C	A	A	Yes

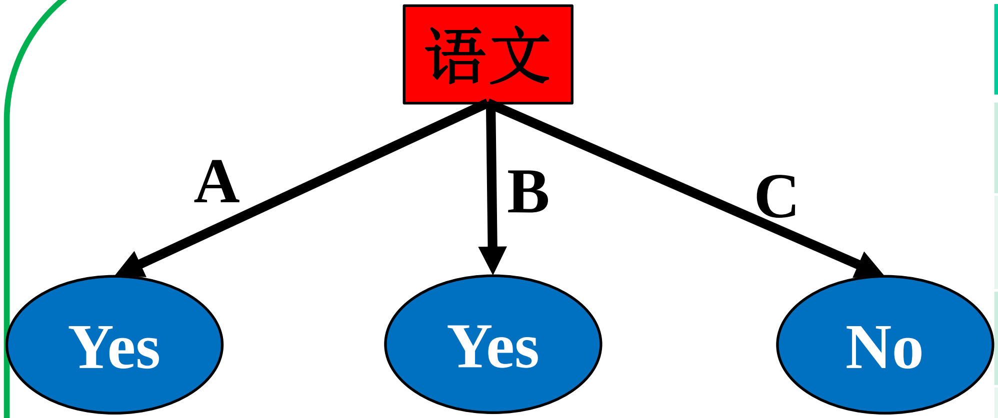
例子1：决策树



例子1：验证集

学号	数学	英语	语文	录取
8	A	A	A	Yes
9	B	B	C	No
10	C	B	B	Yes
11	B	C	A	No
12	C	C	A	No
13	B	B	A	Yes

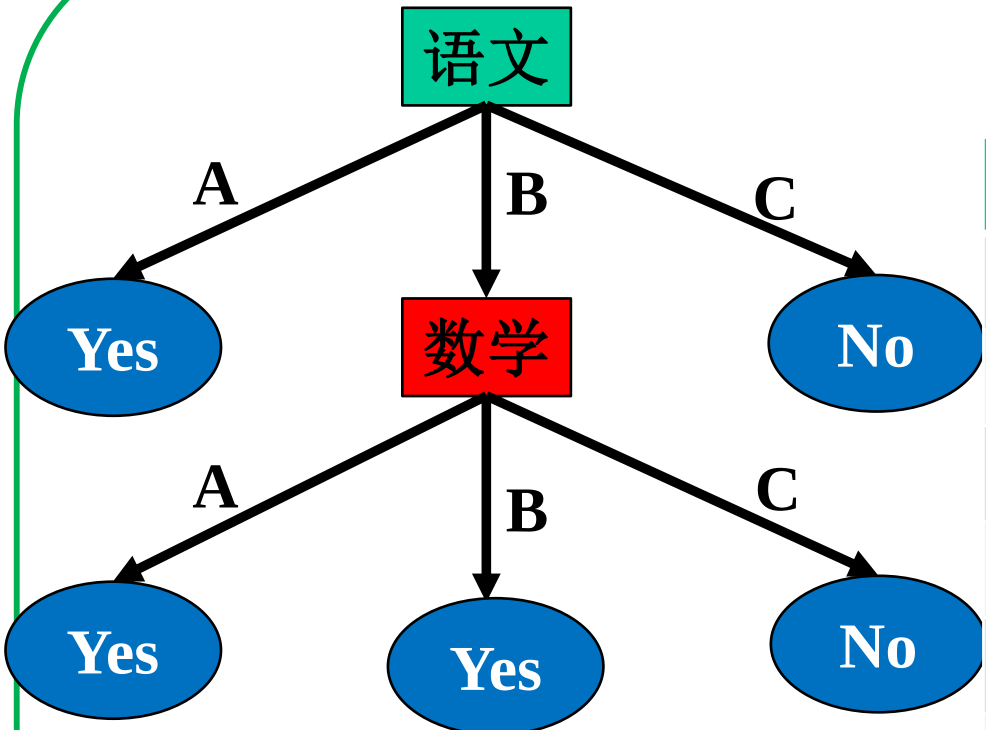
预剪枝(Prepruning)



真实值	划分前	划分后
Yes	Yes	Yes
No	Yes	No
Yes	Yes	Yes
No	Yes	Yes
No	Yes	Yes
Yes	Yes	Yes

划分前：50%　划分后：66.66%
预剪枝决策：划分

预剪枝（Prepruning）



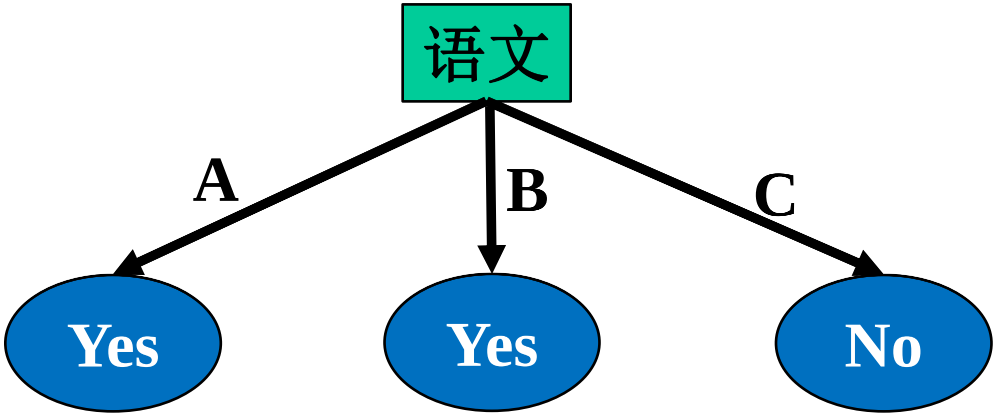
真实值	划分前	划分后
Yes	Yes	Yes
No	No	No
Yes	Yes	No
No	Yes	Yes
No	Yes	Yes
Yes	Yes	Yes

划分前：66.66%　划分后：50%

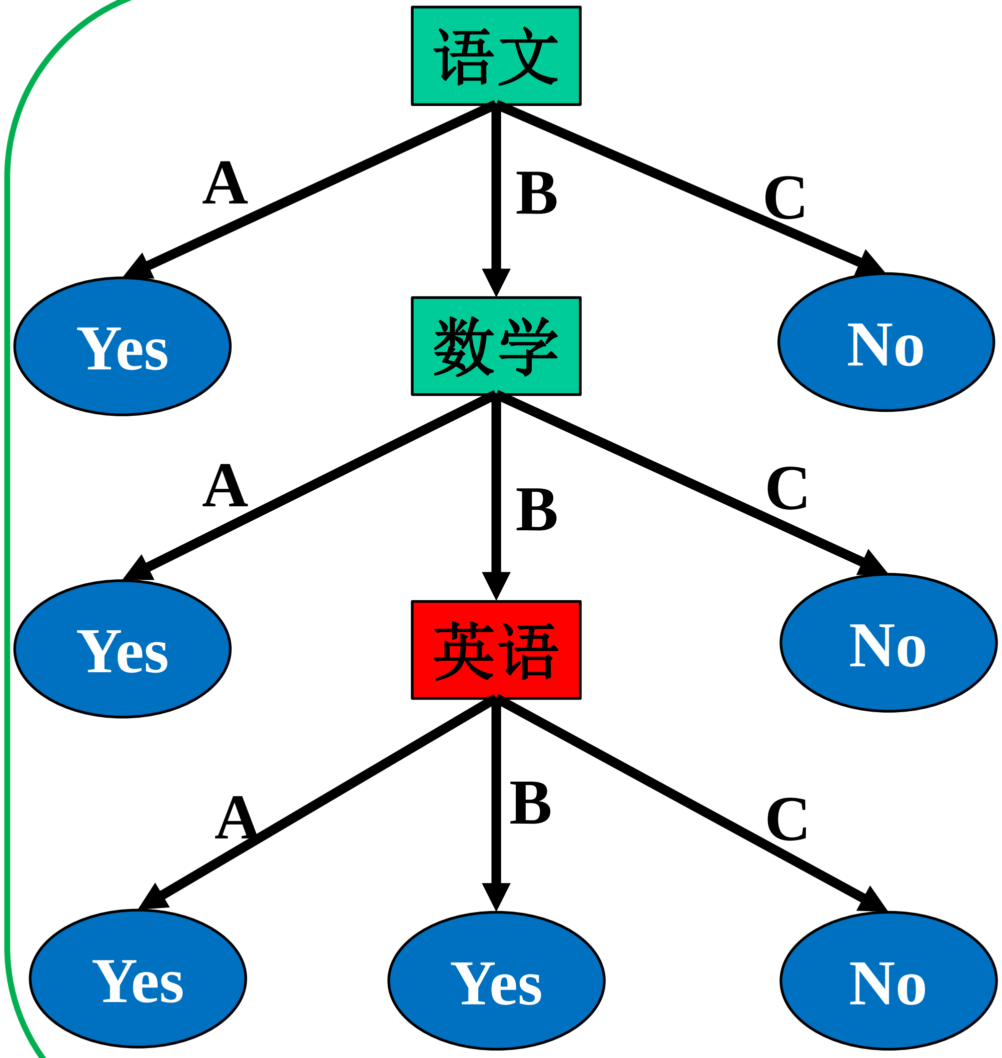
预剪枝决策：禁止划分

预剪枝结果

决策树桩(decision Stump)



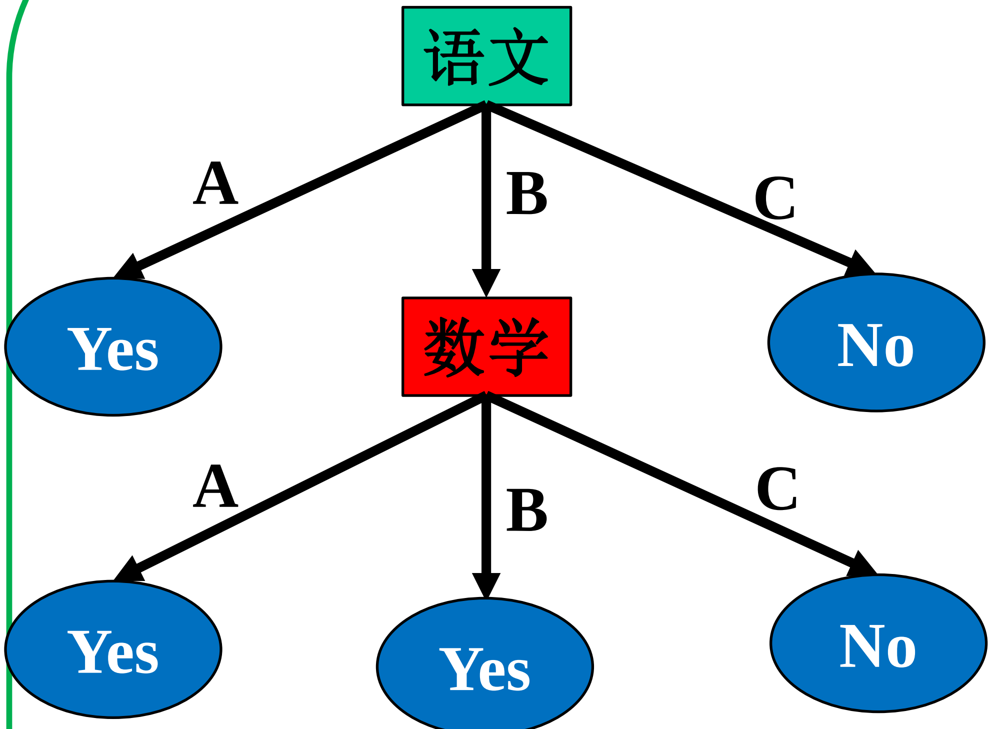
后剪枝（Postpruning）



真实值	剪枝前	剪枝后
Yes	Yes	Yes
No	No	No
Yes	No	No
No	Yes	Yes
No	Yes	Yes
Yes	Yes	Yes

剪枝前：50%
剪枝后：50%
后剪枝决策：剪枝

后剪枝（Postpruning）

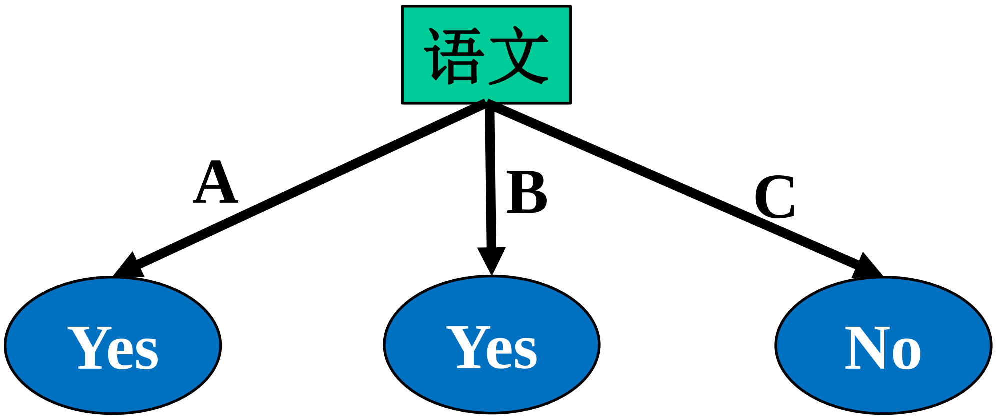


真实值	剪枝前	剪枝后
Yes	Yes	Yes
No	No	No
Yes	No	Yes
No	Yes	Yes
No	Yes	Yes
Yes	Yes	Yes

剪枝前： 50%
剪枝后： 66.66 %
后剪枝决策：剪枝

后剪枝结果

决策树桩(decision Stump)



剪枝策略分析

- 预剪枝：
  - 优点：减少属性划分与测试时间开销。
  - 缺点：可能造成欠拟合。
- 后剪枝：
  - 优点：减少欠拟合风险！
  - 缺点：时间开销大。

连续值处理

动机：利用决策树解决连续属性分类问题。

方法： 连续属性离散化（二分法）。

假设连续属性 a 在数据集上出现n个不同的取值$\{a^1, a^2, \ldots, a^n\}$。

定义候选划分点集合：

$$T_a = \left\{ \frac{a^i + a^{i+1}}{2} \middle| \quad 1 \le i \le n-1 \right\}$$

连续值处理

$$\mathrm{Gain}(D,a) = \max_{t\in T_a} \mathrm{Ent}(D) - \sum_{\lambda\in\{-,+\}} \frac{|D_t^\lambda|}{|D|}\mathrm{Ent}(D_t^\lambda)$$

其中D_t^+包含所有在属性a上取值大于 t 的样本，而D_t^-包含所有在属性a上取值小于 t 的样本。

注意：和离散情况不同，属性a划分完之后还可作为后代结点的划分属性。

决策树总结 I

- 决策树算法的核心是如何确定最佳划分准则。
- 我们学习了三种经典的决策树算法：

  ID3算法：信息增益（Gain）

  C4.5算法：信息增益率 (Gain Ratio)

  CART算法：基尼指数 (Gini Index)
- 决策树性能的指标：泛化能力

决策树总结 II

- 我们可以对决策树进行剪枝操作，是否剪枝取决于生成的决策树在验证集上的精度
- 预剪枝：在决策树生成过程中进行裁剪
- 后剪枝：在决策树生成之后再进行裁剪
- 对于连续属性，我们可以用二分法进行离散，再生成决策树。
- 对于缺失值的处理（自学）。