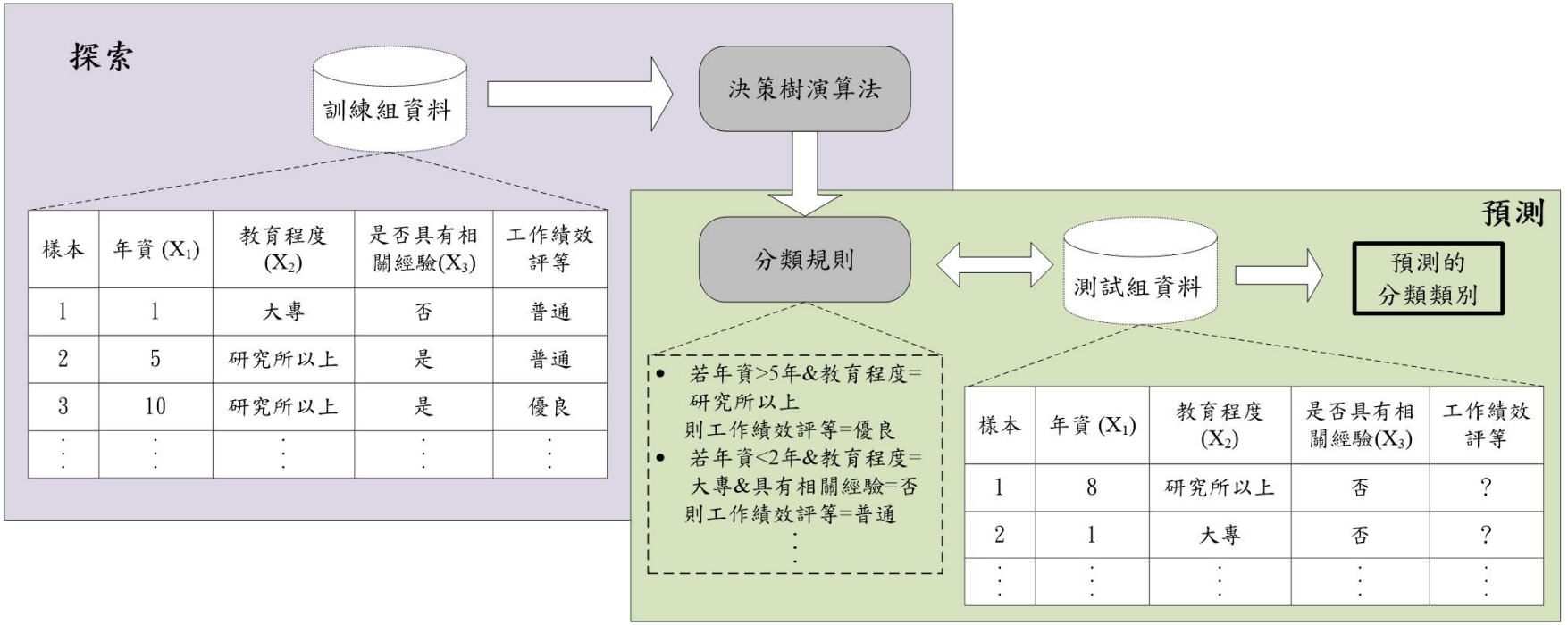
決策樹 (Decision Tree)

2018.10.25

決策樹

- 決策樹的主要功能，是藉由分類已知的事例來建立一樹狀結構，並從中歸納出事例裡的某些規律；而產生出來的決策樹，也能利用來做樣本外的預測。
- 決策樹是功能強大且相當受歡迎的分類和預測工具。這項以樹狀圖為基礎的方法，其吸引人之處在於決策樹具有規則，和類神經網路不同。規則可以用文字來表達，讓人類了解。

- 具有監督式的特徵萃取與描述的功能，將輸入變數根據目標設定來選擇分枝變數與分枝方式，並以樹枝狀的層級架構呈現，以萃取分類規則



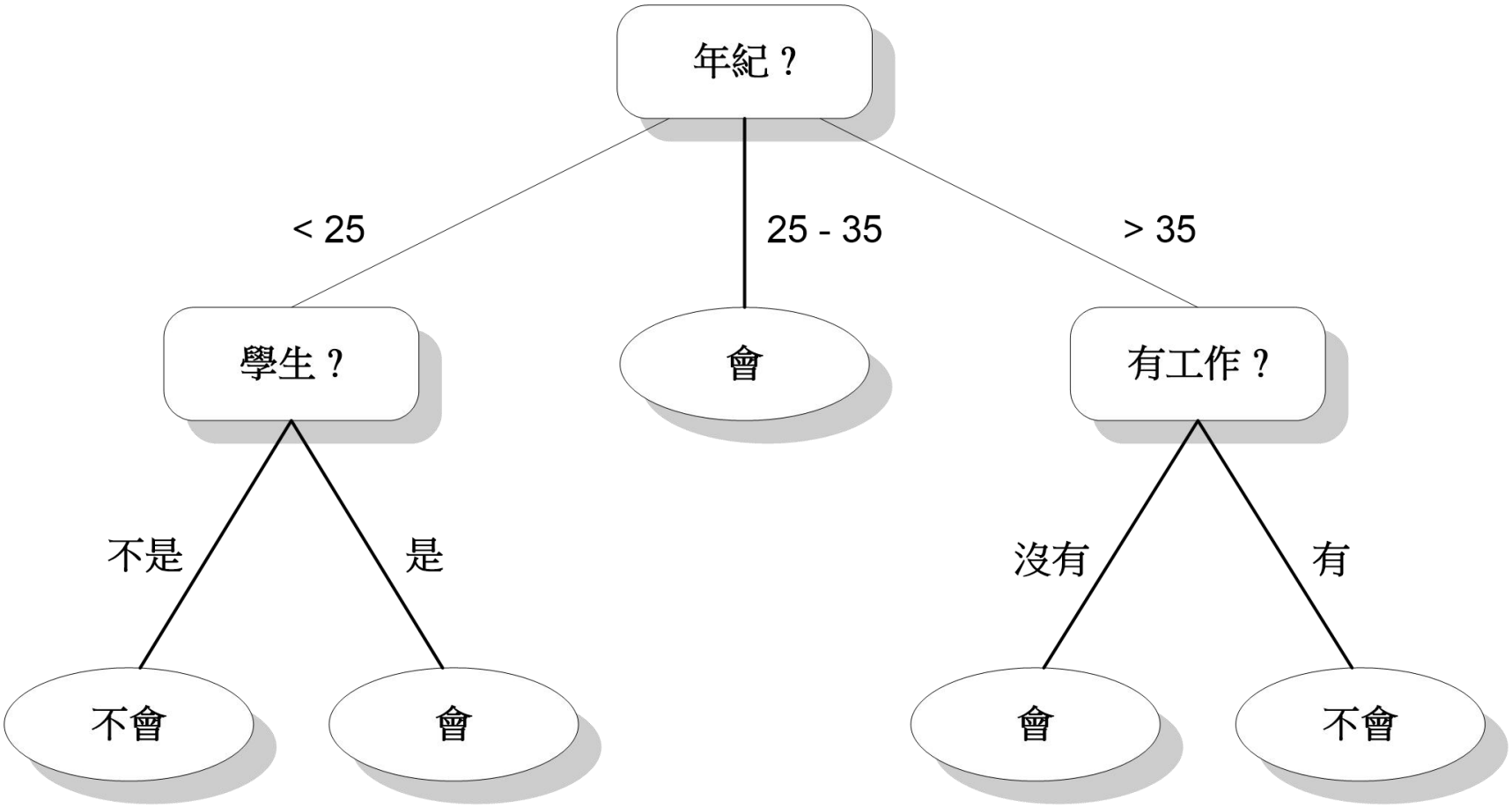
樣本	年資 (X_1)	教育程度 (X_2)	是否具有相關經驗(X_3)	工作績效評等
1	1	大專	否	普通
2	5	研究所以上	是	普通
3	10	研究所以上	是	優良
⋮	⋮	⋮	⋮	⋮

樣本	年資 (X_1)	教育程度 (X_2)	是否具有相關經驗(X_3)	工作績效評等
1	8	研究所以上	否	?
2	1	大專	否	?
⋮	⋮	⋮	⋮	⋮

決策樹基本觀念

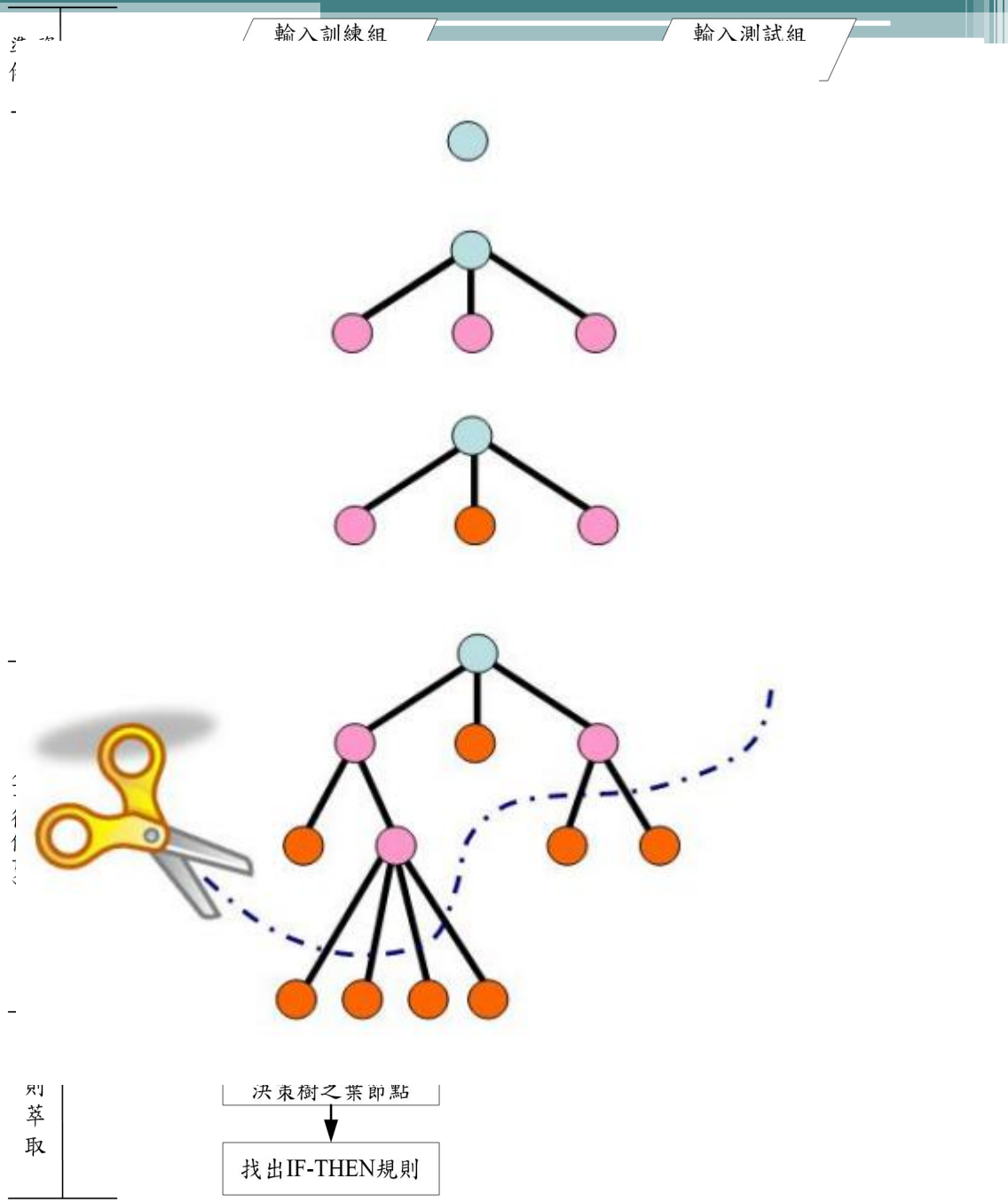
- 一筆資料從根部的節點進入決策樹。在根部，應用一項測驗來決定這筆資料該進入下一層的哪一個子節點（child node）。選擇一開始的測驗有不同的演算法，但目的都是一樣的：這個過程一再重複，直到資料到達葉部節點（leaf node）。
- 從根部到每一個葉部都有一套獨特的路徑，這個路徑就是用來分類資料規則的一種表達方式。

用決策樹預估「是否會玩網路遊戲」



決策樹的建構

1. 資料準備
2. 決策樹生長
3. 修剪
4. 規則萃取

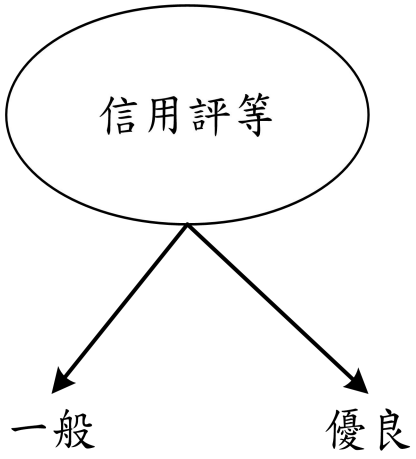


資料準備 (1/2)

- 決策樹的分析資料包含兩種變數：
  1. 根據問題所決定的目標變數
  2. 根據問題背景與環境所選擇的各種屬性作為分枝變數

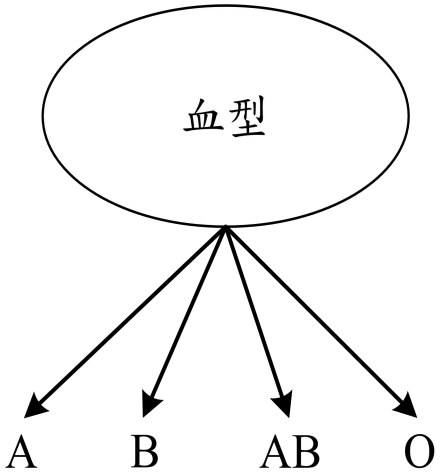
二元屬性

其測試條件可以產生兩種結果



名目屬性

結果可用不同屬性值來表示



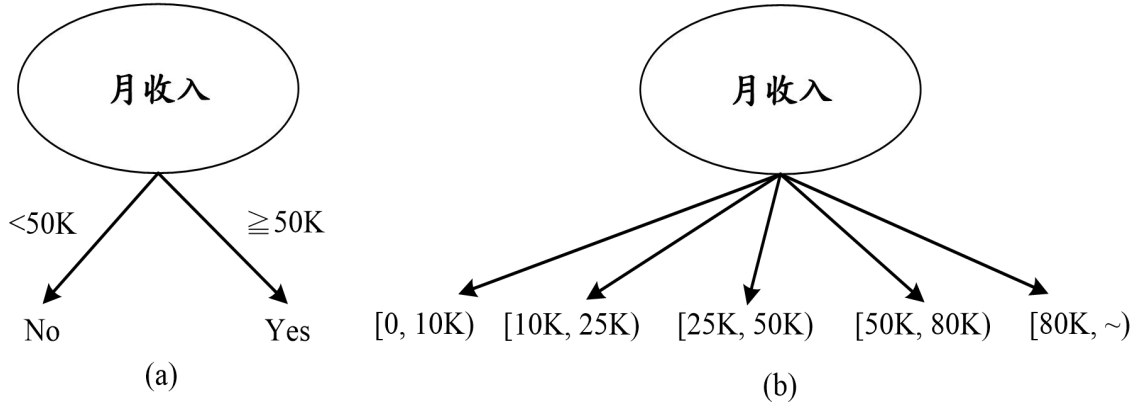
資料準備 (2/2)

順序屬性

- 可以生成二元或二元以上的分割，其屬性可以群組
- 群組必須不違反其屬性值特性

連續屬性

- 可表示成 X<a或 X≧ a的關係
- 須考慮到所有可能的分割點y，再選出最好的分割

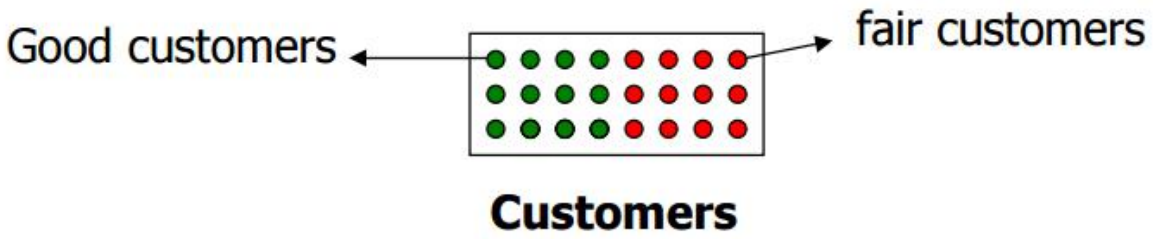


決策樹生長

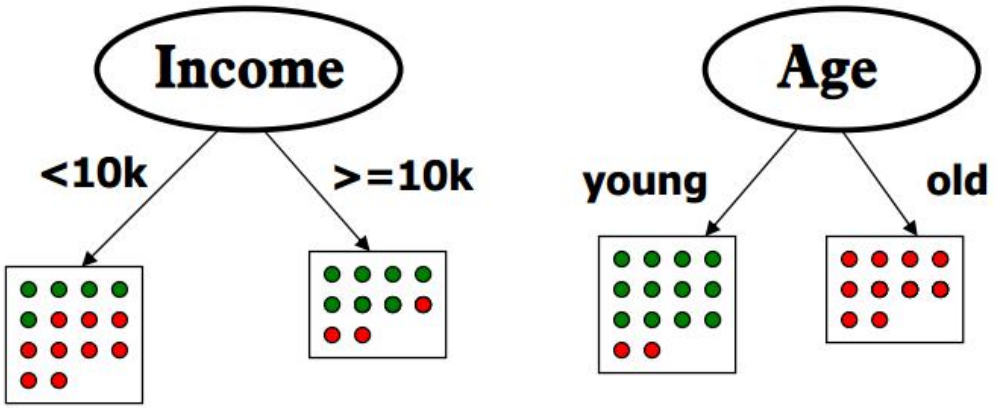
- 利用分枝準則(splitting criteria)決定樹的規模大小，包含樹的寬度以及深度
- 常見的分枝準則：
  - 資訊增益 (information gain) -> ID3
  - Gini係數 (Gini index) -> CART
  - 卡方統計量 (Chi-square statistic) -> CHAID
  - 資訊增益比 (information gain ratio) -> C4.5
- 透過檢驗分枝屬性的顯著性後，分枝準則即能找出具有最佳分枝結果的屬性

分枝準則的基本概念

- 假設有一個表格共有24筆顧客資料。其類別欄位為 “Customers” ，可分成 “好客人 Good Customers” 與 “一 般客人 Fair Customer” 兩類。

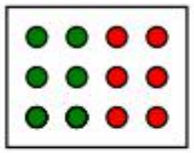


- 分別用Income和Age兩個欄位，對這24筆顧客資料加 以分割，結果如下。



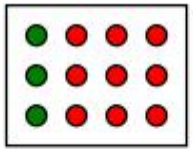
Age 優於 Income

- 分割結果中，若具有較高同質性(Homogeneous)類別的節點，則該分割結果愈佳。
- 因此，需要檢驗節點的不純度(Node Impurity)，



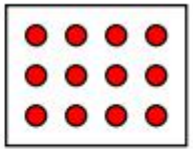
50% red
50% green

High degree of impurity



75% red
25% green

Low degree of impurity

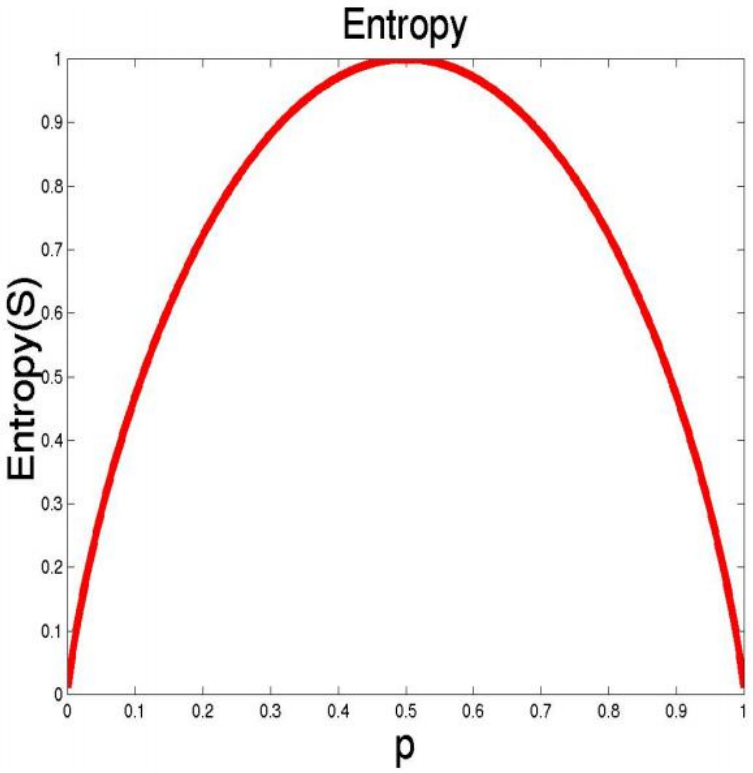


100% red
0% green

pure

ID3 (Iterative Dichotomiser)

- ID3在建構決策樹過程中，以資訊增益(Information Gain)為準則，並選擇最大的資訊增益值作為分類屬性。
- 以熵 (Entropy) 為基礎



資訊增益(Information Gain) (1/2)

- 資訊衡量(Information Measurement)是根據不同訊息的概似值或機率，以衡量不同條件下的資訊量

$$\text{Info(D)} = -\frac{x_{.1}}{N}\log_2(\frac{x_{.1}}{N}) - \frac{x_{.2}}{N}\log_2(\frac{x_{.2}}{N}) - \cdots - \frac{x_{.k}}{N}\log_2(\frac{x_{.k}}{N})$$

$$= -\sum_{j=1}^{k} p_j \cdot \log_2(p_j)$$

- Info(D)又稱熵(entropy)
  - 衡量資料離散程度或亂度，作為評估訓練資料集合D下所有類別的期望訊息
  - 各類別出現的機率相等，則熵值為1，表示分類訊息雜亂度最高

資訊增益(Information Gain)(2/2)

- 假設該資料集合D要根據屬性A進行分割，產生共L個資料分割集合 D_i ，其中 $x_{i.}$ 為各屬性值 A_i 下的分割資料總個數， x_{ij} 為屬性值 A_i 下且為 C_j 類別的個數 Info(A_i) 為：

$$\text{Info}(\text{A}_\text{i}) = -\frac{x_{i1}}{x_{i.}}\log_2\left(\frac{x_{i1}}{x_{i.}}\right) - \frac{x_{i2}}{x_{i.}}\log_2\left(\frac{x_{i2}}{x_{i.}}\right) - \cdots - \frac{x_{ik}}{x_{i.}}\log_2\left(\frac{x_{ik}}{x_{i.}}\right)$$

- 屬性A的資訊則根據各屬性值下的資料個數多寡決定

$$\text{Info}_\text{A}(\text{D}) = \frac{x_{1.}}{N}\text{Info}(A_1) + \frac{x_{2.}}{N}\text{Info}(A_2) + \cdots + \frac{x_{l.}}{N}\text{Info}(A_l)$$

$$= \sum_{i=1}^{l}\frac{x_{i.}}{N}\text{Info}(A_i)$$

$$Gain(\text{A}) = \text{Info}(D) - \text{Info}_\text{A}(D)$$

- 資訊增益可表示為：

範例：資訊增益

- 假設某公司人力資源部門欲瞭解職員的表現，抽取10位現職員工為樣本，將連續屬性離散化後如下表

職員	年資(A)	教育程度(B)	具備相關經驗(C)	員工表現
001	5年以下	研究所	是	優等
002	10年以上	研究所	否	普通
003	5年以下	研究所	是	優等
004	5年以下	大專	是	普通
005	5年以下	研究所	否	優等
006	10年以上	研究所	是	優等
007	5年至10年	大專	否	普通
008	5年至10年	研究所	是	優等
009	5年至10年	大專	否	普通
010	5年以下	研究所	是	普通

根據(A)、(B)、(C)三個屬性分別計算：

$Info(D)$

$Info(A_i)$

$Gain$

資訊增益比 (1/2)

- ID3所採用的資訊增益會傾向選擇擁有許多不同數值的屬性
  - 前述例子的職員編號會產生出許多分支，且每一個分支都是很單一的結果，其資訊增益會最大。但這個屬性對於建立決策樹是沒有意義的。
- C4.5演算法利用屬性的增益比率(Gain Ratio)克服問題(資訊獲利正規化)。而求算某屬性A的增益比率時，除資訊獲增益外，尚需計算該屬性的分割資訊值(Split Information)：

$$Split\ Info(A) = -\sum_{i=1}^{l} \frac{x_{i.}}{N} \cdot \log_2(\frac{x_{i.}}{N})$$

資訊增益比 (2/2)

- 考慮候選屬性本身所攜帶的訊息，再將這些訊息轉換至決策樹，經由計算資訊增益與分枝屬性的資訊量之比值來找出最適合的分枝屬性

$$\mathrm{GR}(A) = \frac{Gain(A)}{Split\ \ Info(A)}$$

Gini係數

- CART (Classification and Regression Tree)由Friedman等人於 1980年代提出，是一種產生二元樹的技術，以吉尼係數做為選擇屬性的依據。
- CART與ID3、C4.5、C5.0演算法的最大相異之處是其在每一個節點上都是採用二分法，也就是一次只能夠有兩個子節點， ID3、C4.5、C5.0則在每一個節點上可以產生不同數量的分枝。
- Gini係數衡量資料集合對於所有類別的不純度(impurity)

$$Gini(D) = 1 - \sum_{j=1}^{k} p_j^2$$

Gini係數分割原理

- 各屬性值 A_i 下資料集合之不純度

$$Gini(A_i) = 1 - (\frac{x_{i1}}{x_{i.}})^2 - (\frac{x_{i2}}{x_{i.}})^2 - \cdots - (\frac{x_{ik}}{x_{i.}})^2 = 1 - \sum_{j=1}^{k} (\frac{x_{ij}}{x_{i.}})^2$$

- 屬性A的總資料不純度則等於所有屬性值分割下的期望平均

$$Gini_A(D) = \frac{x_{1.}}{N} Gini(A_1) + \frac{x_{2.}}{N} Gini(A_2) + \cdots + \frac{x_{l.}}{N} Gini(A_l)$$

- 計算其他屬性作為分枝變數所能帶來的純度

$$\Delta \text{Gini}(A) = Gini(D) - Gini_A(D)$$

- 挑選擁有最大不純度的降低值、或吉尼係數GiniA(S)最小

卡方統計量（ χ^2 statistic ）

- 以列聯表計算兩變數間的相依程度，當計算出的樣本卡方統計值越大，表示兩變數間的相依程度越高

年齡 \ 表現	優秀	普通	總和
(A_1)	3 (2.5)	2 (2.5)	5
(A_2)	1 (1.5)	2 (1.5)	3
(A_3)	1 (1.0)	1 (1.0)	2
總和	5	5	10

教育程度 \ 表現	優秀	普通	總和
(B_1)大專以下	0 (1.5)	3 (1.5)	3
(B_2)研究所以上	5 (3.5)	2 (3.5)	7
總和	5	5	10

年齡 \ 表現	優秀	普通	總和
(A_1) 是	4 (3)	2 (3)	6
(A_2) 否	1 (2)	3 (2)	4
總和	5	5	10

$$\chi^2 = \sum_{i=1}^{l}\sum_{j=1}^{k}\frac{(x_{ij}-E_{ij})^2}{E_{ij}}, \qquad E_{ij} = \frac{x_{i.}\cdot x_{.j}}{N}$$

$$\chi^2(\text{年資}) = \frac{(3-2.5)^2}{2.5}+\frac{(2-2.5)^2}{2.5}+\frac{(1-2.5)^2}{2.5}+\frac{(2-2.5)^2}{2.5}+$$
$$\frac{(1-2.5)^2}{2.5}+\frac{(1-2.5)^2}{2.5}=0.533$$

$$\chi^2(\text{教育程度}) = \frac{(0-1.5)^2}{1.5}+\frac{(3-1.5)^2}{1.5}+\frac{(5-3.5)^2}{3.5}+\frac{(2-3.5)^2}{3.5}=4.286$$

$$\chi^2(\text{具備相關經驗}) = \frac{(4-3)^2}{3}+\frac{(2-3)^2}{3}+\frac{(1-2)^2}{2}+\frac{(3-2)^2}{2}=1.67$$

決策樹的演算法

演算法		CART	C4.5/C5.0	CHAID
處理資料型態		離散、連續	離散、連續	離散
連續型資料分枝方式		只分2枝	不受限制	無法處理
分枝準則	類別型相依變數	Gini分散度指標	資訊增益比	卡方檢定
	連續型相依變數	變異數縮減	變異數縮減	卡方檢定或F檢定（需先轉化為類別變數）
分枝方法	類別型獨立變數	二元分枝	多元分枝	多元分枝
	連續型獨立變數	二元分枝	二元分枝	多元分枝（需先轉化為類別變數）
修剪方法		成本複雜性修剪	基於錯誤的修剪	無

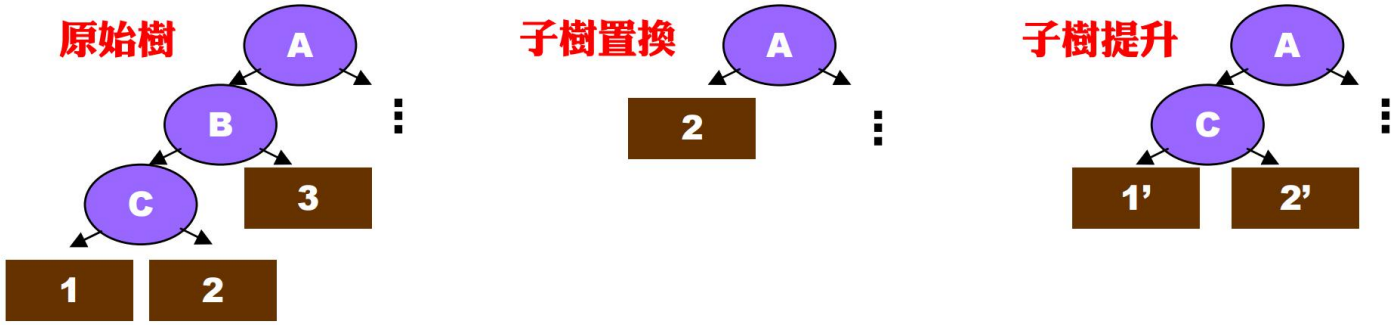
奧坎剃刀理論(Ockham's Razor)

- 當實驗取得的事實能夠得到說明時，不應增添不必要的假設，應把它一剃而盡，此說後被稱為奧坎剃刀
  - 最簡單的解釋就是最好的解釋 (The simplest explanation is the best)。
  - 除非必須，否則無須增多。
- 修剪決策樹可移除不可信賴的分支。有兩種修剪方法：
  - 事前修剪 (Prepruning)
  - 事後修剪 (Postpruning)

決策樹修剪

- 事先修剪(pre-pruning)
  - 事先設定停止決策樹生長的門檻值，當分割的評估值未達此門檻值時，就會停止擴長
  - 優點：較具有執行效率
  - 缺點：可能過度修剪(over-pruning)、門檻值設定不易
- 事後修剪(post-pruning)
  - 在樹完全長成後再修剪，引入測試組樣本來評估決策樹對於新輸入資料的分類與預測結果
  - 優點：可解決過度配適，避免產生稀少樣本樹的葉節點，以及加強對雜訊的忍受程度
  - 缺點：效率較低

事後修剪 (Postpruning)



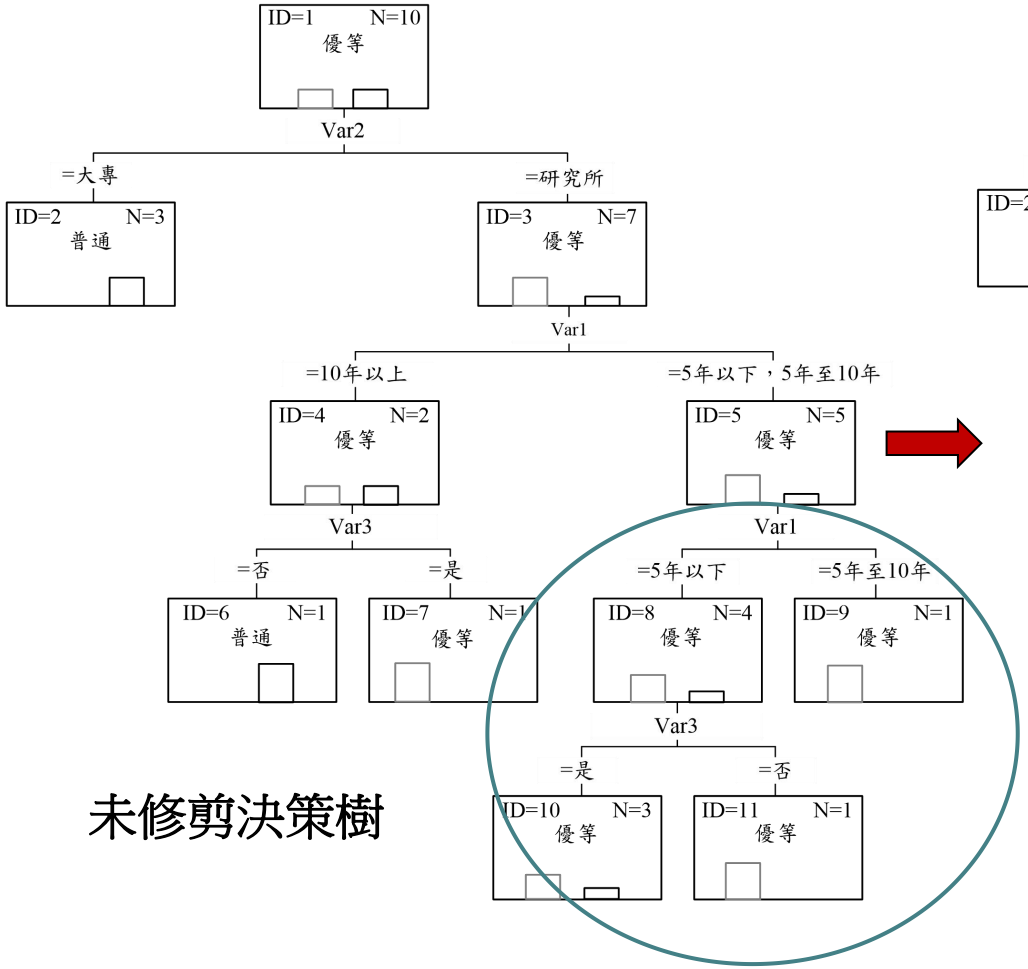
- CART使用成本複雜度修剪(Cost Complexity Pruning)方法，以 決策樹的葉節點與錯誤率構成成本複雜度函數。

$$R_{\alpha}(t) = R(t) + \alpha \times N_{\text{leaf}}$$

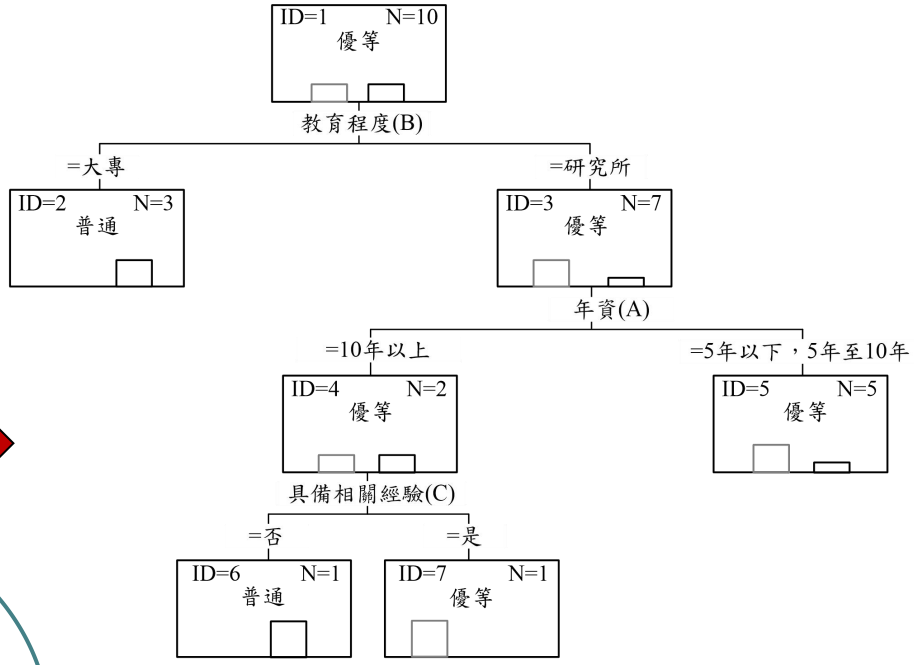
- C4.5使用悲觀修剪(Pessimistic Pruning)方法，也是使用錯誤率來進行修剪。

$$CL = \sum_{x=0}^{E} C_x^N \, p^x (1-p)^{N-x}$$

[範例4.1] 決策樹修剪示例



未修剪決策樹



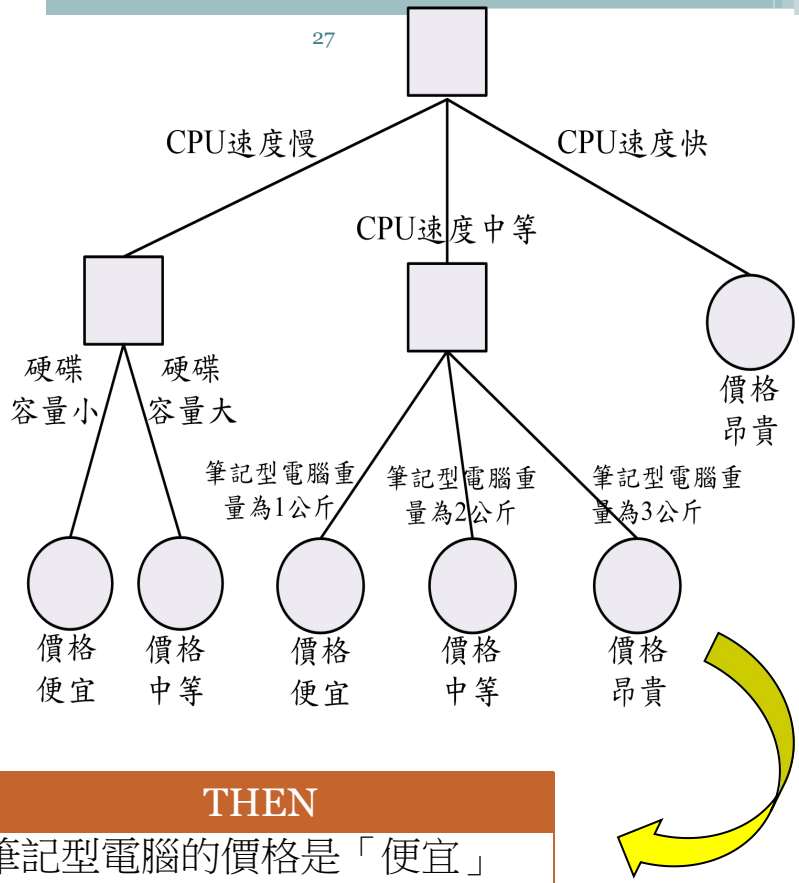
修剪後決策樹

合併連續值屬性

- 許多決策樹學習方法限制為取離散值的屬性
  - 決策樹要預測的目標屬性必須是離散的
  - 樹的內部節點的屬性也必須是離散的
- 簡單刪除上面第2個限制的方法
  - 透過動態地定義新的離散值屬性來實現，即先把連續值屬性的值 域分割為離散的區間集合，或設定門檻值以進行二分法
- 例子: Age
  - 使用門檻值，大於門檻值的資料為yes，小於門檻值的為no。
  - 使用區間值，以區分出多個離散區間。

規則萃取

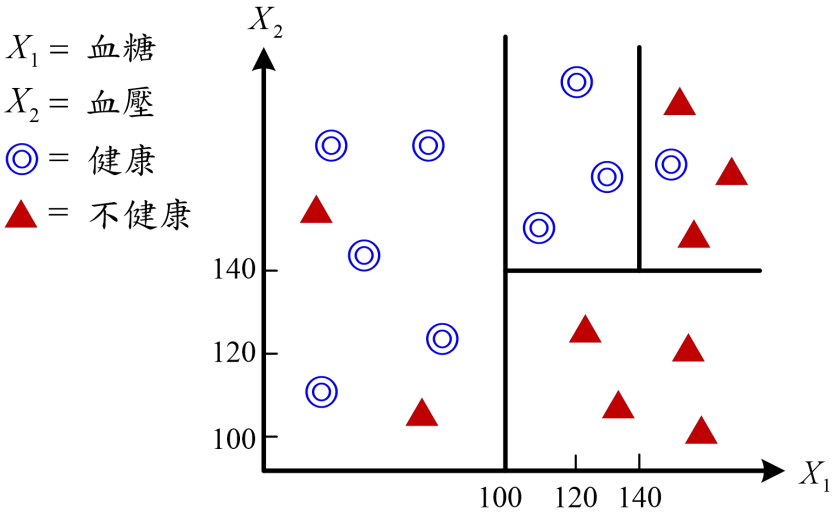
- 利用決策樹萃取資料中所隱含的資訊
- IF-THEN規則即為從根節點至葉節點的可能路徑(path)
- 沿著可能路徑可串連起作為分枝變數的屬性，形成一套具因果關係的分類模型，以分類資料並預測

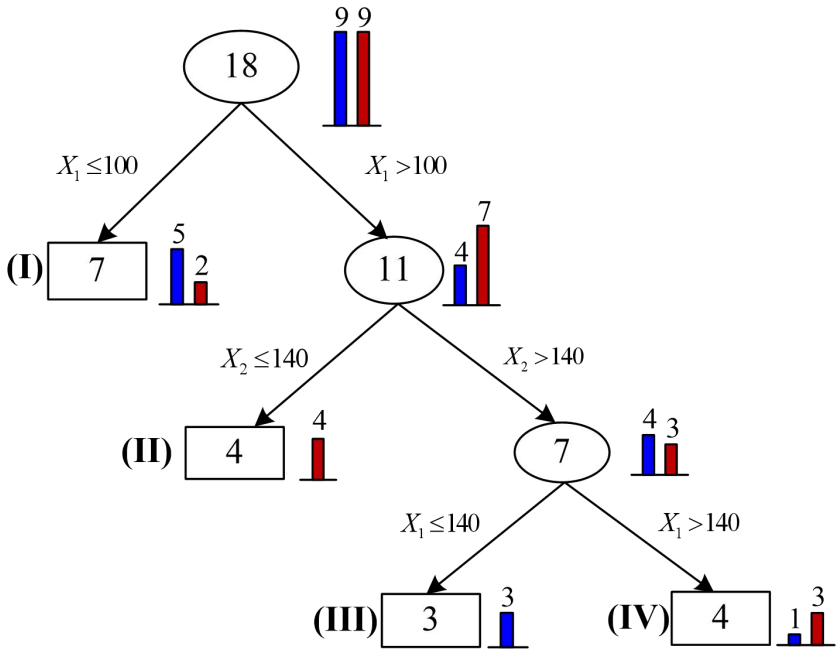


IF	THEN
若「CPU速度慢」，且「硬碟容量小」	筆記型電腦的價格是「便宜」
若「CPU速度慢」，且「硬碟容量大」	筆記型電腦的價格是「中等」
若「CPU速度中等」，且「電腦重量為1公斤」	筆記型電腦的價格是「昂貴」
若「CPU速度中等」，且「電腦重量為2公斤」	筆記型電腦的價格是「中等」
若「CPU速度中等」，且「電腦重量為3公斤」	筆記型電腦的價格是「便宜」
若「CPU速度快」	筆記型電腦的價格是「昂貴」

決策樹的非線性思考

- 以健康為目標建立決策樹，衡量18位人員的血糖與血壓，並以血糖最低標準100與140以及血壓最低標準90來分類





發現規則：

(I)若 $X_1<100$，則為健康(◎)
(II)若$X_1>100$且$X_2<140$，則為不健康(▲)
(III)若$100<X_1\le140$且$X_2>140$，則為健康(◎)
(IV)若$X_1>140$且$X_2>140$，則為不健康(▲)

Decision Tree in R: rpart

- rpart(*formula*, data=, method=,control=)

formula	is in the format outcome ~ predictor1+predictor2+predictor3+ect.
data=	specifies the data frame
method=	"class" for a classification tree "anova" for a regression tree
control=	optional parameters for controlling tree growth. For example, control=rpart.control(minsplit=30, cp=0.001) requires that the minimum number of observations in a node be 30 before attempting a split and that a split must decrease the overall lack of fit by a factor of 0.001 (cost complexity factor) before being attempted.