


資料探勘： 概念與技術

— 第六章 —

第六章.判別與預測

- 何謂判別? 何謂預測? 
- 判別與預測議題
- 決策樹判別
- 貝氏判別
- 規則判別
- 倒傳遞判別
- 支持向量機 (SVM)
- 關連判別
- 偷懶學習 (從鄰居學習)
- 其他判別方法
- 預測
- 正確與錯誤指標
- 組合方法
- 模型選擇
- 總結

判別與預測

■ 判別

- 預測類別類形 (離散或類別值)
- 根據訓練資料及其判斷屬性類別值來判別資料, 並將其應用於判別新資料

■ 預測

- 對連續值函數建立模型 例., 預測未知或遺失值

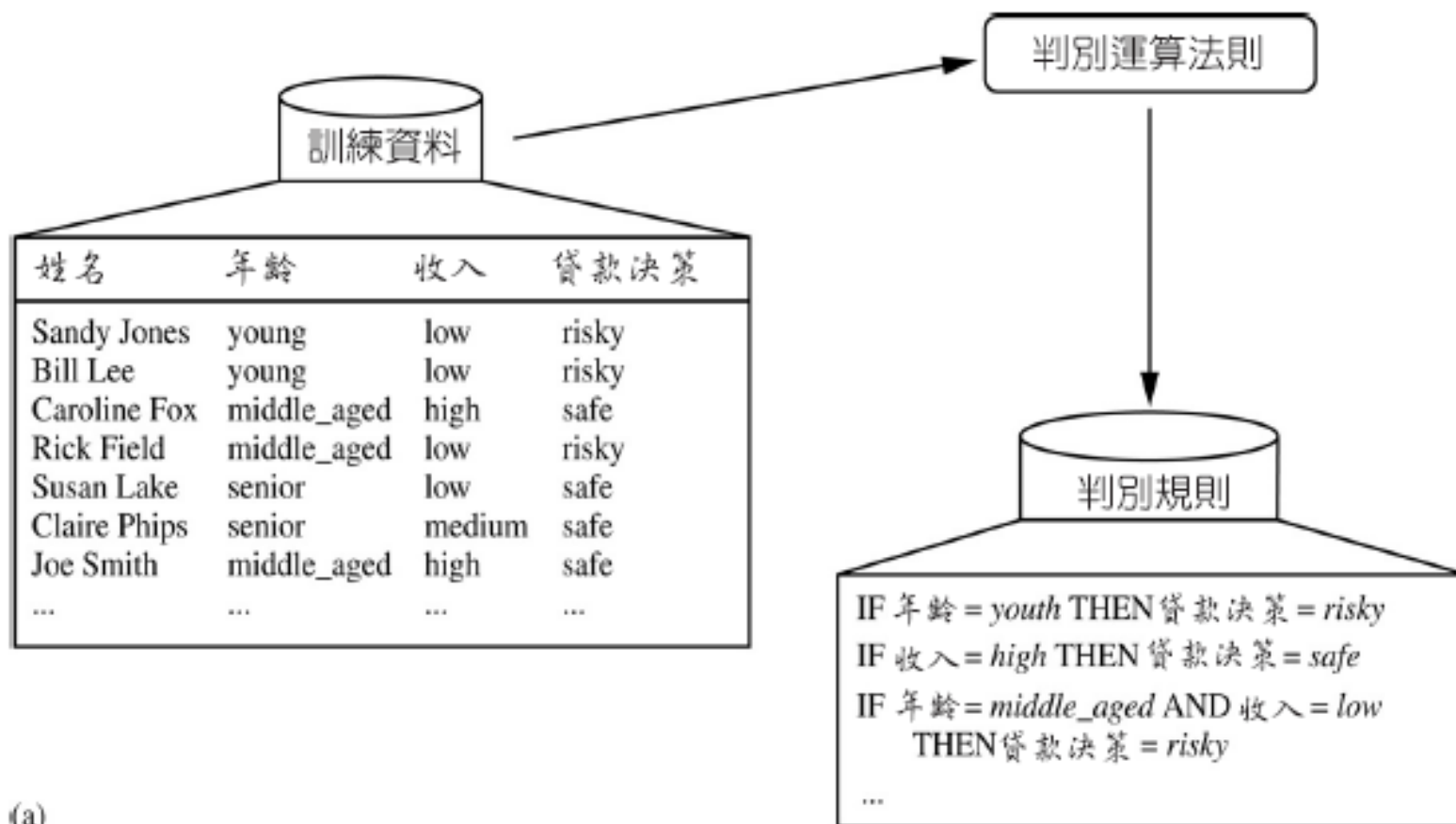
■ 典型應用

- 信用核准
- 目標市場
- 醫學分析
- 詐騙檢測

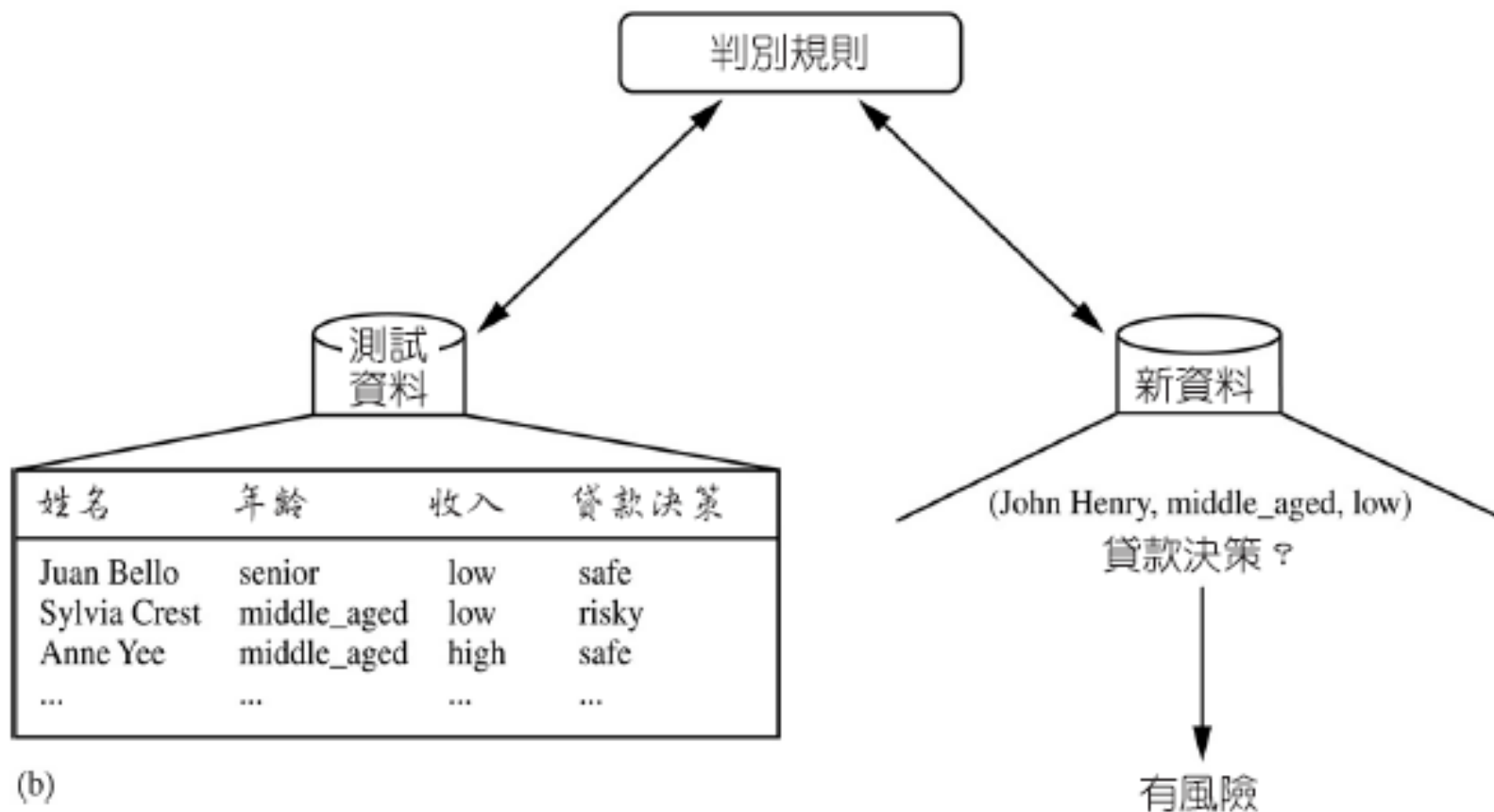
判別一兩個步驟的程序

- **模型建立**: 描述一組事先定義的類別
 - 每個值組 (tuple) 均屬於某個類別，而類別由類別屬性 (class label attribute) 決定
 - 構成訓練集合的值組集合稱為訓練值組集
 - 模型用判別規則、決策樹或數學公式來表示
- **模型使用**: 用於判別未來或未知個體
 - 預估模型的正確性
 - 使用已知類別測試資料來預估模型的正確性
 - 正確率是指測試資料被正確分類的百分比
 - 測試資料與訓練資料獨立, 否則會產生過適(over-fitting)的現象
 - 如果測試正確率是可接受的，模型將用來判別未知的資料

步驟 (1): 模型建立




步驟 (2): 使用模型進行判別



監督與非監督學習

- 監督式學習 (判別)
 - 監督: 從訓練資料學習哪個值組屬於哪個類別
 - 根據訓練資料判斷新資料類別
- 非監督式學習 (分群)
 - 訓練資料的類別是無法事先知道或是未知
 - 將資料依照相似值組進行群組

第六章.判別與預測

- 何謂判別? 何謂預測?
- 判別與預測議題 
- 決策樹判別
- 貝氏判別
- 規則判別
- 倒傳遞判別
- 支持向量機 (SVM)
- 關連判別
- 偷懶學習 (從鄰居學習)
- 其他判別方法
- 預測
- 正確與錯誤指標
- 組合方法
- 模型選擇
- 總結


議題:資料準備

- 資料清除
 - 用於移除雜訊與設定遺失值
- 相關分析 (屬性選擇)
 - 移除多餘或不相關的屬性
- 資料轉換
 - 一般或正規化資料

議題：評估判別方法

- 正確率
 - 判別正確率：預測類別標籤
 - 預測正確率：猜測預測屬性值
- 速度
 - 建立模型時間（訓練時間）
 - 使用模型時間（判別/預測時間）
- 穩固性：在雜訊與不存在值的資料中正確判別或預測的能力
- 可量度性：對於龐大資料有效的建立判別與預測模型
- 解釋力
 - 對判別或預測所提供的了解程度
- 其他的研究包含發掘屬性與其相關判別的關係

第六章.判別與預測

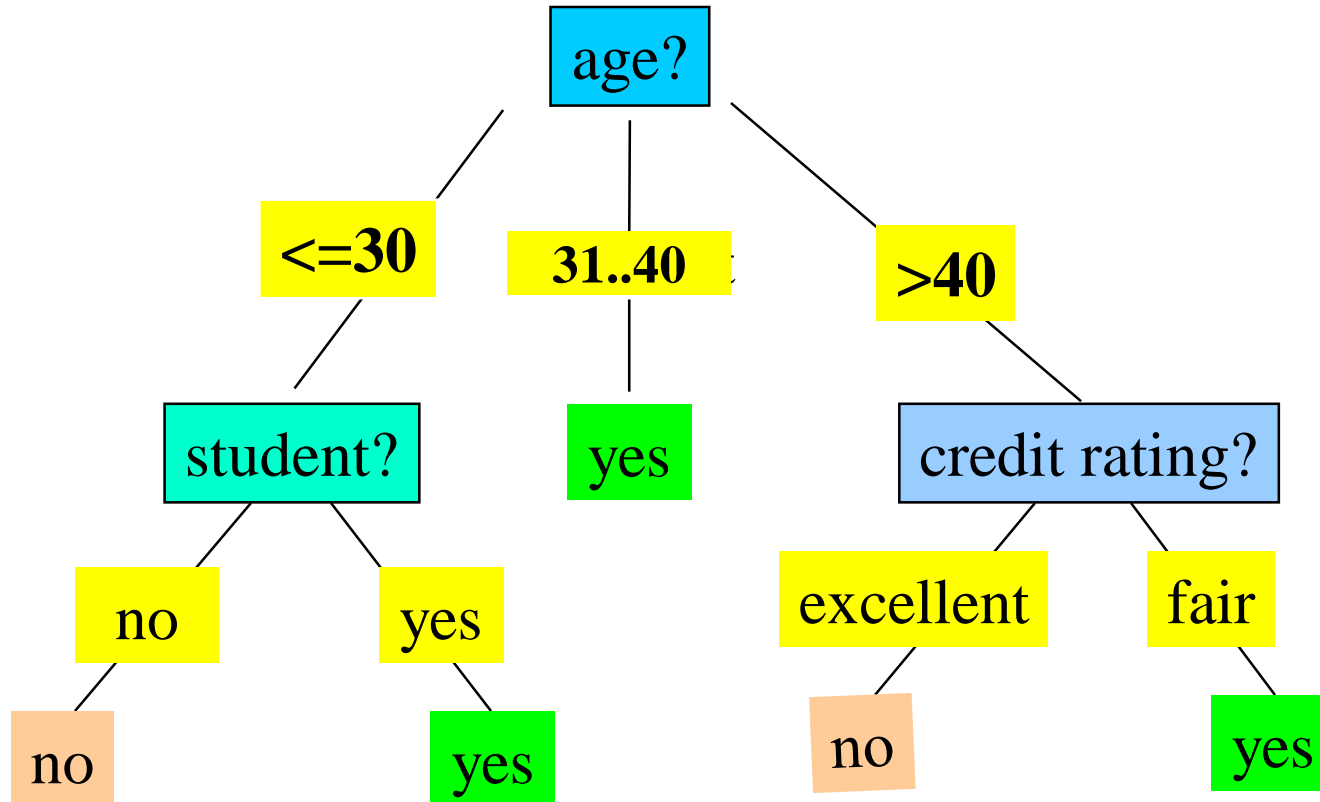
- 何謂判別? 何謂預測?
- 判別與預測議題
- 決策樹判別 
- 貝氏判別
- 規則判別
- 倒傳遞判別
- 支持向量機 (SVM)
- 關連判別
- 偷懶學習 (從鄰居學習)
- 其他判別方法
- 預測
- 正確與錯誤指標
- 組合方法
- 模型選擇
- 總結

決策樹歸納: 訓練資料集

沿用
Quinlan's
ID3
(Playing
Tennis)範
例

age	income	student	credit_rating	buys_computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
31...40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31...40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31...40	medium	no	excellent	yes
31...40	high	yes	fair	yes
>40	medium	no	excellent	no

輸出: “*buys_computer*” 決策樹



決策樹歸納方法

- 基本方法 (貪婪式方法)
 - 決策樹是用由上而下 (top-down) 遞迴式 (recursive) 的分割與克服 (divide-and-conquer) 的方式建立
 - 一開始根節點包含所有的資料
 - 屬性為類別資料 (如果為連續資料必須進行離散化)
 - 根據類別值來選擇最好的屬性進行值組的區別
 - 套用屬性選擇指標如資訊獲利 (information gain) 或吉尼係數
- 停止分割條件
 - 所有值組屬於相同類別
 - 沒有屬性在屬性列可繼續進行分割 – 在這種例子要套用多數決
 - 分割中不包含任何值組

屬性選擇指標:資訊獲利 (ID3/C4.5)

- 選擇屬性中擁有最高資訊獲利
- p_i 為任一 D 中的值組被歸類為類別 C_i 的機率，它可由 $|C_i \cap D|/|D|$ 得之
- 期望訊息 (熵) 用於判別 D 中值組：

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

- 在分割後，為了達到一致的判別我們仍需要下列的訊息：

$$Info_A(D) = \sum_{j=1}^v \frac{|D_j|}{|D|} \times I(D_j)$$

- 分割屬性 A 的訊息獲利

$$Gain(A) = Info(D) - Info_A(D)$$

屬性選擇: 資訊獲利

- Class P: buys_computer = "yes"
- Class N: buys_computer = "no"

$$Info(D) = I(9,5) = -\frac{9}{14} \log_2\left(\frac{9}{14}\right) - \frac{5}{14} \log_2\left(\frac{5}{14}\right) = 0.940$$

$$Info_{age}(D) = \frac{5}{14} I(2,3) + \frac{4}{14} I(4,0) + \frac{5}{14} I(3,2) = 0.694$$

age	p _i	n _i	I(p _i , n _i)
<=30	2	3	0.971
31...40	4	0	0
>40	3	2	0.971

$\frac{5}{14} I(2,3)$ 代表 "age <=30" 在14樣本中佔了 5 個, 其中有 2 yes'es and 3 no's. 因此

$$Gain(age) = Info(D) - Info_{age}(D) = 0.246$$

相同的,

$$Gain(income) = 0.029$$

$$Gain(student) = 0.151$$

$$Gain(credit_rating) = 0.048$$

age	income	student	credit_rating	buys_computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
31...40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31...40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31...40	medium	no	excellent	yes
31...40	high	yes	fair	yes
>40	medium	no	excellent	no

用連續值來計算訊息獲利

- 當屬性A為連續值
- 對屬性A決定最佳的分割點
 - 將屬性A的值進行遞增排序
 - 一般來說兩個相鄰屬性值的中點 (midpoint) 會被設為可能分割點
 - $(a_i + a_{i+1})/2$ 為 a_i 與 a_{i+1} 的中點
 - 擁有最小期望訊息值的可能分割點會被設為屬性A的分割點
- 分割:
 - D1 包含D中滿足 $A \leq$ 分割點值組, D2 包含D中滿足 $A >$ 分割點值組

獲利比率 (C4.5)

- 訊息獲利會傾向於選擇擁有許多不同類別的屬性
- C4.5 (ID3延續)利用獲利比率克服問題 (資訊獲利正規化)

$$SplitInfo_A(D) = -\sum_{j=1}^v \frac{|D_j|}{|D|} \times \log_2\left(\frac{|D_j|}{|D|}\right)$$

- $GainRatio(A) = Gain(A)/SplitInfo(A)$
- 例. $SplitInfo_A(D) = -\frac{4}{14} \times \log_2\left(\frac{4}{14}\right) - \frac{6}{14} \times \log_2\left(\frac{6}{14}\right) - \frac{4}{14} \times \log_2\left(\frac{4}{14}\right) = 0.926$
 - $gain_ratio(i收入) = 0.029/0.926 = 0.031$
- 擁有最大獲利比例的屬性被設為分割屬性

吉尼係數 (CART, IBM IntelligentMiner)

- 資料集 D 包含 n 類別資料, 吉尼係數, $gini(D)$ 定義為

$$gini(D) = 1 - \sum_{j=1}^n p_j^2$$

p_j 為在 D 中的值組屬於類別 j 的機率

- 利用屬性 A 分割 D 為 D_1 與 D_2 。則根據此分割 D 的吉尼係數為

$$gini_A(D) = \frac{|D_1|}{|D|} gini(D_1) + \frac{|D_2|}{|D|} gini(D_2)$$

- 不純粹度的減少值為:

$$\Delta gini(A) = gini(D) - gini_A(D)$$

- 屬性擁有最大不純粹度的降低值, 則選為分割屬性(對每個屬性的所有可能分割點進行測試)

吉尼係數 (CART, IBM IntelligentMiner)

- 例. D 有9筆購買電腦(buy_computer)為yes，剩下5筆為no

$$gini(D) = 1 - \left(\frac{9}{14}\right)^2 - \left(\frac{5}{14}\right)^2 = 0.459$$

- 假設收入屬性分割 D 中 10 筆資料到 D_1 : {低, 中} 剩下 4 筆到 D_2

$$\begin{aligned} gini_{income \in \{low, medium\}}(D) &= \left(\frac{10}{14}\right) Gini(D_1) + \left(\frac{4}{14}\right) Gini(D_2) \\ &= \frac{10}{14} \left(1 - \left(\frac{6}{10}\right)^2 - \left(\frac{4}{10}\right)^2\right) + \frac{4}{14} \left(1 - \left(\frac{1}{4}\right)^2 - \left(\frac{3}{4}\right)^2\right) \\ &= 0.450 = Gini_{收入 \in \{高\}}(D) \end{aligned}$$

但是 $gini_{\{中, 高\}}$ 為 0.30 所以屬性收入的{中, 高}被選為分割條件因為它擁有最小的吉尼係數值

- 假設所有屬性為連續值
- 需要其他工具, 例., 分群, 取得可能分割值
- 修改後可套用至類別屬性

屬性選擇指標比較

- 三種常用指標
 - 訊息獲利：
 - 趨向於包含多個值的屬性
 - 獲利比例：
 - 會產生不平均的分割，也就是分割的一邊會非常小於另一邊
 - 吉尼係數：
 - 傾向於包含多個值的屬性
 - 當類別個數很多時會有困難
 - 傾向那些會導致平衡切割並且兩邊均為純粹的測試


過適與樹修剪

- 過適：歸納樹會對訓練資料產生過適問題
 - 許多分枝會導致異常，這些是由於資料中的雜訊與離異值所引起
 - 導致未知樣本的低正確率
- 兩種常用的方式
 - 事前修剪：透過決策樹不再增長的方式來達到修剪的目的，例佈在分割當指標低於某個界限值
 - 很難選擇適當界限值
 - 事後修剪：從全部的決策樹中移除子樹
 - 要使用不同於訓練資料的資料來決定最佳修剪樹

決策樹的可量度性

- **SLIQ** (EDBT'96 — Mehta et al.)
 - 使用儲存在記憶體中的屬性列 與類別列
- **SPRINT** (VLDB'96 — J. Shafer et al.)
 - 使用不同屬性列的資料結構
- **RainForest** (VLDB'98 — Gehrke, Ramakrishnan & Ganti)
 - 建立 AVC-set (屬性, 值, 類別標籤)
- **BOAT** (PODS'99 — Gehrke, Ganti, Ramakrishnan & Loh)
 - 使用統計技巧自助法來建立許多小樣本

第六章.判別與預測

- 何謂判別? 何謂預測?
- 判別與預測議題
- 決策樹判別
- 貝氏判別 
- 規則判別
- 倒傳遞判別
- 支持向量機 (SVM)
- 關連判別
- 偷懶學習 (從鄰居學習)
- 其他判別方法
- 預測
- 正確與錯誤指標
- 組合方法
- 模型選擇
- 總結

貝氏理論：基礎

- 假設 \mathbf{X} 為資料值組或稱為事證：類別未知
- 假設 H 為 \mathbf{X} 屬於某個類別 C
- 在判別問題我們想要決定 $P(H|\mathbf{X})$ ，而 $P(H|\mathbf{X})$ 代表在事證 \mathbf{X} 之下假設 H 成功的機率
- $P(H)$ (事前機率)
 - 例., 它代表不管任何訊息， \mathbf{X} 會買電腦的機率
- $P(\mathbf{X})$: \mathbf{X} 在樣本資料出現機率
- $P(\mathbf{X}|H)$ (事後機率), 當假設 H 成立下, 樣本 \mathbf{X} 出現的機率

貝氏理論

- 假設 \mathbf{X} 為訓練資料, 透過貝式理論假設 H 的事後機率為

$$P(H | \mathbf{X}) = \frac{P(\mathbf{X} | H)P(H)}{P(\mathbf{X})}$$

- 非正式的, 上式可表示為

posteriori = likelihood x prior/evidence

- 預測 \mathbf{X} 屬於 C_j 若且為若機率 $P(C_i | \mathbf{X})$ 在所有 k 個類別機率 $P(C_k | \mathbf{X})$ 中為最高
- 實際困難: 需要許多機率的起始知識及大量運算成本

單純貝氏判別

- 假設**D**為包含類別標籤的訓練資料值組集，每個值組用**n** 維屬性向量**X** = (x_1, x_2, \dots, x_n)表示
- 假設有個**m**類別 C_1, C_2, \dots, C_m .
- 希望能將一個已知**X**屬於某個類別的事後機率最大化, 也就是最大化 $P(C_i|\mathbf{X})$
- 可以透過貝式理論獲得

$$P(C_i|\mathbf{X}) = \frac{P(\mathbf{X}|C_i)P(C_i)}{P(\mathbf{X})}$$

- 因為對所有的類別**P(X)**為常數, 所以僅需最大化

$$P(C_i|\mathbf{X}) = P(\mathbf{X}|C_i)P(C_i)$$

單純貝氏判別推演

- 簡化假設:假設屬性為條件獨立 (也就是說., 屬性間沒有關係):
$$P(\mathbf{X}|C_i) = \prod_{k=1}^n P(x_k | C_i) = P(x_1 | C_i) \times P(x_2 | C_i) \times \dots \times P(x_n | C_i)$$
- 大大降低計算成本: 僅需計算類別分佈
- 如果 A_k 為類別值屬性, $P(x_k|C_i)$ 則為在 D 中類別 C_i 之屬性 A_k 的值為 x_k 的個數除以 $|C_i, D|$ (C_i 在 D 的值組數目)
- 如果 A_k 為連續值屬性, $P(x_k|C_i)$ 用均值為 μ , 標準差為 σ 的高斯分佈計算得之

$$g(x, \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

$$P(\mathbf{X}|C_i) = g(x_k, \mu_{C_i}, \sigma_{C_i})$$

單純貝氏判別: 訓練資料

類別:

C1:購買電腦 = '是'

C2:購買電腦 = '否'

資料樣本

X = (年齡 = 青年,

收入 = 中,

學生 = 是

信用評等 = 一般)

RID	年齡	收入	學生	信用評等	類別：購買電腦
1	青年	高	否	一般	否
2	青年	高	否	極佳	否
3	中年	高	否	一般	是
4	老年	中	否	一般	是
5	老年	低	是	一般	是
6	老年	低	是	極佳	否
7	中年	低	是	極佳	是
8	青年	中	否	一般	否
9	青年	低	是	一般	是
10	老年	中	是	一般	是
11	青年	中	是	極佳	是
12	中年	中	否	極佳	是
13	中年	高	是	一般	是
14	老年	中	否	極佳	否

單純貝氏判別：範例

$$P(\text{購買電腦} = \text{是}) = 9/14 = 0.643$$

$$P(\text{購買電腦} = \text{否}) = 5/14 = 0.357$$

要計算 $P(X|C_i)$ 我們需要下列條件機率：

$$P(\text{年齡} = \text{青年} | \text{購買電腦} = \text{是}) = 2/9 = 0.222$$

$$P(\text{年齡} = \text{青年} | \text{購買電腦} = \text{否}) = 3/5 = 0.600$$

$$P(\text{收入} = \text{中} | \text{購買電腦} = \text{是}) = 4/9 = 0.444$$

$$P(\text{收入} = \text{中} | \text{購買電腦} = \text{否}) = 2/5 = 0.400$$

$$P(\text{學生} = \text{是} | \text{購買電腦} = \text{是}) = 6/9 = 0.667$$

$$P(\text{學生} = \text{是} | \text{購買電腦} = \text{否}) = 1/5 = 0.200$$

$$P(\text{信用評等} = \text{一般} | \text{購買電腦} = \text{是}) = 6/9 = 0.667$$

$$P(\text{信用評等} = \text{一般} | \text{購買電腦} = \text{否}) = 2/5 = 0.400$$

相同的方法我們可以算出：

$$P(X | \text{購買電腦} = \text{否}) = 0.600 \times 0.400 \times 0.200 \times 0.400 = 0.019$$

接下來我們可以算出 $P(X|C_i)P(C_i)$

$$P(X | \text{購買電腦} = \text{是})P(\text{購買電腦} = \text{是}) = 0.044 \times 0.643 = 0.028$$

$$P(X | \text{購買電腦} = \text{否})P(\text{購買電腦} = \text{否}) = 0.019 \times 0.357 = 0.007$$

因此透過單純貝氏判別我們會預測 X 類別為購買電腦 = 是。

使用上述的機率我們可以得到：

$$\begin{aligned} P(X | \text{購買電腦} = \text{是}) &= P(\text{年齡} = \text{青年} | \text{購買電腦} = \text{是}) \times \\ &\quad P(\text{收入} = \text{中年} | \text{購買電腦} = \text{是}) \times \\ &\quad P(\text{學生} = \text{是} | \text{購買電腦} = \text{是}) \times \\ &\quad P(\text{信用評等} = \text{一般} | \text{購買電腦} = \text{是}) \\ &= 0.222 \times 0.444 \times 0.667 \times 0.667 = 0.044 \end{aligned}$$

避免0-機率問題

- 單純貝氏判別中每個條件機率必須是不為0, 否則預測機率為0

$$P(X | C_i) = \prod_{k=1}^n P(x_k | C_i)$$

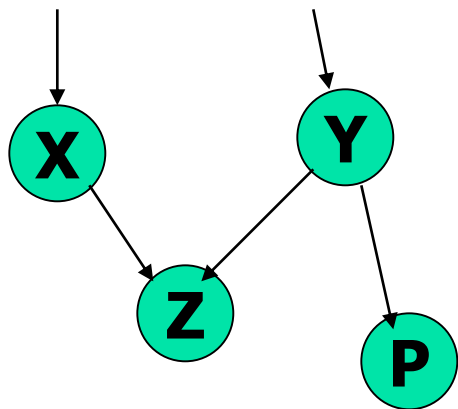
- 例. 假設資料集有 1000 值組, 收入=低 (0), 收入= 中 (990), 收入 = 高 (10),
- 使用拉普拉斯修正 (或拉普拉斯預估)
 - 對所有狀況加上1
Prob(收入=低) = 1/1003
Prob(收入= 中) = 991/1003
Prob(收入 = 高) = 11/1003
 - 修正過的機率非常接近原來機率

單純貝氏判別: 提議

- 優點
 - 容易執行
 - 在大部分案例中都有不錯的結果
- 缺點
 - 假設: 類別條件讀例, 失去正確性
 - 事實上變數間存在相依
- 如何處理相依?
 - 貝式信賴網路

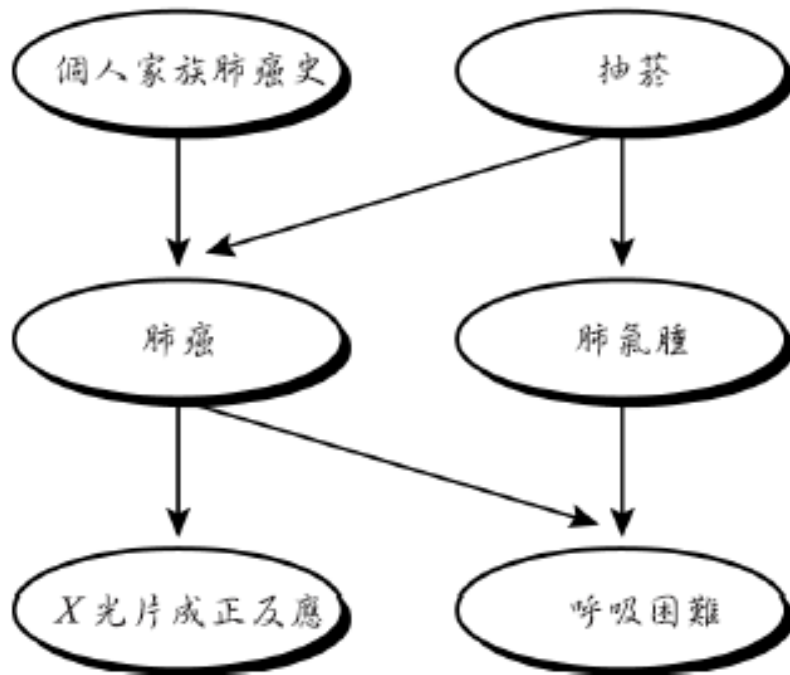
貝氏信賴網路

- 允許定義變數間的類別條件獨立的關係
- 用圖示的方式來表達這種關係
 - 表示變數間相依
 - 給予特定聯合機率分佈



- 節點: 隨機變數
- 箭頭: 相依
- X 與 Y 為 Z 父親, 並且 Y 為 P 父親
- Z 與 P 無相依
- 無迴路

貝氏信賴網路：範例



肺癌變數的條件機率表 (CPT)

	FH, S	$FH, \sim S$	$\sim FH, S$	$\sim FH, \sim S$
LC	0.8	0.5	0.7	0.1
$\sim LC$	0.2	0.5	0.3	0.9

CPT包含每個已知值與其父節點值所有排列組合的條件機率值

假設資料值組 $X=x_1, \dots, x_n$ 而屬性變數為 Y_1, \dots, Y_n ，則從CPT中 X 的結合機率為：

$$P(x_1, \dots, x_n) = \prod_{i=1}^n P(x_i | Parents(Y_i))$$


圖6.11 一個簡單貝氏信賴網路：(a) 一個已知模型用有向無迴路圖代表；(b) 變數肺癌史 (LC) 的條件機率表顯示與其父節點個人家族肺癌史(FH) 與抽菸者 (S)，之間所有可能機率值的排列組合。

貝氏信賴網路

訓練貝氏信賴網路

- 許多方式:
 - 當網路結構與所有變數都是已知: 從CPT中學習
 - 網路結構已知, 某些變數隱藏: 梯度降低 (貪婪爬坡式) 法, 如同類神經網路
 - 網路結構未知, 所有變數都是已知: 搜尋模型空間來重建網路拓撲
 - 未知結構, 所有變數均隱藏: 沒有方法
- 參考 D. Heckerman: Bayesian networks for data mining

第六章.判別與預測

- 何謂判別? 何謂預測?
- 判別與預測議題
- 決策樹判別
- 貝氏判別
- 規則判別 
- 倒傳遞判別
- 支持向量機 (SVM)
- 關連判別
- 偷懶學習 (從鄰居學習)
- 其他判別方法
- 預測
- 正確與錯誤指標
- 組合方法
- 模型選擇
- 總結

使用**IF-THEN**規則進行判別

- 知識以 **IF-THEN** 規則形式表式

R1: IF 年齡青年 AND 學生是 THEN 購買電腦=是

- 規則前提 對應規則結論

- 規則評估:涵蓋率 (coverage) 與正確率 (accuracy)

- n_{covers} =規則R所涵蓋值組的個數

- n_{correct} =規則R能正確判別值組的個數

$\text{coverage}(R) = n_{\text{covers}} / |D|$ /* D: 訓練資料集 */

$\text{accuracy}(R) = n_{\text{correct}} / n_{\text{covers}}$

- 如果超過一個以上的規則被啟動，我們就需要衝突調解策略

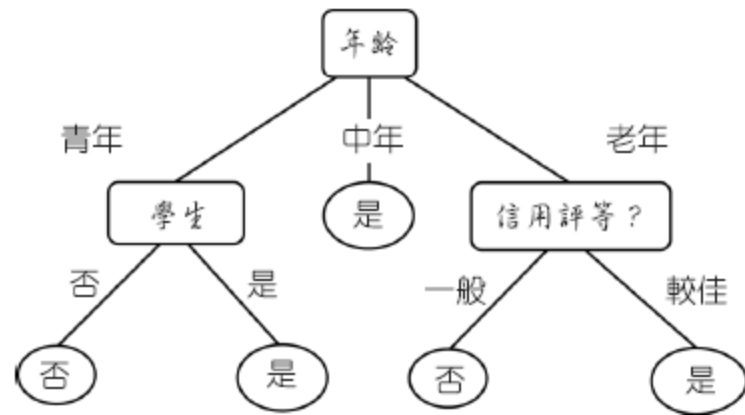
- 大小排序:計算滿足規則的前提大小，並利用這個大小為指標，擁有最大指標值的規則將被執行

- 類別式排序:將規則按照類別種類遞減分組，擁有最大類別個數的規則被執行

- 規則式排序:使用規則排序時，我們產生決策清單 (decision list)，在決策清單中擁有最高指標值的規則才會被執行

從決策樹找出規則

- 規則比大樹狀圖容易了解
- 從根節點到葉節點的方式建立規則
- 從根節點到葉節點的每一對屬性值形成一個結合關係：葉節點包含類別預測
- 規則彼此是互斥 (mutually exclusive) 而且完全 (exhaustive)



- 範例：從決策樹擷取規則

IF 年齡=青年 AND 學生=否 THEN 購買電腦=否

IF 年齡=青年 AND 學生=是 THEN 購買電腦=是

IF 年齡=中年 THEN 購買電腦=是

IF 年齡=老年 AND 信用評等=極佳 THEN 購買電腦=是

IF 年齡=老年 AND 信用評等=一般 THEN 購買電腦=否

從訓練資料擷取規則

- 順序涵蓋法：直接從訓練資料擷取規則
- 典型順序涵蓋法：FOIL, AQ, CN2, RIPPER
- 規則在每個類別順序被產生，從一個類別 C_i 產生規則，我們會希望規則能包含所有 C_i 的值組，並且不包含其他類別的值組
- 步驟：
 - 一次產生一個規則
 - 被這個規則涵蓋的值組會被移除
 - 接下來對剩下的值組繼續產生新的規則直到滿足終止
- 決策樹歸納：同時學習一組規則

Learn-One-Rule?

- 從規則前提為空集合開始: `condition = empty`
- 使用貪婪深度優先法加入新屬性
 - 選擇最能改善規則品質的屬性
- 規則品質指標: 同時考慮正確率與涵蓋率
 - **Foil-gain (in FOIL & RIPPER):**根據訊息獲利來進行一次邏輯規則學習

$$FOIL_Gain = pos' \times (\log_2 \frac{pos'}{pos' + neg'} - \log_2 \frac{pos}{pos + neg})$$


會傾向於包含許多正向值組，並擁有高正確率的規則

- 根據獨立的測試資料值組集進行規則修剪

$$FOIL_Prune(R) = \frac{pos - neg}{pos + neg}$$

Pos代表規則所涵蓋正向值組的個數，**neg**代表規則所涵蓋負向值組的個數。
如果**FOIL_Prune**的值在移除**R**後比較大，我們就會移除規則**R**

第六章.判別與預測

- 何謂判別? 何謂預測?
- 判別與預測議題
- 決策樹判別
- 貝氏判別
- 規則判別
- 倒傳遞判別 
- 支持向量機 (SVM)
- 關連判別
- 偷懶學習 (從鄰居學習)
- 其他判別方法
- 預測
- 正確與錯誤指標
- 組合方法
- 模型選擇
- 總結

利用倒傳遞進行判別

- 倒傳遞:一種類神經網路 (neural network) 的學習方法
- 類神經網路是由心理學家與神經學家為了發展與神經元 (neurons) 類似的計算方式而產生的
- 類神經網路:一組互相連接的輸入輸出單元 (input/output units)，而每個單元都附帶有一個權重 (weight)
- 在學習的過程，權重會被更新以便能正確預測輸入值的類別
- 類神經網路也被稱為連接式學習 (connectionist learning)，因為它將所有的單元連接起來

用類神經網路進行判別

■ 劣勢

- 需要長的訓練時間
- 需要一些經由實驗才能找出最好值的參數項網路拓撲
- 低理解度及一般很難去理解學習權重與隱藏單元背後的意義

■ 優勢

- 對雜訊值的容許度
- 對未知資料樣式的判斷力
- 非常適用於連續值的資料
- 成功的應用於真實世界資料
- 與生俱來的平行化
- 最近有一些方法被用來對類神經網路產生規則

多層前授型網路

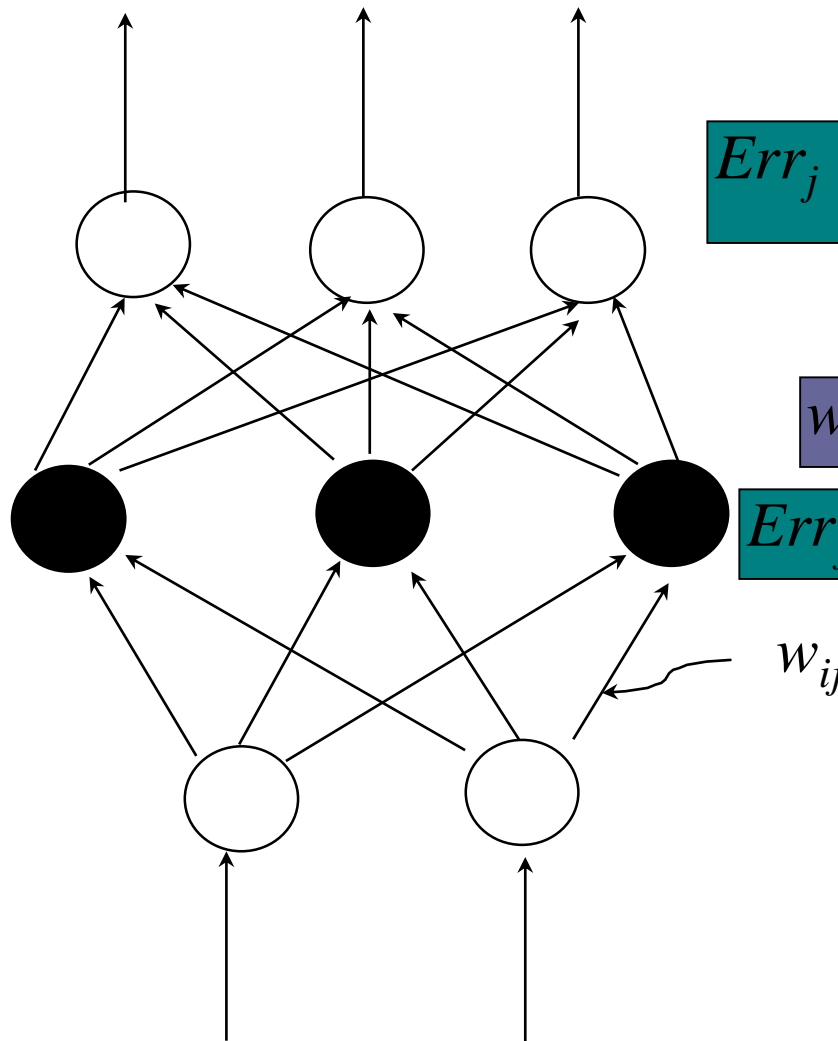
輸出向量

輸出層

隱藏層

輸入層

輸入向量 \mathbf{r} : \mathbf{X}



$$Err_j = O_j(1 - O_j) \sum_k Err_k w_{jk}$$

$$\theta_j = \theta_j + (l) Err_j$$

$$w_{ij} = w_{ij} + (l) Err_j O_i$$

$$Err_j = O_j(1 - O_j)(T_j - O_j)$$

$$O_j = \frac{1}{1 + e^{-I_j}}$$

$$I_j = \sum_i w_{ij} O_i + \theta_j$$

多層前授型網路如何工作？

- 利用值組的每個屬性值進行輸入
- 一個值組的所有屬性值同時輸入至輸入單元，而所有的輸入單元構成輸入層
- 輸入層中的輸入值陪同權重值同步的傳給第二層的單元，而第二層則被稱為隱藏層
- 隱藏層的數目是不固定的，經常是由實驗決定
- 最後一層隱藏層將權重輸出至輸出層 (**output layer**)，然後由輸出層決定值組預測的類別
- 一個網路稱為前授型 (**feed-forward**)，當權重不會繞回至輸入層或是比自己更前面的層
- 從統計的觀點，類神經網路執行一個非線性迴歸：只要能給予足夠的隱藏單元與訓練樣本，多層前授型網路可以相當接近地估計任何函數

設計網路拓撲

- 決定網路拓撲:決定輸入層的單元數、隱藏層的層數、每一層隱藏層的單元數以及輸出層的單元數
- 將輸入值正規化至**0.0**到**1.0**
- 一個輸入單元代表一個輸入值, 每個從**0**開始
- 當類別超過兩個以上, 一般是以一個輸出單元代表一個類別
- 當網路訓練後結果不理想, 經常會利用不同的網路拓撲與不同起始值的權重重新進行訓練


倒傳遞

- 不斷的對每個資料值組的預測類別與實際類別的比較進行學習
- 對於每個值組，依照網路預測類別與實際類別平均誤差值平方 (mean squared error) 的最小化來調整權重
- 調整是倒傳 (backward) 的方式:從輸出層，到最後一層的隱藏層，一直到第一層的隱藏層，所以稱為倒傳遞
- 步驟
 - 起始網路權重與偏差值
 - 向前傳遞輸入值 (套用活化函數)
 - 倒傳遞錯誤值 (對權重與偏差值進行更新)
 - 結束條件 (例當錯誤非常小時)

倒傳遞與理解

- 倒傳遞計算效能:假設有 $|D|$ 的值組與 w 個權重，每次訓練需時間 $O(|D| * w)$,但是最差的狀況訓練次數會是輸入變數個數的冪次方
- 從網路擷取規則: 網路修剪
 - 權重可以被移除，如果移除它不會降低網路的判別正確率
 - 進行連結、單元與活化值的群組
 - 規則根據這些活化值與相對應類別的組合產生，相同的輸入值與活化值的規則也會產生
- 敏感度分析:評估一個輸入變數對於輸出的影響.透過分析所產生的知識可以以規則表示

第六章.判別與預測

- 何謂判別? 何謂預測?
- 判別與預測議題
- 決策樹判別
- 貝氏判別
- 規則判別
- 倒傳遞判別
- 支持向量機 (SVM) 
- 關連判別
- 偷懶學習 (從鄰居學習)
- 其他判別方法
- 預測
- 正確與錯誤指標
- 組合方法
- 模型選擇
- 總結

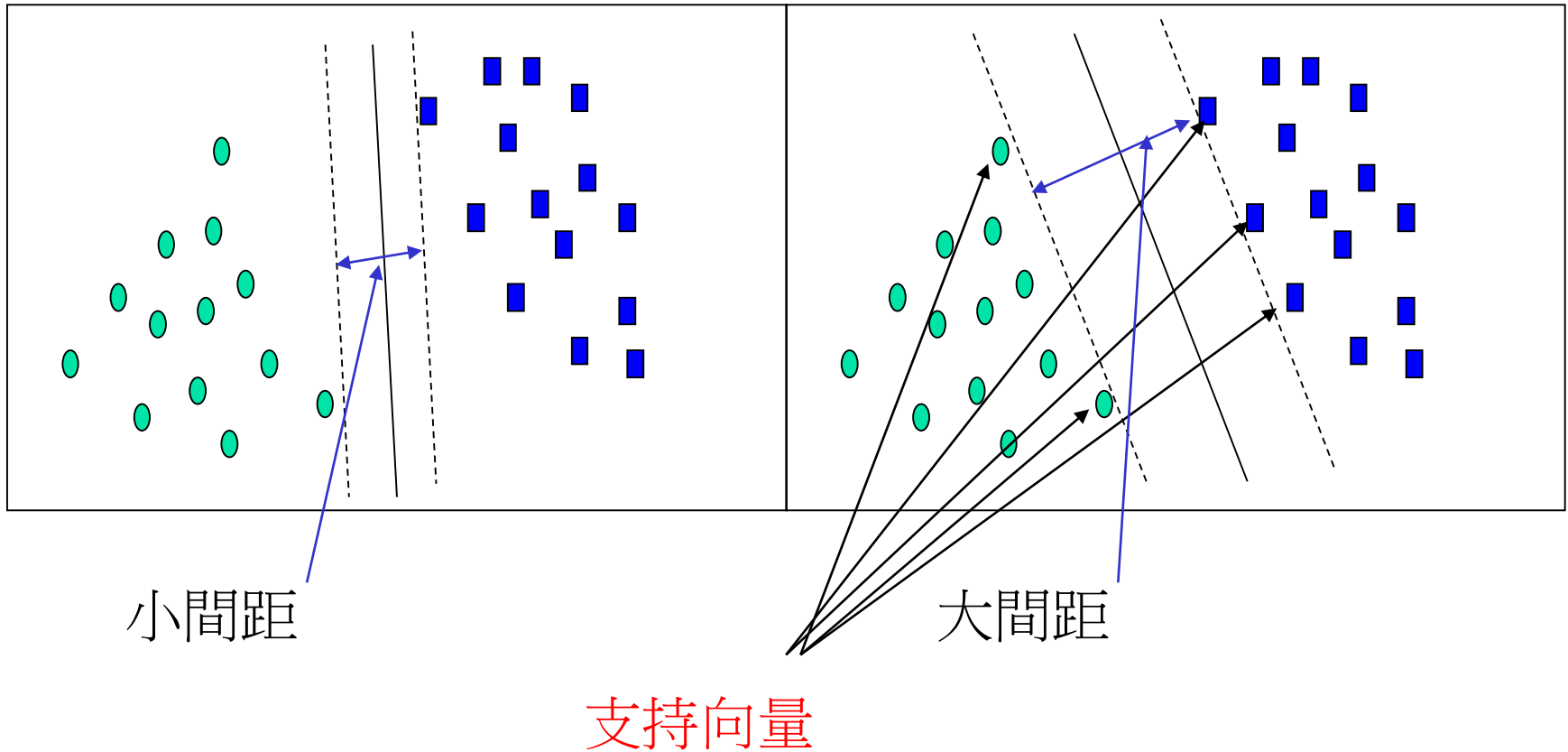
SVM—支持向量機

- 一個在判別線性與非線性資料有效的新方法
- 利用一個非線性轉換將原始資料對應至較高維度資料
- 在較高維度中進行最佳線性超平面分隔
- 透過適當非線性轉換至較高維度，兩種類別的資料都可以透過超平面進行分割
- 支持向量機利用支持向量 (support vectors) 與邊界 (margins) 來尋找超平面

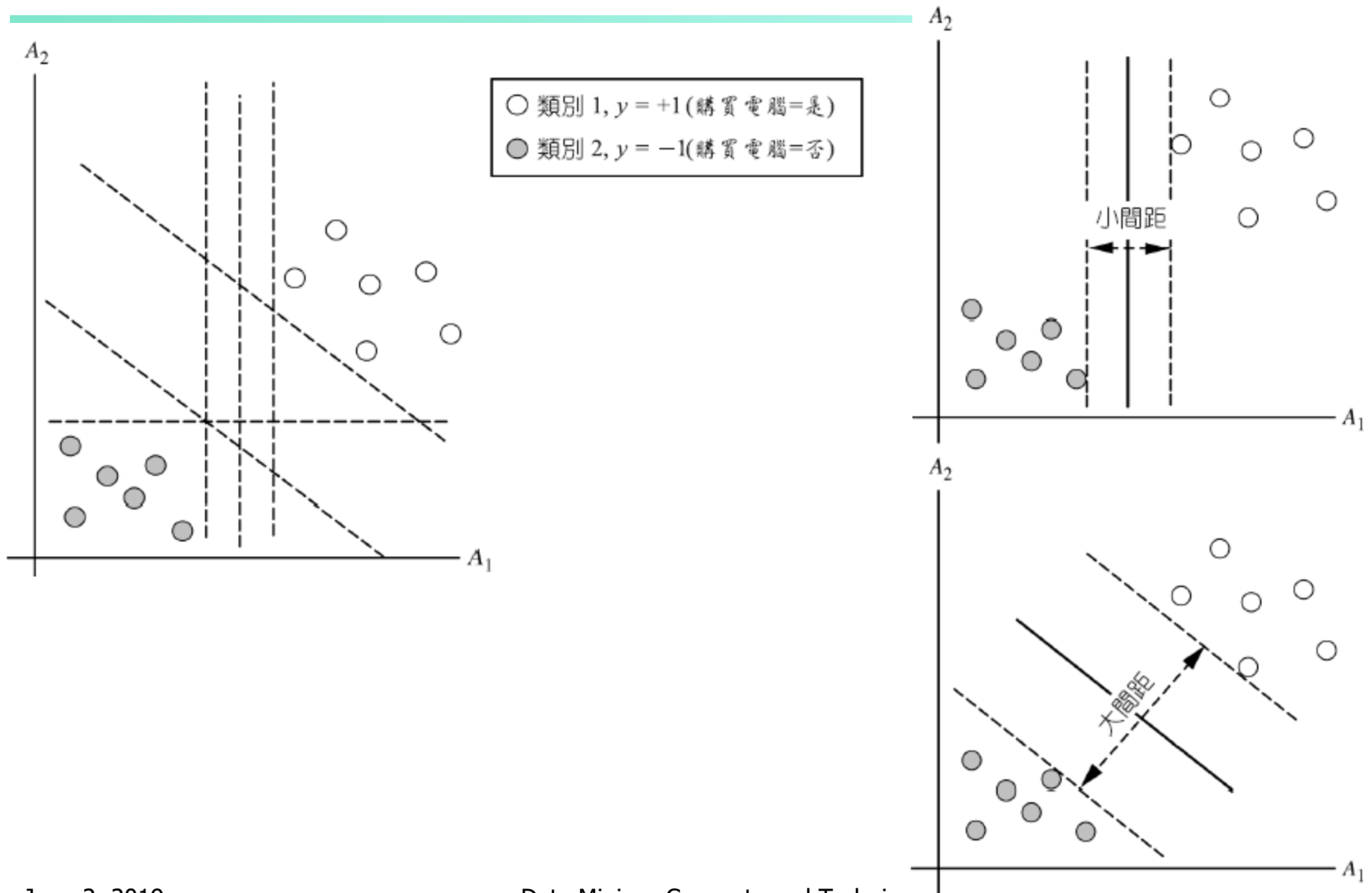
SVM—歷史與應用

- 由Vladimir Vapnik、Bernhard Boser與Isabelle Guyon在1992年提出
- 特色:訓練時間需要很久，但是由於能夠描述複雜非線性決策邊界，支持向量機具有很高的正確率
- 可用於判別或預測
- 應用:
 - 手寫數字辨識
 - 物體辨識
 - 時間序列預測

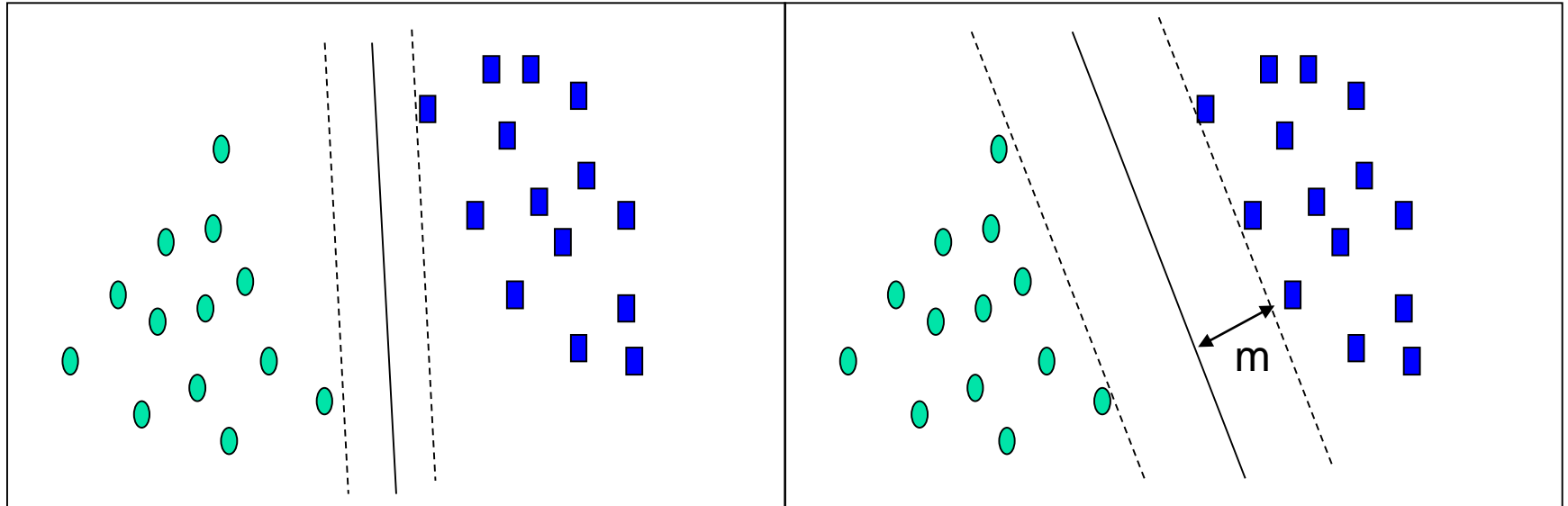
SVM——一般原理



SVM—間距與支持向量



SVM—資料為線性分隔



假設資料集 D 包含值組 $(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_{|D|}, y_{|D|})$, \mathbf{x}_i 代表具有類別 y_i 的值組

分割兩個類別的直線會有無限多條，但是我們希望找到一條能對於未知值組有最小的判別誤差

SVM的方法是尋找最大邊距的超平面(**maximum marginal hyperplane (MMH)**)

SVM—線性可分隔

- 一個線性可分隔的超平面可以表示為

$$\mathbf{W} \bullet \mathbf{X} + b = 0$$

$\mathbf{W} = \{w_1, w_2, \dots, w_n\}$ 代表權重向量， n 代表屬性個數， b 為一個常數(偏差)

- 一個 2-維超平面可以表示為

$$w_0 + w_1 x_1 + w_2 x_2 = 0$$

- 超平面可以用於定義邊界方向：

$$H_1: w_0 + w_1 x_1 + w_2 x_2 \geq 1 \quad \text{for } y_i = +1$$

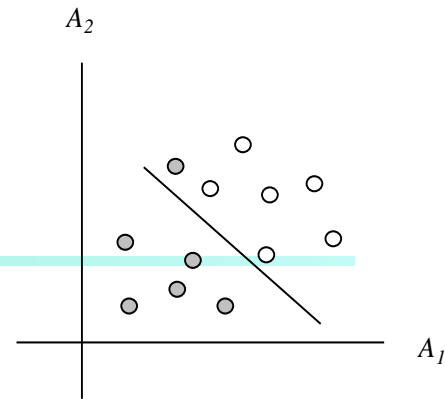
$$H_2: w_0 + w_1 x_1 + w_2 x_2 \leq -1 \quad \text{for } y_i = -1$$

- 落在超平面 H_1 與 H_2 上的值組稱為支持向量(**support vectors**)
- 有條件(凸面)二次方程式最佳化的問題: 二次目標方程式與線性限制 \rightarrow 二次線性規畫 (QP) \rightarrow 拉格朗日公式乘數

為何 **SVM** 對高維度資料有效?

- 複雜度是取決於支持向量的個數而不是資料的維度
- 支持向量是非常重要的訓練值組，因為它們就在最大邊距的超平面 (MMH)
- 如果我們移除其他訓練值組再重新訓練**SVM**，我們會得到相同的最大邊距的超平面
- 支持向量的個數可以用來計算**SVM**的判別期望錯誤率 (expected error rate)，而錯誤率與資料維度無關
- 一個擁有少數支持向量的**SVM**，即使是高維度資料，其通用性仍然很好

SVM—非線性分割



■ 將原始資料轉換至較高維度資料

範例 6.10 將原始資料轉換至較高維度資料 (nonlinear transformation of original input data into a higher dimensional space)

一個 3-維輸入向量 $X = (x_1, x_2, x_3)$ 使用對應 $\phi_1(X) = x_1$, $\phi_2(X) = x_2$, $\phi_3(X) = x_3$, $\phi_4(X) = (x_1)^2$, $\phi_5(X) = x_1x_2$, $\phi_6(X) = x_1x_3$, 對應至 6-維空間 Z 。在新空間的決策超平面 (decision hyperplane) 為 $d(Z) = WZ + b$, W 與 Z 為向量。它為線性，我們解 W 與 b 並將其代入新空間 Z 的決策超平面，會得到一個在原始 3-D 空間的非線性二次多項式

$$\begin{aligned} d(Z) &= w_1x_1 + w_2x_2 + w_3x_3 + w_4(x_1)^2 + w_5x_1x_2 + w_6x_1x_3 + b \\ &= w_1z_1 + w_2z_2 + w_3z_3 + w_4z_4 + w_5z_5 + w_6z_6 + b \end{aligned}$$

■ 在較高維度中進行最佳線性超平面分隔

SVM—核心函數

- 核心函數用來取代點乘積,非線性轉換的點乘積在數學上我們可以用核心函數 (kernel function) $K(\mathbf{X}_i, \mathbf{X}_j)$ 來代表, 也就是說., $K(\mathbf{X}_i, \mathbf{X}_j) = \Phi(\mathbf{X}_i) \cdot \Phi(\mathbf{X}_j)$
- 典型核心函數


Polynomial kernel of degree h : $K(\mathbf{X}_i, \mathbf{X}_j) = (\mathbf{X}_i \cdot \mathbf{X}_j + 1)^h$

Gaussian radial basis function kernel : $K(\mathbf{X}_i, \mathbf{X}_j) = e^{-\|\mathbf{X}_i - \mathbf{X}_j\|^2 / 2\sigma^2}$

Sigmoid kernel : $K(\mathbf{X}_i, \mathbf{X}_j) = \tanh(\kappa \mathbf{X}_i \cdot \mathbf{X}_j - \delta)$

- SVM 也可用於判別多個類別(> 2) 與迴歸分析

第六章.判別與預測

- 何謂判別? 何謂預測?
- 判別與預測議題
- 決策樹判別
- 貝氏判別
- 規則判別
- 倒傳遞判別
- 支持向量機 (SVM)
- 關連判別 
- 偷懶學習 (從鄰居學習)
- 其他判別方法
- 預測
- 正確與錯誤指標
- 組合方法
- 模型選擇
- 總結

關聯判別

- 關聯判別

- 它產生關聯規則並利用關聯規則進行判別
- 尋找頻繁樣式集與類別的有效關聯
- 判別: 根據分析一組像下列規則

$$P_1 \wedge p_2 \dots \wedge p_l \rightarrow "A_{\text{class}} = C" \text{ (信賴度, 支持度)}$$

- 為何有效?

- 由於關聯規則尋找高信賴度多個屬性間的關聯，因此它可以避免一些方法產生的限制。例如決策樹一次只考慮一個屬性
- 在許多研究中顯示，關聯判別比一些傳統的方法要來的正確，如C4.5

典型關聯判別方法

- CBA (Classification By Association: Liu, Hsu & Ma, KDD'98)
 - 探勘下列可能關連規則
 - 條件集 (屬性-屬性值集合) → 類別標籤
 - 將規則按照其信賴度 (confidence) 與支持度 (support) 遞減排序
- CMAR (Classification based on Multiple Association Rules: Li, Han, Pei, ICDM'01)
 - 判別: 對多個規則進行統計分析
- CPAR (Classification based on Predictive Association Rules: Yin & Han, SDM'03)
 - 產生預測規則 (類似FOIL分析)
 - 高效率, 正確率類似 CMAR

第六章.判別與預測

- 何謂判別? 何謂預測?
- 判別與預測議題
- 決策樹判別
- 貝氏判別
- 規則判別
- 倒傳遞判別
- 支持向量機 (SVM)
- 關連判別
- 偷懶學習 (從鄰居學習)
- 其他判別方法
- 預測
- 正確與錯誤指標
- 組合方法
- 模型選擇
- 總結



偷懶與期待學習

- 偷懶與期待學習
 - 偷懶學習 (也稱為事例學習): 將測試資料與訓練資料進行比對，並找出最相似的訓練資料類別
 - 期待學習: 先利用訓練資料建立一個判別模型以便進行測試
- 偷懶: 不會花很多時間在事先利用訓練資料建立判別模型，反而是花很多時間在進行測試資料的判別上

偷懶學習:事例學習

- 事例學習:
 - 儲存值組或事例並直到有新值組或事例產生才進行判別
- 典型方法
 - k-最近鄰判別法
 - 值組或事例貝表示為歐幾里得空間上點.
 - 案例式推理
 - 使用象徵性代表與知識推論


k-最近鄰判別法

- 每個值組代表n-維空間上的一點
- 最近鄰是以歐幾里得距離來表示, $\text{dist}(\mathbf{X}_1, \mathbf{X}_2)$
- 目標函數可以是離散或實數值
- 離散值, **k-NN** 傳回在**k**訓練範例中最接近 \mathbf{x}_q 的最普遍值

案例式推理(CBR)

- CBR:使用問題解答的資料庫來解決新的問題
- CBR將解決問題的值組或案例儲存成複雜的敘述
- 應用:顧客服務 (產品相關的診斷問題),法律判例
- 方法
 - 將解決問題的值組或案例儲存成複雜的敘述 (例., 功能圖)
 - 尋找類似案例, 可整合多個案例
 - 套用背景知識與問題解決策略
- 挑戰
 - 尋找好的相似指標
 - 如何選擇顯著的特性對訓練個案進行索引與有效率的索引技術

第六章.判別與預測

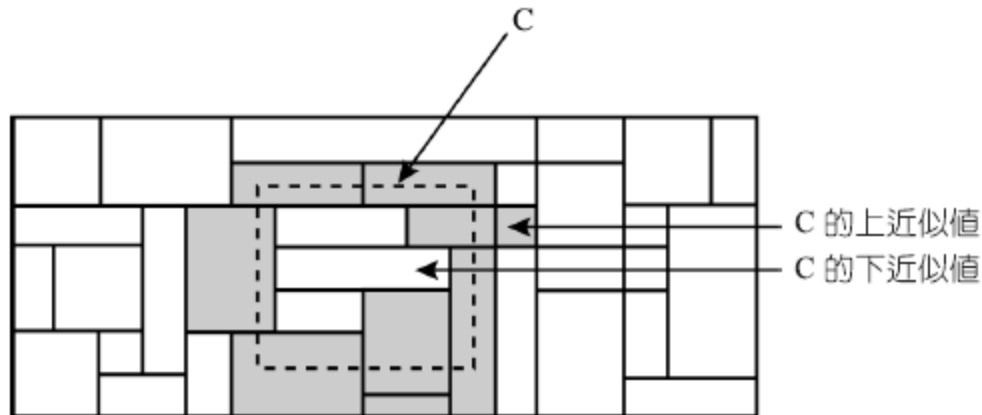
- 何謂判別? 何謂預測?
- 判別與預測議題
- 決策樹判別
- 貝氏判別
- 規則判別
- 倒傳遞判別
- 支持向量機 (SVM)
- 關連判別
- 偷懶學習 (從鄰居學習)
- 其他判別方法 
- 預測
- 正確與錯誤指標
- 組合方法
- 模型選擇
- 總結

基因運算法 (GA)

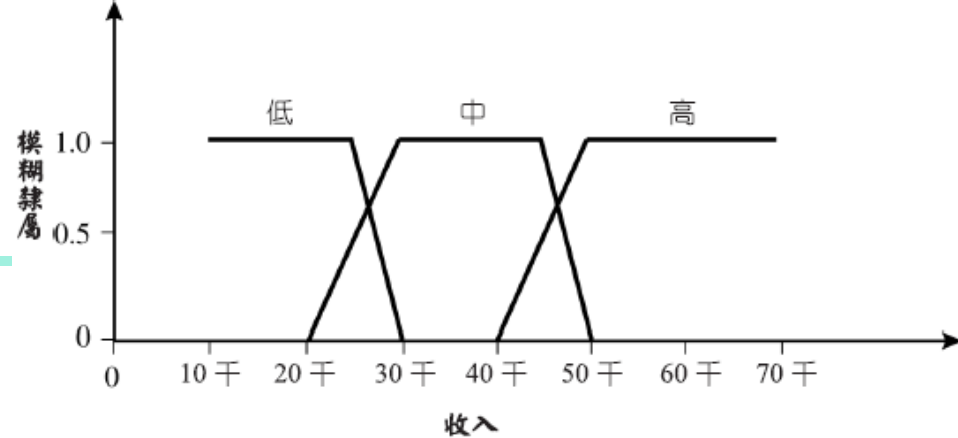
- 基因運算法:套用自然進化 (evolution) 的想法
- 隨機產生一組規則被設為起始群體
 - 每個規則用一串位元表示
 - 例., if A_1 and $\neg A_2$ then C_2 可表示為 100
 - 如果屬性有k個值而且，我們可以用k位元來代表一個屬性值
- 根據自然最適生存的方式，每個規則都會有一個最適值 (fitness value)，接下來利用起始群體產生新的群體，新群體稱為後代(offsprings)
- 規則的最適值通常為訓練資料的正確性
- 後代是經由基因運算如繁衍 (crossover) 與突變 (mutation) 產生
- 整個程序利用新的群體再繼續產生新的群體，一直到新群體的規則滿足預設的限制
- 執行緩慢但是容易被平行化

粗糙集合

- 粗糙集合用於大概定義相等類別
- 給定類別C, 粗糙集合定義兩個近似值: 最小近似值 (類別一定為C的值組) 與最大近似值 (類別可能為C的值組)
- 尋找所有最小屬性集合 (**reducts**) 為 NP-hard 但是區別矩陣 (**discernibility matrix**) 可用於降低計算




模糊集合



- 利用介於0.0到1.0的真實值 來代表隸屬於某個類別的程度 (如同模糊隸屬圖)
- 將屬性值轉換為模糊值
 - 例., 將收入對應至 具有模糊值的類別{低, 中, 高}
- 對於一個新樣本會套用一個以上模糊值
- 每一個規則對類別的隸屬會產生貢獻
- 每個預測類別的真實值會被加總起來

第六章.判別與預測

- 何謂判別? 何謂預測?
- 判別與預測議題
- 決策樹判別
- 貝氏判別
- 規則判別
- 倒傳遞判別
- 支持向量機 (SVM)
- 關連判別
- 偷懶學習 (從鄰居學習)
- 其他判別方法
- 預測 
- 正確與錯誤指標
- 組合方法
- 模型選擇
- 總結

何謂預測?

- (數值) 預測類似於判別
 - 建立模型
 - 對一個輸入值進行一個連續或順序值結果的預測
- 預測不同於判別
 - 判別域內預測類別式類別
 - 預測連續值函數
- 預測主要方法: 迴歸
 - 建立一個或多個獨立或預測變數 (independent or predictor variables) 與相依或回應變數 (dependent or response variable) 間的關係
- 迴歸分析
 - 線性迴歸與多變量線性迴歸分析
 - 非線性迴歸
 - 其他迴歸分析: 通用線性模型, 波松迴歸, log線性模型, 迴歸樹

線性迴歸

- 線性迴歸:包含一個回應變數 y 與一個預測變數 x

$$y = w_0 + w_1 x$$

W_0 為Y截距 (Y-intercept) 與 w_1 為迴歸係數代表上述迴歸直線的斜率

- 最小平方法:預估線性函數錯誤最小化的方法

$$w_1 = \frac{\sum_{i=1}^{|D|} (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{|D|} (x_i - \bar{x})^2} \quad w_0 = \bar{y} - w_1 \bar{x}$$

- 多元線性迴歸:包含超過一個以上的預測變數
 - 訓練資料包含 $|D|$ 點, 以 $(\mathbf{X}_1, y_1), (\mathbf{X}_2, y_2), \dots, (\mathbf{X}_{|D|}, y_{|D|})$ 表示
 - 例. 2-維資料, 可表示為: $y = w_0 + w_1 x_1 + w_2 x_2$
 - 可以使用如SAS、SPSS或S-Plus等軟體
 - 許多非線性函數可轉換成上述方程式

非線性迴歸

- 某些非線性模型可用多項式函數表示
- 透過變數轉換將非線性模型轉換為線性模型, 例,

$$y = w_0 + w_1 x + w_2 x^2 + w_3 x^3$$

利用新變數: $x_2 = x^2$, $x_3 = x^3$ 可轉換為


$$y = w_0 + w_1 x + w_2 x_2 + w_3 x_3$$

- 其他函數如冪次方函數也可轉換為線性模型
- 某些非線性模型無法轉換為線性函數, 如指數加總函數 (sum of exponential)
 - 可以透過最小平方差的方法來進行預估

其他迴歸模型

- 通用線性模型:
 - 依據線性迴歸可用於建立類別回應變數模型的假設
 - 回應變數 y 的變異數不是一個常數，而是變數 y 均值的函數
 - 邏輯斯迴歸:將回應變數與預測變數間的關係視為一線性函數
 - 波松迴歸:如果資料分佈為波松分佈
- 對數線性模型: (類別資料)
 - 用於預估離散多維機率分佈
 - 對資料壓縮 (data compression) 與資料平滑化 (data smoothing) 非常有用
- 迴歸樹與模型樹
 - 用於預測連續值函數

第六章.判別與預測

- 何謂判別? 何謂預測?
- 判別與預測議題
- 決策樹判別
- 貝氏判別
- 規則判別
- 倒傳遞判別
- 支持向量機 (SVM)
- 關連判別
- 偷懶學習 (從鄰居學習)
- 其他判別方法
- 預測
- 正確與錯誤指標 
- 組合方法
- 模型選擇
- 總結

判別正確指標

真實類別	C_1	C_2
	真實正向 錯誤正向	錯誤負向 真實負向

類別	購買電腦=是	購買電腦=否	總計	辨識率(%)
購買電腦=是	6,954	46	7,000	99.34
購買電腦=否	412	2,588	3,000	86.27
總計	7,366	2,634	10,000	95.52

- $\text{acc}(M)$: 為模型M的正確率
 - 模型M的錯誤率為 $= 1 - \text{acc}(M)$
 - 混亂矩陣 $CM_{i,j}$ 表示類別 i 被歸類為類別 j 的值組個數
- 其它正確指標 (例., 癌症檢測)
 - 敏感度 = $t\text{-pos}/\text{pos}$ /*衡量真實正向在正向值組的比例*/
 - 特定性 = $t\text{-neg}/\text{neg}$ /*衡量真實負向在負向值組的比例*/
 - 精確性 = $t\text{-pos}/(t\text{-pos} + f\text{-pos})$
 - 正確率 = 敏感度 * $\text{pos}/(\text{pos} + \text{neg})$ + 特定性 * $\text{neg}/(\text{pos} + \text{neg})$
 - 可用於評估成本 (costs) 與效益 (benefits)

預測錯誤指標

- 衡量預測正確：衡量預測值離已知值多遠
- 損失函數：衡量 y_i 與 y_i' 之間的誤差
 - 絕對誤差： $|y_i - y_i'|$
 - 平方差： $(y_i - y_i')^2$
- 測試錯誤率：測試資料損失函數值的平均

■ 均值絕對誤差：	$\frac{\sum_{i=1}^d y_i - y_i' }{d}$	均值平方差：	$\frac{\sum_{i=1}^d (y_i - y_i')^2}{d}$
■ 相對絕對誤差：	$\frac{\sum_{i=1}^d y_i - y_i' }{\sum_{i=1}^d y_i - \bar{y} }$	相對平方差：	$\frac{\sum_{i=1}^d (y_i - y_i')^2}{\sum_{i=1}^d (y_i - \bar{y})^2}$

當離異值存在時，均值平方差會有極大影響而均值絕對誤差則否。如果我們將均值平方差取平方根，則它被稱為平方根均值平方差

衡量判別或預測模型的正确性 (I)


- 測試(Holdout)法
 - 將資料隨機切成兩個獨立部分
 - 訓練資料 (例., 2/3) 用於建立模型
 - 測試資料 (例., 1/3) 用於評估模型的正确率
 - 隨機取樣: 另一種測試法
 - 進行 k 次測試, 正确率為 k 次的平均
- 交叉驗證 (k -fold, $k = 10$ 為最普遍)
 - 隨機將原始資料分割成 k 個不重疊的部分 (folds), 每個大小相近
 - 當執行第 i 次時, 將 D_i 設為測試資料, 剩下的當作訓練資料。
 - 捨去1 : k folds 當 $k =$ 值設為資料值組的個數
 - 分層交叉驗證: 對原始資料進行分割, 而每個不重疊部分的資料類別分佈大約相似

衡量判別或預測模型的正確性(II)

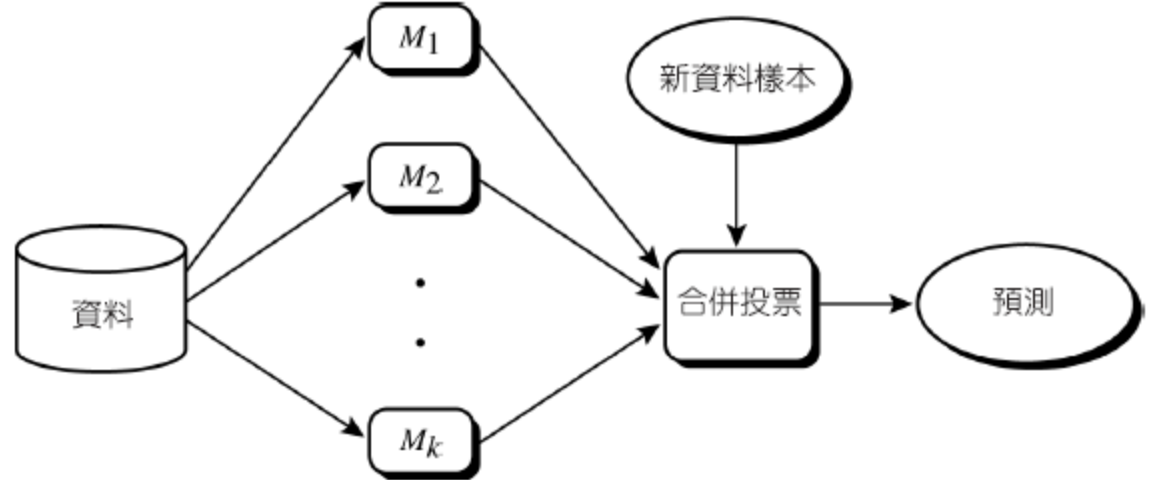
- 自助法
 - 適用於小資料集
 - 使用均勻取出放回 (uniformly with replacement) 的方式從原始資料選取訓練樣本
 - 每一次被選為當作訓練資料的值組都具有相同的機率
- 有許多不同的自助法，而最常用的是.632自助
 - 假設資料包含個 d 值組，我們會利用均勻取出放回的方式從原始資料選擇 d 個值組，這個值組當作訓練資料，而不在訓練資料的值組當作測試資料。假設我們重複許多次這樣的步驟，63.2%的原始資料值組會在自助 (bootstrap) 區域中，而剩下的36.8%的原始資料會在測試資料中(因為 $(1 - 1/d)^d \approx e^{-1} = 0.368$)
 - 重複執行 k 次，則模型的平均正確率為：

$$acc(M) = \sum_{i=1}^k (0.632 \times acc(M_i)_{test_set} + 0.368 \times acc(M_i)_{train_set})$$

第六章.判別與預測

- 何謂判別? 何謂預測?
- 判別與預測議題
- 決策樹判別
- 貝氏判別
- 規則判別
- 倒傳遞判別
- 支持向量機 (SVM)
- 關連判別
- 偷懶學習 (從鄰居學習)
- 其他判別方法
- 預測
- 正確與錯誤指標
- 組合方法 
- 模型選擇
- 總結

組合方法:增加正確率



■ 組合方法

- 組合多個模型來改善整體的正確率
- 組合 k 個學習模型 M_1, M_2, \dots, M_k 來改善整體的正確率

■ 常用組合方法

- 掛袋法:將所有預測模型的平均預測值當作未知值組的預測值
- 推進法:對每個訓練值組設定一個權重
- 組合: 整合一組不同質判別模型

掛袋法：自助法聚合

- 如同：根據多數決診斷
- 訓練
 - 在第 i 次執行時，我們從原始資料 D 中利用均勻取出放回法 (sampled with replacement) 找出 d 的訓練值組 D_i (如同., bootstrap)
 - 利用訓練值組 D_i 產生判別模型 M_i
- 判別：判別未知樣本 X
 - 將未知值組 X 代入判別模型 M_1, M_2, \dots, M_k ，出現最多次數的判別類別會當作未知 X 值組的類別
- 預測：將所有預測模型的平均預測值當作未知值組 X 的預測值
- 正確率
 - 會比在原始資料中僅僅產生一個判別模型的正確率來的高
 - 雜訊資料對它的影響也比較小

推進法

- 如同：參考許多醫生，對每個診斷設權重，結果根據這些具權重診斷組合
- 執行方式？
 - 對每個訓練值組設定一個權重
 - 產生 k 個判別模型進行學習
 - 第 i 次模型 M_i 會對模型判別錯誤的值組進行權重的調整，讓下一個判別模型 M_{i+1} 能對這些判別錯誤的值組能多注意
 - 將未知值組代入 k 個模型，並對這些模型設定不同權重進行判別
- 推進法可延伸至預測連續值
- 與掛袋法比較：正確率一般會比掛袋法高，但具有過適的風險

Adaboost (Freund and Schapire, 1997)


- D個包含類別資料值組 $(\mathbf{X}_1, y_1), \dots, (\mathbf{X}_d, y_d)$,
- I一開始每個值組權重設為 $(1/d)$
- 在第*i*次產生判別模型時
 - 利用取出放回的方法從原始資料產生*d*個值組的訓練資料 D_i
 - 根據每個值組的權重進行選取
 - 利用 D_i 產生判別模型 M_i
 - 將 D_i 中的值組當作測試資料代入判別模型 M_i ,並計算錯誤率
 - 當值組被判斷錯誤時,我們增加該值組的權重,否則我們降低它的權重
- 錯誤率: $\text{err}(\mathbf{X}_j)$ 為值組 \mathbf{X}_j 的錯誤率.判別模型 M_i 的錯誤率計算如下:

$$\text{error}(M_i) = \sum_j^d w_j \times \text{err}(\mathbf{X}_j)$$

- 判別模型 M_i 的權重如下

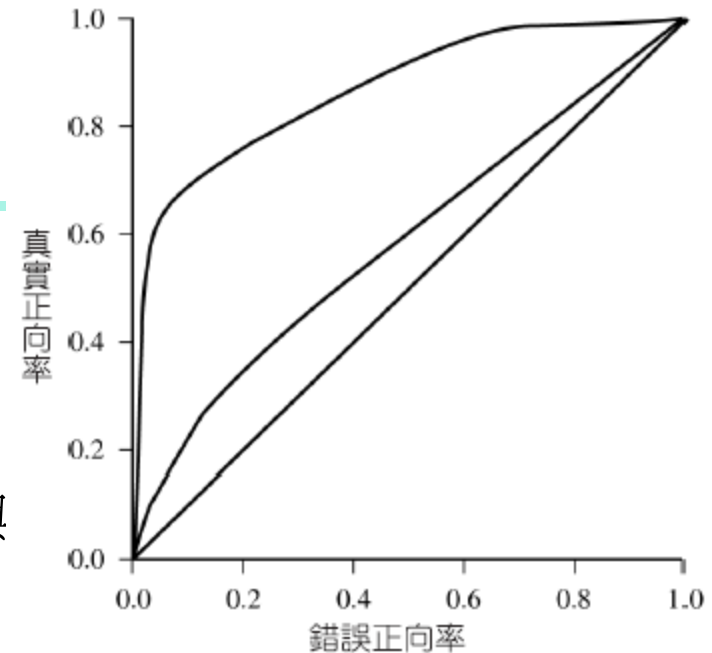
$$\log \frac{1 - \text{error}(M_i)}{\text{error}(M_i)}$$

第六章.判別與預測

- 何謂判別? 何謂預測?
- 判別與預測議題
- 決策樹判別
- 貝氏判別
- 規則判別
- 倒傳遞判別
- 支持向量機 (SVM)
- 關連判別
- 偷懶學習 (從鄰居學習)
- 其他判別方法
- 預測
- 正確與錯誤指標
- 組合方法
- 模型選擇 
- 總結


模型選擇: ROC 曲線

- ROC (Receiver Operating Characteristics) 曲線: 利用圖式比較兩個判別模型
- 源自訊號偵測理論
- 顯示真實正向正確率 (true positive rate) 與錯誤正向正確率 (false positive rate) 的折衷程度
- ROC 曲線下方區域代表模型正確率的指標
- 將測試值組遞減排序: 最屬於真實正向排最上面
- 越接近對角線區域的模型愈不正確 (也就是, 愈接近 0.5 區域)



- 縱軸代表真實正向
- 橫軸代表錯誤正向
- 對角線表示模型預測每個值組真實正向的機率與錯誤正向相同
- 最好的模型區域應為 1.0

第六章.判別與預測

- 何謂判別? 何謂預測?
- 判別與預測議題
- 決策樹判別
- 貝氏判別
- 規則判別
- 倒傳遞判別
- 支持向量機 (SVM)
- 關連判別
- 偷懶學習 (從鄰居學習)
- 其他判別方法
- 預測
- 正確與錯誤指標
- 組合方法
- 模型選擇
- 總結 

總結 (I)

- 判別與預測為兩種資料分析的形式，它們可以建立模型來描述重要類型或預測未來資料趨勢。
- 有效與可量度方法：決策樹歸納,單純貝氏判別,貝氏信賴網路,規則式判別,支持向量機,關聯判別,最近鄰法與案例式推論法,基因運算法, 粗糙集合, 模糊集合。
- 線性、非線性與通用線性迴歸可用於預測，許多非線性的問題可以透過預測變數轉換將其轉換為線性問題。不同於決策樹，迴歸樹與模型樹也可用於預測。

總結 (II)

- 分層交叉驗證 (stratified -fold cross-validation) 法在正確性預估是被推薦使用的，掛袋法與推進法套用不同模型來改善整體正確性.
- 顯著檢定 與 ROC曲線 在模型選擇非常有用
- 有許多比較判別與預測模型的研究
- 沒有一個特定的模型是優於所有的模型
- 在尋求方法時，正確率、訓練時間、一致性、解釋性與可量度性之間的折衷必須加以考量