南 華 大 學

（資訊管理學系）

碩士論文

使用概念漂移偵測之分類法來探勘隨時間變動之資料流

Classification of time--changing data streams based on concept drift detection

研 究 生：張全男

指導教授：邱宏彬

中華民國 九十七年 七月 十四日

南　華　大　學

（資訊管理學系）

碩 士 學 位 論 文

（使用概念漂移偵測之分類法來探勘隨時間變動之資料流）

研究生：（張全男）

經考試合格特此證明

口試委員：

李翔詣

邱宏彬

指導教授：邱宏彬

系主任(所長)：

口試日期：中華民國　九十七年 六月 二十八日

iv

使用概念漂移偵測之分類法來探勘隨時間變動之資料流

學生：張全男　　指導教授：邱宏彬

南　華　大　學　資訊管理學系碩士班

摘　　要

本論文在探討資料流在隨時間變動產生概念漂移的環境下 (Data Stream)之分類(Classification)問題，由於這個連續成長的資料環境下存在著 One-pass 的限制使得我們無法回顧其歷史資料。目前已經有些可應用的演算法，但它們均針對在如何保留資料的時效性而言（意即對目前時間點最有意義），而忽略掉為了保留時效性而付出的嘗試錯誤的成本，與概念穩定時所浪費的維護成本。因此偵測概念漂移之分類法可用於避免上述問題；然而這方法卻因為偵測方法的限制使它在多類別資料偵測上可能導致一些效率上的問題，故我們在統計的基礎上提出一個以卡方檢定為偵測方法的演算法稱為卡方漂移偵測演算法 CDDC(Concept Drift Detection of Chi-Square)，(往後以 CDC、CDDC 交替使用)用以針對漂移修建的觀念將其「屬性值－類別－概念元」觀念更正為「屬性值－概念元」，並實驗該偵測評估方法的有效性；且將其漂移調整案例再作細分。實驗證明在多類別分類問題上能確實能降低因概念元比對所造成的不必要的維護成本，以避免隨類別增加而可能導致的調整成本增加的問題，以及調整案例粗糙所可能造成的成本，達成高速資料流環境下多類別分類概念漂移問題之可行方案。

Classification of time-changing data streams based on concept drift detection.

Student：Chen-Nan Chang Advisors：Dr. Hung-Pin Chiu

Department of Information Management
The M.I.M. Program
Nan-Hua University

ABSTRACT

The present paper flows in the discussion material in changes as necessary produces under the concept drifting environment (DataStream) the classification the question. Because this continuously grows under the material environment has One-pass the limit to cause us to be unable to review its historicalmaterial. At present already some might the application develop the algorithm. How but do they aim at in retain the material the effectiveness for a period of time to say. But neglects for retain the attempt wrong cost which the effectiveness for a period of time pays, is stable with the concept when wastes maintenance cost. Detects classification of the Concept Drifting to be possible to avoid the above question. However this method actually because detects the method the limit to cause it detects in the multi-categories material on possibly causes in some efficiency the question. Therefore we in the statistical foundation proposed as detects the method take the card side examination to develop the algorithm to be called the Chi-Square drifting to detect develops the algorithm. CDDC(Concept Drift Detection of Chi-S-quare). With take aims at the drifting construction the idea it "the attribute value -category-concept unit" the idea correction as "the attribute value-concept unit".

目　錄

第一章 緒論

第一節　研究背景與動機

在資料成長快速的今日，分析方法的運用日益受到重視，也由於資料連續大量的成長，而產生了一些連續性資料的分析問題，其中一種問題便是資料流(data stream)之分類問題。其主要問題在於以往我們建置分類器時，資料已存在於資料庫之中，而資料流環境下卻並非如此，且資料流具有 One-Pass 的環境限制。所謂 One-Pass 意指隨時間經過之每筆原始資料僅能被使用一次，無法再回顧。因此本論文在探討一個隨時間變動之資料流分類問題；它基於一個 One-Pass 的資料來源為前提，且需考慮概念漂移(concept drift)；概念漂移即為隨時間經過，而兩時間區段之資料所代表的意義產生改變稱之。因其連續性資料無法回顧追溯，故演算法必須保留歷史資料概念，而由歷史資料概念來推估未來資料概念。使得現存分類器能以最少的資源而保存最大歷史資料之價值，和涵括對目前最有意義的部份。目前有些演算法如 CVFDT(Concept adapting Very Fast Decision Tree Learner)[13]，WAH(Window Adjustment Heuristic)[23、24]和 DNW[14]等，它們都是由分類器無法適用時，正確率突然降低來獲知漂移的發生，藉以重建分類器，作為演算法在資料概念不穩定的情況下之解決方案。因此也由於這樣的特性，在修正分類器概念的同時其演算法卻增加了許多的成本。

因此有人提出以偵測方式作為概念漂移處理之演算法 CDP-Tree(Concept Drift Probing Tree) [1]。(往後以 CDP 作為簡稱)這種方法它使用統計檢定概念，將兩段時間區段的資料集合（以下稱為資料區塊），化為兩資料區塊之次數資

料的檢定。檢定目標為資料區塊以屬性與屬性值和目標類別所區分之次數資料。每次作單一屬性下之單一屬性值之單一類別的檢定，藉此在分類器錯誤之前，即以次數資料來推估下一區塊是否有異同（漂移），以達到偵測的效果。偵測方法有兩個好處，其一它免除了嘗試錯誤成本，其二它在概念穩定時可節省訓練成本。然而這個演算法卻並不適用在任何情況，它存在著因為偵測檢定方法所導致的限制。我們列舉兩個例子說明這種推估的情況如下：

壹、例子 1-目標類別有 2 類

(1).設類別 1 比率值+類別 2 比率值=1.

(2).則類別 1 比率=1-類別 2 比率值.

(3).若類別 1 比率為 0.3，則可知類別 2 比率等於 1 減 0.3，等於 0.7.

貳、例子 2-目標類別有 4 類

(1).設類別 1 比率+類別 2 比率+類別 3 比率+類別 4 比率= 1.

(2).則類別 1 比率=1-(類別 2 比率+類別 3 比率+類別 4 比率).

(3).若類別 1 比率=0.3；可推得(類別 2 比率+類別 3 比率+類別 4 比率)=0.7

但無法推估類別 2、類別 3、類別 4 之各別單一比率為何.

如上述例子 1，使用 CDP-Tree，則可以以類別 1 之檢定結果依比率來推算出類別 2 之檢定結果。而若使用 CDP-Tree 來檢定例子 2，無論先檢定哪一個類別，都必須再將其他的類別檢定作完才能獲得檢定結果。因此 CDP-Tree 在兩類別資料下，雖可藉由同屬性值之任一類別檢定結果，推估另一類別之檢定結果，但在多類別資料下卻無法以這樣的方式進行。因此當類別增加時 CDP-Tree 必須增加大量的檢定運算才得以判斷兩區塊資料之差異性檢定結果。

第二節　研究目的

在這個研究中我們希望在兩類別資料的分類正確率能與 CDP-Tree 相當，

並且在多類別分類問題上能含括偵測方式的優點來完成多類別的資料分類。找出一個能適用在探勘高速資料串流分類所存在之概念漂移問題之方法。在各種漂移情況下作有效偵測更新分類器概念，以更少的計算複雜度，更廣泛的應用，來減少在重建分類器時所花費的巨額成本，使決策樹可以更貼近事實，作出最佳分類決策，解決這種隨類別遞增而遞增之分類器調整成本增加的問題。

第三節　研究流程

在本論文中，我們使用卡方檢定為其漂移的檢定方法提出卡方漂移偵測演算法 CDC(Concept Drift Detection of Chi-Square)，首先確認目前研究概念漂移的演算法之優缺點，並且比較以偵測為基礎的漂移演算法與本論文發展之方法在兩目標類別與四目標類別的資料之下的正確率與未調整比例等分析結果，藉以比較這兩種方法在案例偵測上是否有所差異（意即在維持正確率之成本是否有所差異），最後分析結果以提出結論與未來研究問題。本論文的研究流程中分為五個階段：問題分析、發展 CDC、測試 CDC、實驗設計、評估分析，順序如圖 1.1。

壹、問題分析：我們研究目前所存在的方法之優缺點，並試著找出可改善之部份，且針對其缺點作改善。

貳、發展 CDC：針對這些缺點與優點考量保留優點，並將其缺點作改良策略。

參、測試 CDC：針對各種功能特性作一般性的單一功能測試，以確定演算法本身沒有邏輯性錯誤。

肆、實驗設計：我們將相似的演算法一起作不同的實驗設計，來比較分析各情況下之結果。

伍、評估分析：根據實驗結果來評估不同演算法之間的優劣與適用時機。

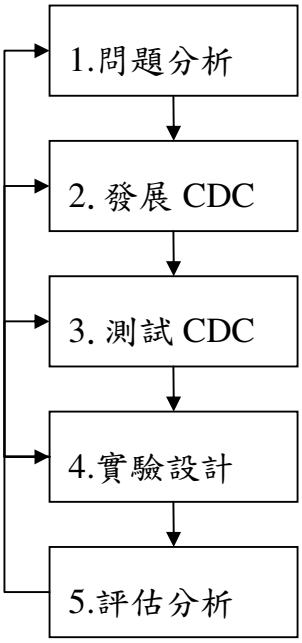


圖 1.1 研究流程圖

第二章、文獻探討

本章將依次介紹分類問題，分類規則演算法、決策樹、資訊理論、以及資料流分類演算法。進一步探討在資料流上的問題與限制，以及目前方法的優劣之處，例如它有 One-Pass 的特性（無法回顧歷史資料），需要快速反應需求，(因其連續不間斷的特性，使其不能為 Off-Line 方式處理)與無窮的儲存需求和概念漂移的問題存在。

第一節　分類(Classification)

壹、分類問題

分類問題是一個存在於資料探勘(Data Mining)中的其中一個重要的工作。它的目的在於將一個事物的類別(Class)的屬性作清楚之定義，將已知的類別樣本（Pre-classified examples）拿來當作是訓練集合(Training Set)，並藉以建造出模型（model）。即為分類器，作為日後未知類別之資料區分之用。達到利用歷史資料對推估未來的目的[21、22]。圖 2.1 為一個分類法的流程圖，圖中訓練資料集合是由歷史資料，或整體資料的一部份所組成。分類演算法扮演資料萃取及建立分類器的角色。分類器含有資料代表的規則，圖中分類器所含有的規則表示為，若職級為經理或年資 20 年含以上者，方可獲得年終獎金。

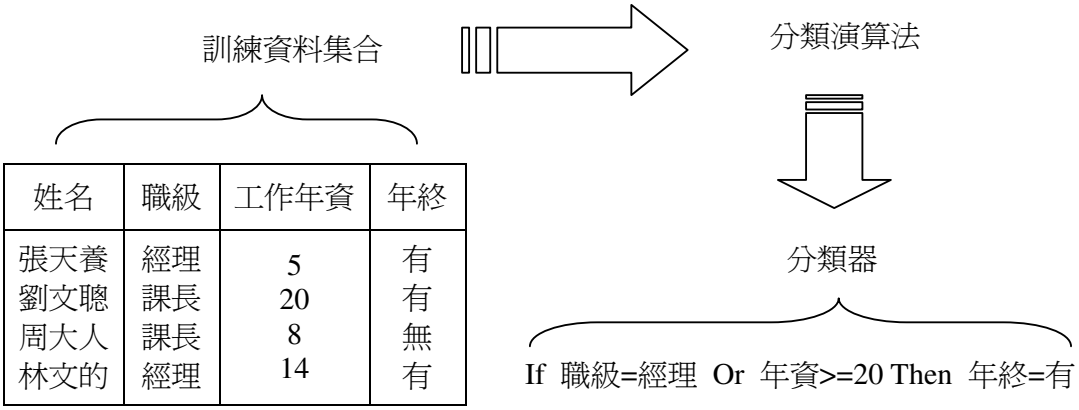


圖 2.1 分類

貳、分類規則學習法

分類規則學習法[15、16、17、18]是根據已知類別的資料集合來學習分類規則，以便針對未分類的資料集合作分類之用。主要的分類規則學習法之技術區分為三大類，分別為決策樹、貝氏網路、其他。圖 2.2 為分類規則學習法常見之技術。

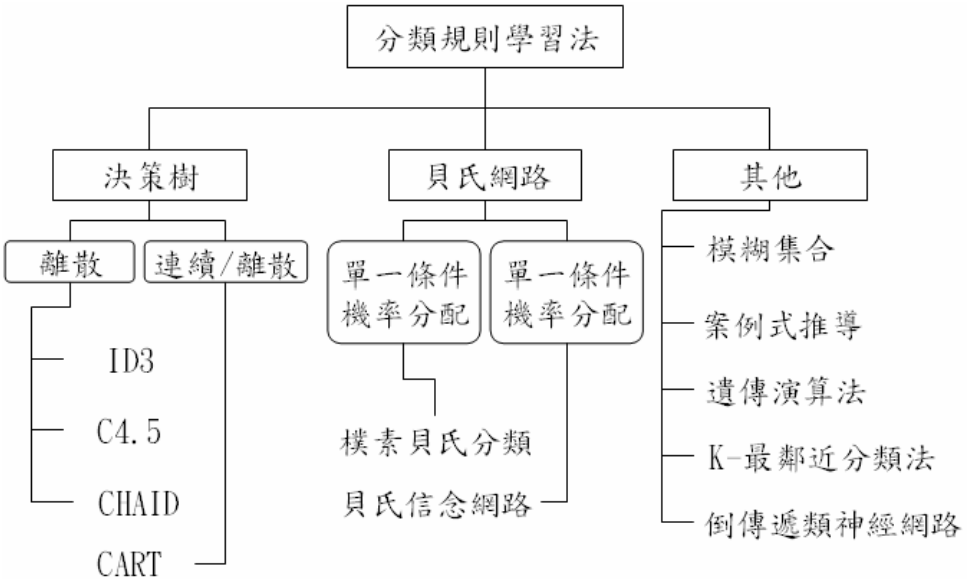


圖 2.2 分類規則學習法技術

參、決策樹演算法

決策樹是一種用樹狀結構來表示規則的方法,每一棵決策樹均含有根節點(Root Node)，子節點(Child Node)與葉節點(Leaf Node)，它們可以用來表示 IF-THEN 條件規則，並應用在多準則決策的問題上面。常見的決策樹演算法主要有 ID3、C4、C4.5、C5、CART、CHAID、QUEST[19、20]。

一、決策樹之建立流程

經由資料歸納分析，依據最能區分資料相異點的屬性，來生成每一節點（Node）。而依據資料逐次生成的節點再依該節點之區分屬性再次區分成多個區域,依次區分至最終獲得歸類類別；依此建立決策樹的方式，以擷取出資料中之規則，以便使用在未來分析資料上。

二、決策樹建立問題

決策樹，乃以純淨度高低作為分裂節點時之依據。而純淨度則按不同應用而有不同公式與方法。其主要概念為目前所在節點的資料可依哪一屬性變數區分，以獲得最大純淨度為依據。且因各屬性變數之屬性值（產生的分岔數）不同，故需可將分岔的子節點純淨度與母節點純淨度相互比較，以評估子節點之分岔是否有其必要性。

三、決策樹應用範例

表 2.1 為一個 ID3 決策樹演算法之購買電腦的範例資料集，演算法經由以往購買記錄來取得此表格，此購買記錄表格即為分類器之訓練集合(Training Set)，而該決策樹則為此訓練集合擷取規則所產生的分類器。購買記錄表格中共有四個屬性可用來區分類別，而類別的種類有「購買」與「不購買」兩類。為了簡化得到的規則，我們必須要以最簡單化原則來選擇決策樹先後所要選用的屬性，以作為決策樹的建立準則，意即先挑選出最能夠區分出資料類別的屬性，並依循著下一個決策樹子節點所屬的資料集合再進行下一個屬性的挑選。需特別注意若為連續性屬性，則應先將之化為不連續型屬性再作處理。一般以資訊理論(Information Gain)或 Gini Index 作為挑選屬性時的選擇工具。這裡選擇使用資訊理論(Information Gain)作為挑選的準則。如圖 2.3 為利用表 2.1 所建造的一棵決策樹，此一決策樹即為分類器，它可對後續資料作為分類之用。

表 2.1　電腦購買資料表

Age	income	Student	Credit_rating	Buy_computer
<=30	High	No	Fair	No
<=30	High	No	Excellent	No
30..40	High	No	Fair	Yes
>40	Medium	No	Fair	Yes
>40	Low	Yes	Fair	Yes
>40	Low	Yes	Excellent	No
30..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
30..40	Medium	No	Excellent	Yes
30..40	High	Yes	Fair	Yes
>40	medium	no	excellent	no

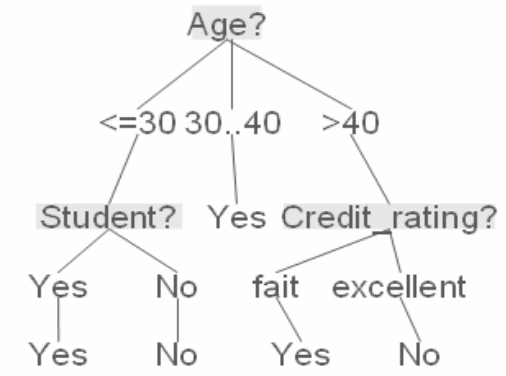


圖 2.3 ID3 決策樹

（一）、事件案例：

1.有一個小於 30 歲的學生則依據決策樹的分類結果，他將被歸類在會購買電腦的這個類別。

2.有一個大於 40 歲的他的信用評價為 Excellent，則他將被歸類在不會購買電腦的這個類別。

註：值得注意的是，在決策樹的規則中並非包含所有的情況，意即容許誤判發生。

第二節　資訊獲利(Information Gain)

在選擇決策樹的分裂屬性的方法之中最常見的為 Information Gain[2]。它根據 Information Gain 作為決策樹分裂節點的依據。這種方法的好處在於可以提供快速的分類，且足夠應付在高速資料流中的需求。

壹、以下是 Information Gain 的計算公式：

$$Gain(a) = I(s_1,...,s_m) - E(a) \qquad \text{公式 2.1}$$

$$\mathrm{I}(s_1,...,s_m) = -\sum_{k-1}^{m} p_k \log(p_k), p_k = \frac{s_k}{s} \qquad \text{公式 2.2}$$

$$E(a) = \sum_{j-1}^{v} \frac{s_{1j} + ... + s_{mj}}{s} I(s_{1j},...,s_{mj}) \qquad \text{公式 2.3}$$

$$\mathrm{I}(s_{1j},...,s_{mj}) = -\sum_{k-1}^{m} p_{kj} \log(p_{kj}) \qquad \text{公式 2.4}$$

設屬性 a 有 v 個屬性值，若每屬性值對應之樣本視為資料子集，則屬性 a 能將事件 S 區分為 v 個子集。sj 表示包含屬性值 aj 的樣本數。若 a

為最佳分裂屬性，則會長出 v 個分支的子集。設 Skj 為每個子集 Sj 裡各類別 Ck 的數量，訓練集合被屬性 a 測試後，求出屬性 a 測試後的期望資訊值如公式 2.3。

Information Gain 的概念是將未分類的資料資訊度減去總和的已分類之子資料資訊度*上該分類的資料資訊度，即測試前減去測試之後的資訊即為 Information Gain，其 $I(s_1,...,s_m)$ 為測試前資訊，而 $E(a)$ 為測試後的資訊，$E(a)$ 愈小表示其亂度愈低，而其 Information Gain 則愈大愈好。Information Gain 之計算茲舉例如下：

一、表 2.2 為一個 15 筆資料集合的屬性次數表(屬性 a 其含有屬性值{v1,v2,..,vn})，設時間 t0~t1 進入的資料區塊有兩個類別 C1、C2 可被屬性 a1 區分為三個屬性值 V1、V2、V3 如表 2.2 ，Information Gain 值計算方式如下：

表 2.2

	a1			a2		
	v1	v2	v3			
c1	3	1	2			
c2	2	4	3			

（一）、表 2.1 根據公式 2.2，s_1 為第 1 個類別的總合與 s_m 第 m 類別的總合, m=2 代入式子 s_1=6, s_2=9，列式如下

$$I(6,9) = (-\frac{6}{15}\log\frac{6}{15}) + (-\frac{9}{15}\log\frac{9}{15}) = 0.292$$

(二)、根據公式 2.3 可列出 $E(a)$ 如下

$$E(a)=\frac{3+2}{15}I(3,2)+\frac{1+4}{15}I(1,4)+\frac{2+3}{15}I(2,3)=\frac{1}{3}I(3,2)+\frac{1}{3}I(1,4)+\frac{1}{3}I(2,3)$$

(三)、同第(一)、項 $I(3,2)$ 、 $I(1,4)$ 、 $I(2,3)$ 可參照公式 2.2 如下

$$I(3,2)=(-\frac{3}{15}\log\frac{2}{15})+(-\frac{3}{15}\log\frac{2}{15})=0.256$$

$$I(1,4)=(-\frac{1}{15}\log\frac{1}{15})+(-\frac{4}{15}\log\frac{4}{15})=0.231$$

$$I(2,3)=(-\frac{2}{15}\log\frac{3}{15})+(-\frac{2}{15}\log\frac{3}{15})=0.256$$

(四)、根據公式 2.1 結果計算如下:

$$Gain(a)=0.292-0.248=0.044$$

第三節　資料流分類演算法

本節介紹一些常見的資料流分類演算法與資料流中的問題；資料流分類演算法是基於一個連續資料流的環境中，來進行資料分析、分類器建置、資料推估分類的演算法。在這個環境中有著三個主要的挑戰分別為：概念漂移、高速資料、記憶需求三種。1.概念漂移：意即資料流隨勒時間變動而資料所代表的意義亦隨之變動，故若使用前段時間所建立之分類器便無法適用在後續的資料中。2.高速資料：演算法必須快速的因應進入的資料，因其需求為連續且快速增加的。3.記憶需求：演算法必須能在有限量的記憶體需求下，快速的處理進入的資料，並保留其歷史資料對目前最有意義的部份。而概念漂移桃戰上大

部份的資料流分類演算法均假設資料是屬於平穩分佈，並未考慮到概念漂移的問題；然而這個假設在真實的環境是不成立的。如表 2.3 中我們可以看到 10 種常見的演算法，其在三個挑戰中之應用的關係。而我們在這三種問題比較上，加上了一個正確率的比較，我們可以看到在正確這欄大部份的演算法都是先遭遇錯誤再予以調整，而 CDP-Tree 在基於一個偵測的基礎上，可以經由偵測方式來得知，其分類器適用下一個區塊與否，藉以避免正確率先遭遇錯誤下降後才回升。簡略介紹幾種分類技術如下：

表 2.3 常見資料流分類演算法之比較

演算法	概念 漂移	高速 資料	記憶 需求	正確 率
VFDT		O	O	Δ
Ensemble based classification	O			Δ
On Demand Classification	O	O	O	Δ
Online Information Network	O			Δ
LW Class Algorithm		O	O	Δ
ANNCAD Algorithm	O			Δ
SCALLOP Algorithm	O			Δ
CVFDT	O	O	O	Δ
DNW	O	O		Δ
CDP-Tree	O	O	O	Y
備註	O：可適用		Δ：下降後調升	

壹、VFDT(Very Fast Decision Trees)：這是一個基於 Hoeffding 樹的決策樹學習系統[3、4]。使用者藉由指定門檻值來解決屬性關係的問題，並逐步除去最不活躍的葉節點及劣質的屬性，以保持一定的記憶空間。而在高速性的問題上則使用批次處理，在低速問題上使用多重掃描來增加準確度。

貳、Ensemble Based Classification：為一種合議式分類器[5、6]。它為了避免單一類別的影響過劇而採取建立許多分類器，再依據各分類器的權重去預測最後的輸出結果藉以得到一定的穩定性。

參、On Demand Classification：On-Demand 分類法使用了兩個階段的作法來執行，第一階段是存放連續的資料流摘要統計來進行分類，因其連續儲存摘要統計故可使用在概念漂移問題上。，第二階段是使用連續的摘要統計來進行分類，且因為是使用摘要統計的資訊來執行，故在新資料到達時可以有很不錯的更新效率[7、8]。

肆、Online Information Network(OLIN)：此演算法係根據分類的錯誤率來調整視窗的大小的機制來執行，如誤差率大表示很可能發生概念漂移，而誤差小表示概念穩定，分類器穩定時便可以縮小視窗以減少訓練成本，而在分類器不穩定時便可以加大視窗藉以求得較為穩定可靠的模型[9]。

伍、LWClass Algorithm：LWClass 演算法以固定的記憶體空間為限制[12]。當一筆記錄進來時則會找尋在記憶體中離它最近的一個項目，且演算法會檢查它的類別標籤，若類別標籤相同則為這個項目增加一次權重，若找不到則減少一個權重以快速因應進入的資料流。且以固定的記憶體空間為限制，當類別標籤權重減到零時這個項目就會從記憶體空間中釋放掉，藉以排除掉一些過時的項目。

陸、ANNCAD Algorithm：一種進行 Haar wavelet 轉換後使用以 Grid-based 為基礎的表示法來找出種類的標籤的演算法[10]。

柒、SCALLOP Algorithm：它以使用者指定的記錄來建立各類別的規則，再依後進的資料作為規則的加強或擴展，亦或是削減，故可針對漂移的資料概念轉換規則的強度[11]。

捌、CVFDT：CVFDT 為 VFDT 的改良版，它主要改良了 VFDT 無法使用在概念漂移資料的缺點[13] 。CVFDT 使用一個固定大小的移動視窗來解決概念漂移的問題。

玖、DNW：DNW 採用在每個區塊上都建立分類器的方式，然後以目前區塊的分類器之三項指標，準確率(Precision)、回收率(Recall)、正確率(Accuracy)來和其他的分類器的指標相比較，以獲得概念變動的情況並藉以調整視窗的大小[14] 。

第四節　CDP-Tree 演算法(Concept Drift Probing Tree)

CDP-Tree 演算法[1]是基於一個偵測的概念上，它將兩區塊（區塊為某時間區段所進入的資料）的概念比對問題轉化成為統計檢定的問題，其檢定的基本單位稱為概念元，而每一個概念元即為在兩區塊間之同屬性、同屬性值、同類

別的一個計次總和單位稱為一個概念元，(如表 2.4 age 屬性下屬性值為”<=30”且類別為”yes”的總計值”2”)並藉以檢定兩個區塊之間的概念元來判定兩區塊是否產生概念漂移的現象，進而採取不同的調整或重建之策略。以表 2.1 之電腦購買資料表為例，將資料區分如表 2.4。設表 2.4 為初始資料之計次表，表中每一個總和次數即為 CDP-Tree 之概念元，依據 CDP-Tree 對概念元的定義，設表 2.4 為初始資料區塊，而與下一區塊作檢定，若兩區塊之概念元無顯著差異，則可推估兩個區塊之資訊量亦沒有顯著差異。故其建立之決策樹亦無顯著差異，因此可以經驗法則，續用舊區塊所建立之分類器，來使用在下一區塊上。它以偵測的方式來解決概念漂移的問題，且由於保留的資料為 Count 資料，且資料量固定，故反應快速，可適用在高速資料下，而保留 Count 資料不保留訓練資料，故無太大的記憶需求。

表 2.4 電腦購買計次表

屬性		age			income			student		c_ratin	
屬性值		<=30	30..40	>40	High	Medium	Low	Yes	No	Fair	Excellent
類別數	YES	2	4	3	2	4	3	6	3	6	3
	NO	3	0	2	2	2	1	1	4	2	3
	總計	5	4	5	4	6	4	7	7	8	6

$$ConD_{D\to D'}(i,j,k,)=\frac{S_{ijk}\Big/S_{ij}-S'_{ijk}\Big/S'_{ij}}{\sqrt{pq\left(\frac{1}{S_{ij}}+\frac{1}{S'_{ij}}\right)}};p=\frac{S_{ijk}+S'_{ijk}}{S_{ij}+S'_{ij}};q=1-p \qquad \text{公式 2.5}$$

公式 2.5 為 CDP-Tree 用以檢定概念元之檢定公式。計算過程舉例如下。設資料區塊 D 有 2 屬性分別各有 3 個屬性值，目標類別為”Yes”與”No”兩類，如表 2.5。以下使用 CDP-Tree 檢定方式檢定其概念元是否有顯著差異以判定是否需要修建決策樹。

表 2.5 兩資料區塊計次表

資料集		D						D'					
屬性		A1			A2			A1			A2		
屬性值		A11	A12	A13	A11	A12	A13	A11	A12	A13	A11	A12	A13
類別數	C1	18	206	22	20	20	206	35	210	17	17	30	215
	C2	84	112	58	48	74	132	76	96	66	66	66	106
	總計	102	318	80	68	94	338	111	306	83	83	96	321

以 5%顯著水準舉例 t 值為 1.96，依公式 2.5 計算下列檢定結果如下：

$$ConD_{D\to D'}(1,1,1)=\frac{18/102-35/111}{\sqrt{pq\left(\frac{1}{102}+\frac{1}{111}\right)}} \quad ; \quad P=\frac{18+35}{102+111}=0.2488 \quad ; \quad q=1-0.2488=0.7512$$

$$ConD_{D\to D'}(1,1,1)=\frac{-0.13884}{0.059298}=-2.3414$$

如式子所列，分子：為區塊 D 的屬性 A1 的屬性值 A11 下之類別 C1 與 D-A1-A11 所有類別的加總比，減去區塊 D' 的屬性 A1 的屬性值 A11 下之類別 C1 與 D-A1-A11 所有類別的加總比。分母：為兩區塊對應屬性值之總和的倒數相加乘上 pq。p:為兩區塊對應屬性值之概念元值加總與概念元所在之屬性值加總比。q:為 p-1。

$ConD_{D\to D'}(1,1,1)=|-2.3414|<|1.96|$ 有顯著差異，可推得(1,1,2)亦有顯著差異。

$ConD_{D\to D'}(1,2,1)=|-1.0192|<|1.96|$ 無顯著差異，可推得(1,2,2)亦無顯著差異。

$ConD_{D\to D'}(1,3,1)=|1.0499|<|1.96|$ 無顯著差異，可推得(1,3,2)亦無顯著差異。

$ConD_{D\to D'}(2,1,1)=|1.2693|<|1.96|$ 無顯著差異，可推得(2,1,2)亦無顯著差異。

$ConD_{D\to D'}(2,2,1)=|-1.5608|<|1.96|$ 無顯著差異，可推得(2,2,2)亦無顯著差異。

$ConD_{D\to D'}(2,3,1)=|-1.611|<|1.96|$ 無顯著差異，可推得(2,3,2)亦無顯著差異。

經由上列依 CDP-Tree 檢定可得 $ConD_{D\to D'}(1,1,1)$ 時有顯著差異，故 CDP-Tree 推估 $ConD_{D\to D'}(1,1,2)$ 亦有顯著差異。此時決策樹需要針對 Bt+1（新的區塊之資料）之屬性 a1 下之屬性值 a11 之資料進行修建。

第五節　統計檢定-卡方齊一性檢定

由於 CDP-Tree 將漂移問題轉化為檢定問題，故我們可以使用檢定方式以作為應用之方法。本節介紹一個統計的齊一性檢定，它是用來檢定兩個或以上的母體是否有某一特性的分配（意即各類別分配比例在各分層與母體是否一致）是否相近，並代一個例子讓我們了解卡方檢定的運算過程及其特性。

壹、齊一性檢定統計量:

$$X^2=\sum_{i=1}^{r}\sum_{j=1}^{c}\frac{(O_{ij}-\hat{E}_{ij})^2}{\hat{E}_{ij}} \qquad \text{公式 2.6}$$

1. If $x^2 > x^2(c-1)(r-1),a$ ，Then Reject H_0 。
2. If $x^2 \le x^2(c-1)(r-1),a$ ，Then Don't Reject H_0 。

註：r：橫列個數，c：縱行個數，Oij：樣本觀察次數，$\hat{E}_{ij}$：估計期望次數，自由度為(r-1)(c-1)。

貳、齊一性檢定例子：

假設如表 2.6 想知道甲校與乙校在期中考成績上各成績分佈上是否一樣？

現在假設在甲校抽出 600 個學生，在乙校抽出 750 個學生，檢定過程與結果如下：

表 2.6 齊一性檢定範例資料

成績等級	甲校	乙校	合計次數
大於 79	200	300	500
70~79	150	200	350
60~69	150	150	300
小於 60	100	100	200
合計次數	n1=600	n2=750	n=1,350

一、設立假設：

H_0：兩間學校的成績分佈比例一樣。

H_1：兩間學校的成績分佈不一樣。

二、選擇檢定統計量：

以卡方分配之齊一性檢定為檢定方式。

三、決定拒絕範圍跟接受範圍：

顯著水準 α=0.05

自由度 df=(r-1)(c-1)=(4-1)(2-1)=3

查表可得，χ^2檢定的臨界值是 $\chi^2_{(4-1)(2-1),0.05}=7.815$

四、檢定統計量與臨界值相比較：

$$\hat{E}_{11}=600\times\frac{500}{1,350}=222.22 \qquad \hat{E}_{12}=750\times\frac{500}{1,350}=277.78$$

$$\hat{E}_{21}=600\times\frac{350}{1,350}=155.56 \qquad \hat{E}_{22}=750\times\frac{350}{1,350}=194.44$$

$$\hat{E}_{31}=600\times\frac{300}{1,350}=133.33 \qquad \hat{E}_{32}=750\times\frac{300}{1,350}=166.67$$

$$\hat{E}_{41}=600\times\frac{200}{1,350}=88.89 \qquad \hat{E}_{42}=750\times\frac{200}{1,350}=111.11$$

成績等級	甲校	乙校	合計次數
大於 79	200(222.22)	300(277.28)	500
70~79	150(155.56)	200(194.44)	350
60~69	150(133.33)	150(166.67)	300
小於 60	100(88.89)	100(111.11)	200
合計次數	n1=600	n2=750	n=1,350

$$\chi^2=\frac{(200-222.22)^2}{222.22}+\frac{(150-155.56)^2}{155.56}+\frac{(150-133.33)^2}{133.33}+\frac{(100-88.89)^2}{88.89}$$
$$+\frac{(300-277.28)^2}{277.28}+\frac{(200-194.44)^2}{194.44}+\frac{(150-166.67)^2}{166.67}+\frac{(100-111.11)^2}{111.11}$$
$$=2.22+0.20+2.08+1.39+1.78+0.16+1.67+1.11=10.61$$

五、檢定結論

檢定的統計量 $\chi^2 = 10.61$ 大於臨界值 $\chi^2(4-1)(2-1),0.05 = 7.815$

不接受虛無假設 H_0。結論為兩間學校成績的四個等級下分佈比例不一樣。

第三章、研究方法

本章第一節介紹漂移的形式，藉以了解概念漂移的樣式。並且在第二節以 CDC 與 CDP-Tree 在兩類與四類的檢定來點出問題處。第三節為概念漂移案例與修建方法。第四節將介紹卡方漂移偵測演算法 CDC 與演算法之程式流程圖。第五節以一個完整的例子來帶過 CDC 的整個流程。

第一節　概念漂移形式(Concept Drift Type)

本節依照概念漂移的程度與情況區分為兩大類，並且呈現圖形概念藉以說明在資料集合中之真實情形，概念漂移演算法可以依據這些漂移的形式來調整分類器。(漂移的形式有下列幾種，然而重點著重於演算法是否可以適用在這些不同的形式下，仍能偵測漂移情況，且以更低的成本取得相似的分類器正確率)。

壹、依類別的漂移程度區分

一、概念穩定：如圖 3.1 所示在圖中之黑色圓圈表示為負向例，而白色圓圈表示為正例，橫線則為最佳的決策線。而圖 3.1 中(a)為 t0~t1 時間到達的資料,而(b)則為 t1~t2 時間內到達的資料。其兩段區域間之正負例均沒有巨變，最佳決策線仍保持在原來位置，表示用(a)資料區塊所

建立之分類器仍能適用在(b)的區段中，因為資料所代表的概念並沒有偏移。

二、概念漂移：圖 3.1 中我們可以看到在圖(b)至圖(c)之黑色圓圈(正例部份)有些許轉移成白色圓圈，此時由圖(b)所建立之分類器的正確率在圖(c)時會有正確率逐漸下降的情形，若不修建分類器將可能使錯誤率愈來愈高。

三、概念轉移：由圖 3.1 我們可以看到在(c)到(d)部份的正例與負例有明顯的差別。且與(c)部份時的情況完全相反，若我們繼續使用(c)所建立的決策樹，來對(d)推論，則所有正例部份將被誤判為反例，而所有反例部份則會被誤判為正例，此時分類器的正確率將急劇下滑。

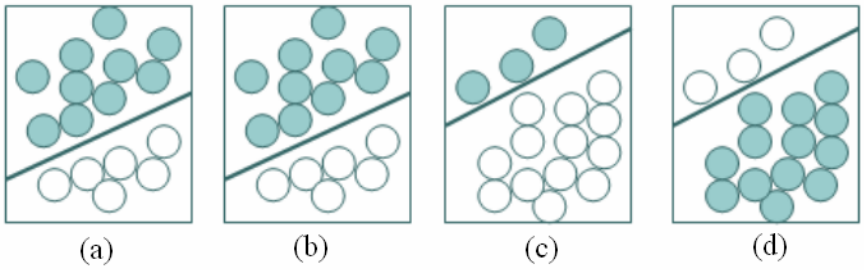
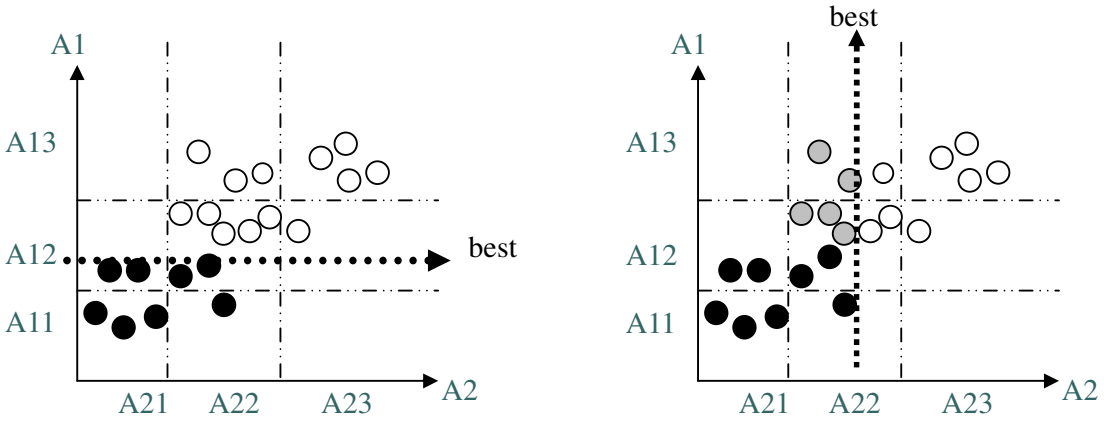


圖 3.1.

貳、依類別的漂移情況區分：

一、單向漂移：如圖 3.2a 部份為 t0～t1 時間到達的資料，而圖 3.2b 部份是由 t1~t2 時間到達的資料，讓我們看圖 3.2a 的部份在資料中白色空

心圈代表正例，而黑色實心圈代表負例，且在圖 3.2a 中的最佳決策分界線為水平線箭頭，而圖 3.2b 部份的最佳決策分界線則為垂直線箭頭，它將正負向例區分開來，若我們使用圖 3.2a 的部份所建立的分類器對圖 3.2b 部份的區塊作使用的結果將會是在圖 3.2b 的灰色部份的負向例將被誤判為圖 3.2a 部份的正向例，類別單向的改變所引起的漂移，此稱之為單向漂移。



a)單向漂移區塊 1　　b) 單向漂移區塊 2

圖 3.2. 單向漂移之資料屬性類別分佈圖，X 軸為屬性 A2 的三個屬性值 A21、A22、A23;Y 軸則為屬性 A1 的三個屬性值 A11、A12,A13;如圖 a)為 t0~t1 時間區段所進入的資料，白色圈代表正例，黑色圈代表負例。而圖 b)則為 t1~t2 時間區段所進入的資料。Best 為其最佳決策分界線。

表 3.1. 單向漂移之計次資料；依圖 3.2 之兩時間區段資料(區塊)依照其資料之屬性與其屬性所屬之屬性值，與目標類別區分如下：

a) 單向漂移區塊 1 計次資料

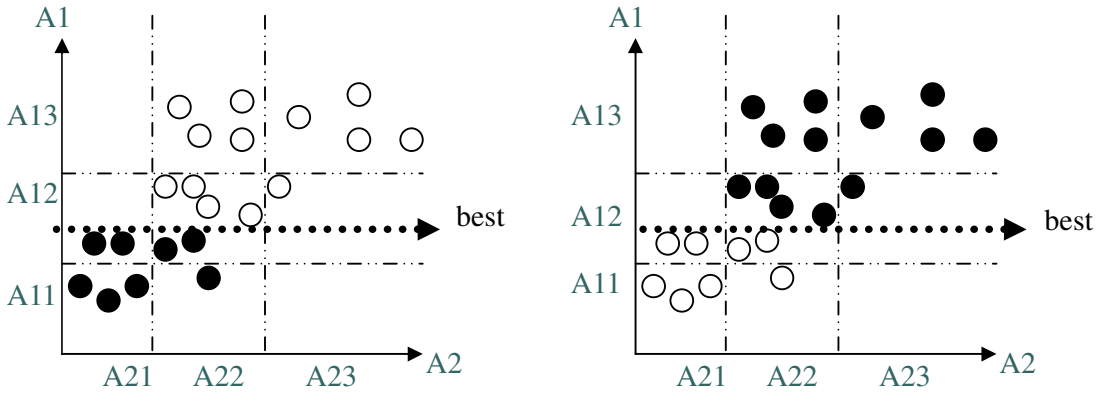
	Bt					
	A1			A2		
	A11	A12	A13	A21	A22	A23
C1 ○	0	6	7	0	8	5
C2 ●	4	4	0	5	3	0

b) 單向漂移區塊 2 計次資料

	Bt+1					
	A1			A2		
	A11	A12	A13	A21	A22	A23
C1 ○	0	3	5	0	3	5
C2 ●	4	7	2	5	8	0

如表 3.1(a)、表 3.1(b)可以對照在圖 3.2 的兩維資料分佈下類別的分佈情況，如圖 3.2 兩區塊的X軸之屬性 3 均為 5 個正向例，而相對的在計次表上也有 5 個正向例;即可依此看出在屬性 A2 的屬性值 A23 下這兩計次表的次數均為 C1=5,C2=0。因此可表示在此 A23 之區域沒有漂移。

二、循環漂移:設如圖 3.3a 所示為 t0~t1 時間區間所到達的資料,而圖 3.3b 則為 t1~t2 所到達的資料，X 軸為屬性 A2；Y 軸為屬性 A1，讓我們看圖 3.3a 的資料中白色空心圈代表正向例，而黑色實心圈表示為負向例,且在圖 3.3a 的部份中的最佳決策分界線與圖 3.3b 部份的最佳決策分界線相同，然而兩類比例卻互換。若我們使用圖 3.3a 部份所建立的分類器來對圖 3.3b 的部份作使用，則會誤將正向例部份全部誤判為負向例，而負向例部份全部誤判為正向例，類別的交替改變，此為循環漂移；表 3.2a 與表 3.2b 為圖 3.3a 與圖 3.3b 之計次資料。



a) 循環漂移區塊 1　　b) 循環漂移區塊 2

圖 3.3 循環漂移之資料屬性類別分佈圖，X 軸為屬性 A2 的三個屬性值 A21、A22、A23;Y 軸則為屬性 A1 的三個屬性值 A11、A12、A13;如圖 a)為 t0~t1 時間區段所進入的資料，白色圈代表正例，黑色圈代表負例。而圖 b)則為 t1~t2 時間區段所進入的資料。Best 為其最佳決策分界線。

表 3.2.　循環漂移之計次資料；依圖 3.3 之兩時間區段資料(區塊)依照其資料之屬性與其屬性所屬之屬性值，與目標類別區分如下：

a) 循環漂移區塊 1 之計次資料

	Bt					
	A1			A2		
	A11	A12	A13	A21	A22	A23
C1 ○		5	8		8	5
C2 ●	4	4		5	3	

b) 循環漂移區塊 2 之計次資料

	Bt+1					
	A1			A2		
	A11	A12	A13	A21	A22	A23
C1 ○	4	4		5	3	
C2 ●		5	8		8	5

第二節　卡方檢定χ^2與 CDP-Tree 檢定

本節依卡方檢定與 CDP-Tree 檢定方法各在兩類別資料與四類別資料上作比較，並藉此點出 CDP-Tree 在多類別資料檢定上之問題。

壹、兩類別資料檢定

一、CDP-Tree 檢定方式

設表 3.3 為兩不同時間區段到達之資料 Bt 和 Bt+1，以第二章介紹之公式 2.1，檢定表 3.3 之各概念元，我們可以看到在這個例子中的兩個概念元之檢定值是相同的(0.633358=0.633358)，故在兩類別區塊資料之同屬性值下，可以任一個概念元檢定結果推估同屬性值下之另一概念元檢定結果，以此例而言 CDP-Tree 只需作三次檢定，且可推估另一檢定結果達百分之百。

表 3.3 兩資料區塊之計數資料

資料區塊		Bt			Bt+1		
屬性值		a11	a12	a13	a11	a12	a13
類別	C1	192	41	13	208	42	12
	C2	33	142	79	42	133	63
總合	SUM	225	183	92	250	175	75

(一)、CDP-Tree 之檢定運算如下：

$$ConD_{D\to D'}(1,1,1)=\frac{192/225-208/250}{\sqrt{0.8421\,(1-0.8421)\left(\frac{1}{225}+\frac{1}{250}\right)}}=0.633358$$

$$ConD_{D\to D'}(1,1,2)=\frac{33/225-42/250}{\sqrt{0.1579\,(1-0.1579)\left(\frac{1}{225}+\frac{1}{250}\right)}}=0.633358$$

二、卡方-齊一性檢定：

以表 3.3 的兩區塊之屬性值 a11 為例，先取出欲檢定之屬性值欄位，與下一區塊相對應的屬性值欄如表 3.4。卡方-齊一性檢定在這個例子裡需要三次檢定運算，以每屬性值（每欄）一次運算。相較 CDP-Tree 檢定方法似乎沒有較具優勢，然而若這個例子的類別增加為四時，由於此時 CDP-Tree 檢定方式無法以屬性值內之任一個類別檢定結果來推估其他類別的檢定結果，故 CDP-Tree 檢定方法則需要 4 類別 x3 屬性值=12 次的檢定才能評估這兩區塊之屬性值底下，資料有沒有顯著差異，因此我們的目的是在上述問題上設計一個可適用於多類別資料的概念漂移偵測法之可行方案；由於想知道 Bt-Value1 與 Bt+1-Value1 在兩個類別的分佈是否有差異，故可設 H0：沒有顯著差異，H1：有顯著差異，若檢定值大於臨界值，則表示有顯著差異，則無法拒絕 H1。

表 3.4 觀察次數表

	Bt-value1	Bt+1-Value1	總合
類別 C1	192	208	400
類別 C2	33	42	75
總合	225	250	475

(一)、計算期望值。

$$\hat{E}11 = \frac{225 * 400}{475} = 189.473 \qquad \hat{E}21 = \frac{250 * 400}{475} = 210.526$$

$$\hat{E}12 = \frac{225 * 75}{475} = 35.526 \qquad \hat{E}22 = \frac{250 * 75}{475} = 39.473$$

(二)、計算檢定量。

$$\frac{(192-189.473)^2}{189.473}=0.033684211 \qquad \frac{(208-210.526)^2}{210.526}=0.030315789$$

$$\frac{(33-35.526)^2}{35.526}=0.179649123 \qquad \frac{(42-39.474)^2}{39.474}=0.161684211$$

(三)、依臨界值與統計量計算出檢定結果。

$$\chi^2 = 0.048020822+0.05021606 +0.014493557 +0.01515612 =0.12789$$

臨界值=3.84146(查表(2-1)(2-1)$_{0.05}$)

0.12789<3.84146 無法拒絕 H1(意即沒有顯著差異)

貳、四類別資料檢定

一、CDP-Tree 檢定方式

以下四個式子分別為區塊 1、2 之概念元的檢定運算，從檢定值可以看到在這個例子中的四個檢定值是不相同的,故在四類別區塊資料之同屬性值之下,無法以任一個類別檢定結果推估在另一個類別之檢定結果。以此例來講 CDP-Tree 需作 3X4=12 次檢定運算。檢定結果僅有第三屬性值的第四類別有顯著差異（統計量超過 1.96)。然而我們看到實際在表 3.5 的計數資料中，其第三屬性值的第四類別的計數都是 58，顯示 CDP-Tree 在這個例子中的檢定結果,對類別層下並沒有檢定意義(無法檢定其某類別與其相對應的類別間的計數資料)。

表 3.5 四資料區塊之計數資料							
資料區塊		Bt			Bt+1		
屬性值		a11	a12	a13	a11	a12	a13
類別	C1	192	41	13	193	34	32
	C2	33	142	79	33	125	120
	C3	18	216	12	18	220	22
	C4	74	122	58	53	92	58
總合	SUM	317	521	162	297	471	232

(一)、CDP-Tree 之檢定運算如下：

$$ConD_{D \to D'}(1,3,1) = \frac{13/162 - 32/232}{\sqrt{0.1142 \quad (1 - 0.1142 \quad)\left(\frac{1}{162} + \frac{1}{232}\right)}} = 1.771279$$

$$ConD_{D \to D'}(1,3,2) = \frac{79/162 - 120/232}{\sqrt{0.5051 \quad (1 - 0.5051 \quad)\left(\frac{1}{162} + \frac{1}{232}\right)}} = 0.577973$$

$$ConD_{D \to D'}(1,3,3) = \frac{12/162 - 22/232}{\sqrt{0.0863 \quad (1 - 0.0863 \quad)\left(\frac{1}{162} + \frac{1}{232}\right)}} = 0.721856$$

$$ConD_{D \to D'}(1,3,4) = \frac{58/162 - 58/232}{\sqrt{0.2944 \quad (1 - 0.2944 \quad)\left(\frac{1}{162} + \frac{1}{232}\right)}} = 2.314841$$

二、卡方-齊一性檢定

以表 3.5 的兩區塊之屬性值 a13 為例，先取出欲檢定之屬性值欄位，與下一區塊相對應的屬性值欄位，如表 3.6。卡方-齊一性檢定在這個例子裡需要三次檢定運算，以每屬性值（每欄）一次運算。相較 CDP-Tree 檢定方法較具優勢，由於此時 CDP-Tree 檢定方式無法以屬性值內之任一個類別檢定結果來推估其他類別的檢定結果，故 CDP-Tree 檢定方法則需要 4 類別 x3 屬性值=12 次的檢定才能評估這兩區塊

之屬性值底下，資料有沒有顯著差異，因此我們的目的是在上述問題上設計一個可適用於多類別資料的概念漂移偵測法之可行方案。

表 3.6 觀察次數

	Bt-value1	Bt+1-Value1	總合
類別 1	13	32	45
類別 2	79	120	199
類別 3	12	22	34
類別 4	58	58	116
總合	162	232	394

(一)、計算期望值。

$$\hat{E}11 = \frac{162*45}{394} = 18.503 \qquad \hat{E}21 = \frac{232*45}{394} = 26.298$$

$$\hat{E}12 = \frac{162*199}{394} = 81.822 \qquad \hat{E}22 = \frac{232*199}{394} = 117.178$$

$$\hat{E}13 = \frac{162*34}{394} = 13.98 \qquad \hat{E}23 = \frac{232*34}{394} = 20.02$$

$$\hat{E}14 = \frac{162*116}{394} = 47.695 \qquad \hat{E}24 = \frac{232*116}{394} = 68.305$$

(二)、計算檢定量。

$$\frac{(13-18.503)^2}{18.503} = 1.636 \qquad \frac{(32-26.298)^2}{26.298} = 1.143$$

$$\frac{(79-81.822)^2}{81.822} = 0.097 \qquad \frac{(120-117.178)^2}{117.178} = 0.068$$

$$\frac{(12-13.98)^2}{13.98} = 0.28 \qquad \frac{(22-20.02)^2}{20.02} = 0.196$$

$$\frac{(58-47.695)^2}{47.695} = 2.226 \qquad \frac{(58-68.305)^2}{68.305} = 1.555$$

(三)、依臨界值與統計量計算出檢定結果。

總$\chi^2 = 1.636 + 0.097 + 0.28 + 2.226 + 1.143 + 0.068 + 0.196 + 1.555 = 7.2014$
臨界值 = 7.81473(查表(4 - 1)(2 - 1)0.05)
7.2014 < 7.81473(小於臨界值表示無顯著差異)

參、CDP-Tree 演算法之問題

無法檢定類別間計數的實際變動（其檢定結果對類別的變動沒有直接性意義），然而卻需要每類別的檢定結果，以判斷兩群體之間的檢定結果，使得計算上較為複雜。

第三節 概念漂移案例

概念漂移案例為根據概念漂移的情況，來針對分類器進行調整與修建的準則依據，而在 CDP-Tree 演算法中包含四種漂移案例，而本研究基於降低維護分類器成本的前提下，本研究再針對其案例四（多屬性漂移）的部份提出改進與調整。在本節中將說明在漂移案例中的情形，以及漂移的案例處理。圖 3.4 中的樹狀圖為案例之區分情形，分別以屬性是否相同（是否集中）與屬性值是否相同（是否集中）作為區分；以及在多屬性漂移中是否存在屬性具一個以上之屬性值漂移以作為區分。案例五為原 CDP-Tree 案例四之細分。

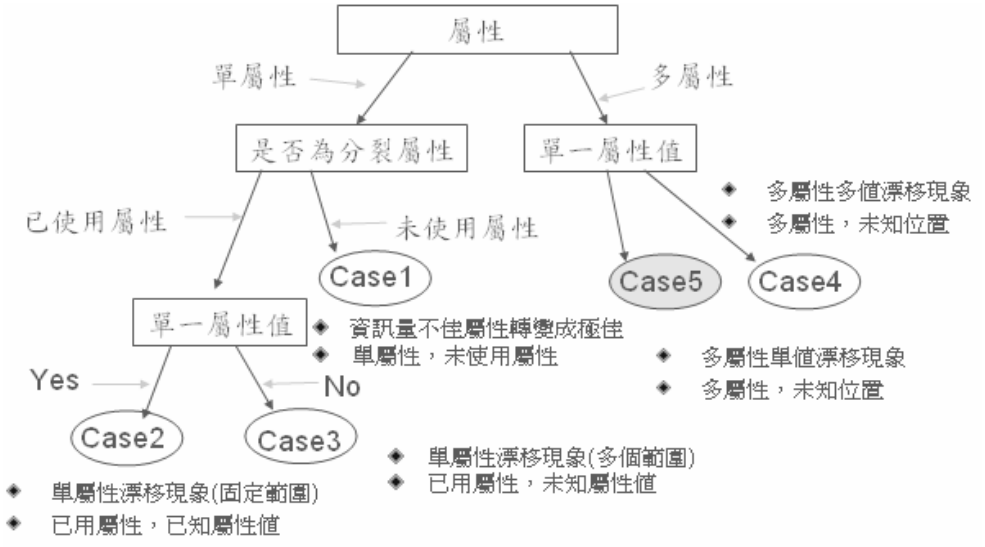


圖 3.4 概念漂移案例

在圖 3.4 中我們從根節點開始看到在屬性下區分為單屬性與多屬性（一個以上屬性）。而多屬性漂移依照在同屬性下漂移個數來區分為（各別漂移屬性下包含一個以上之漂移屬性值）Case4 與（各別漂移屬性下僅有一個漂移屬性值）Case5。而在單屬性（某個別屬性之下）下根據是否為分裂屬性區分為已使用（已存在於現今決策樹）與未使用（未存在於現今決策樹）之屬性 Case1。單一已使用之屬性的（已存在於現今決策樹之屬性下）屬性值是否相同（是否集中）區分為單屬性值（同屬性值）Case2，與多屬性值（單一屬性下之不同屬性值內）Case3 之漂移。由於 CDP-Tree 之 Case4 調整方法為完全重建，故其耗用成本相當可觀，因此本論文增加 Case5 案例，提出以上之概念漂移案例圖改良架構。

壹、案例 1：(未使用之屬性)

一、Case1 概念：

Step1：判斷是所有漂移的概念元皆對應至同一屬性：如表 3.7 與表 3.8 所示陰影部份都集中在屬性 a3 的屬性值 v1&v2 均對應到屬性 a3。

Step2：判斷不為原決策樹上之分裂屬性：如圖 3.5 所示 a3 屬性不屬於原決策樹上之任何分裂節點。

Step3：調整針對決策樹上所有葉節點作再次分裂：如圖 3.6(a)所示，我們使用 a3 針對原決策樹作再次分裂測試，得出圖 3.6(b)。

表 3.7 Bt

Bt	a1	a2	a3		
			v1	v2	v3
c1			1	1	2
c2			2	4	3

表 3.8 Bt+1

Bt+1	a1	a2	a3		
			v1	v2	v3
c1			3	4	2
C2			2	1	3

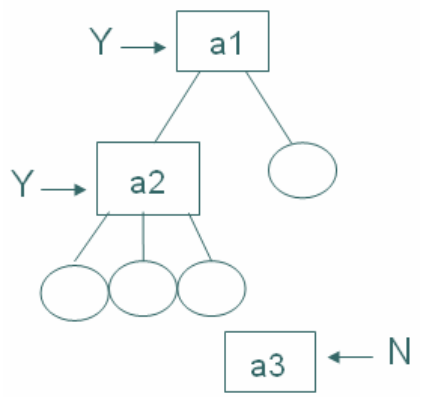


圖 3.5 原決策樹

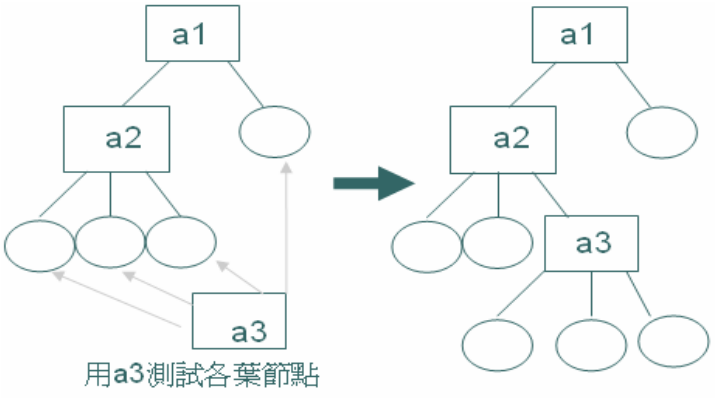


圖 3.6 Case1 修建樹

二、Case1 實例：

如圖 3.7 所示，在第二個屬性 income 的屬性值 High 部份發生了漂移的現象（兩者作卡方檢定有顯著差異），圖 3.8 為資料區塊 Bt 時所建

立之決策樹，而圖 3.9 則為區塊 Bt 的決策樹 DTt 經過案例 1 的調整後的決策樹 DTt+1,我們可以看到標明虛線部份為與前一個決策樹 DTt 不同之部份，而所代表之規則如下。(正確率如右：修建前：50；修建後：85.71；重建後：78.57；平均正確率為實際漂移案例正確率相加除以總區塊數)

		age			income			student		c_ratin	
		<=30	30..40	>40	High	Medium	Low	Yes	No	Fair	Excellent
Bt (初始爲ID3)	YES	5	4	2	6	2	3	3	8	5	6
	NO	2	0	1	2	0	1	2	1	1	2
Bt+1	YES	1	1	4	2	2	2	3	3	4	2
	NO	4	2	2	1	2	5	7	1	2	6
卡方顯差		-0.75574	-0.10812	-3.84146	-3.76507	-2.34146	-1.62836	-2.59146	-3.43110	-3.39701	0.15854

圖 3.7　Case1 實例計次表

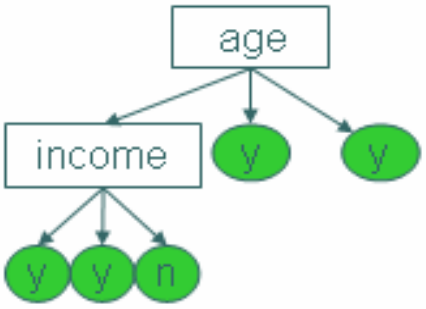


圖 3.8 原決策樹

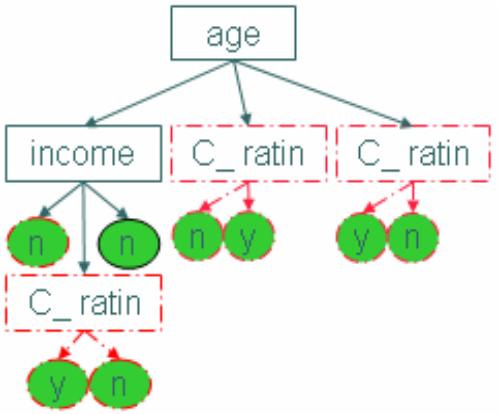


圖 3.9 Case1 修建樹

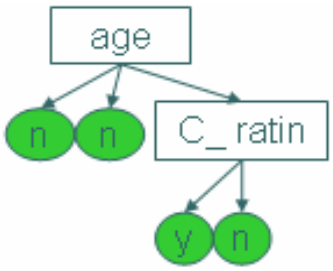


圖 3.10 重建樹

(一)、資料區塊 Bt(圖 3.8)所求得的規則為：

$age = "<= 30" \cap income = "high" \rightarrow class = yes$

$age = "<= 30" \cap income = "medium" \rightarrow class = yes$

$age = "<= 30" \cap income = "low" \rightarrow class = no$

$age = "30..40" \rightarrow class = yes$

$age = "> 40" \rightarrow class = yes$

(二)、資料區塊 Bt 根據 Bt+1 而修建所得求得的規則(圖 3.9)為：

$age = "<= 30" \cap income = "high" \rightarrow class = no$

$age = "<= 30" \cap income = "medium" \cap c\_ratin = "fair" \rightarrow class = yes$

$age = "<= 30" \cap income = "medium" \cap c\_ratin = "excellent" \rightarrow class = no$

$age = "<= 30" \cap income = "low \rightarrow class = no$

$age = "30..40" \cap c\_ratin = "fair" \rightarrow class = no$

$age = "30..40" \cap c\_ratin = "excellent" \rightarrow class = yes$

$age = "> 40" \cap c\_ratin = "fair" \rightarrow class = yes$

$age = "> 40" \cap c\_ratin = "excellent" \rightarrow class = no$

(三)、資料區塊根據 Bt+1 重建決策樹所求得的規則(圖 3.10)為：

$age = "<= 30" \rightarrow class = no$

$age = "30..40" \rightarrow class = no$

$age = "> 40" \cap c\_ratin = "fair" \rightarrow class = yes$

$age = "> 40" \cap c\_ratin = "excellent" \rightarrow class = no$

貳、案例 2：單屬性漂移（固定範圍）

一、Case2 概念：

Step1：判斷所有漂移的概念元皆對應至同一屬性：如表 3.9 和表 3.10 所示，漂移的屬性皆對應到 a2 屬性的 v1 屬性值。

Step2：判斷為原決策樹上之分裂屬性：如圖 3.8 所示，該漂移屬性存在於 Bt 決策樹上。

Step3：判斷所有漂移之概念元皆對應到同一屬性值：如表 3.10 所示所有漂移的概念元皆對應到 a1=v1∩a2=v1 的屬性裡。

Step4：調整：針對原圖 3.11 上所有該漂移屬性之屬性值節點再次分裂；如圖 3.12 在 a2 屬性的屬性值 v1 發生漂移（即檢定有顯著差異），則我們使用區塊 Bt+1 (表 3.10)下之 a1=v1∩a2=v1 的資料重新作為圖 3.11 該節點的資料來重建此子節點。

表 3.9 Bt

Bt

	a1			a2		
				v1	v2	v3
c1				1	1	2
c2				2	4	3

表 3.10 Bt+1

Bt+1

	a1			a2		
				v1	v2	v3
c1				3	1	2
C2				2	4	3

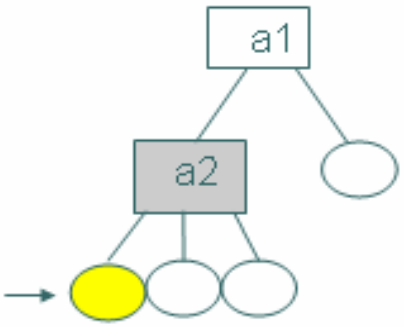


圖 3.11 決策樹

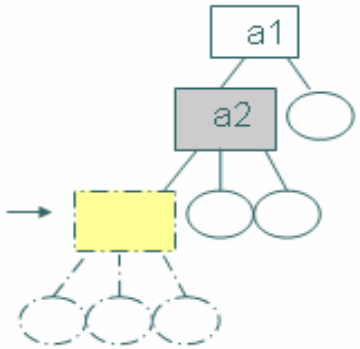


圖 3.12 Case2 修建樹

二、Case2 實例：

如圖 3.13 所示，在第一個屬性 age 的屬性值 30..40 部份有發生

了漂移的現象（兩者作卡方檢定有顯著差異），圖 3.14 為資料區塊 Bt 時所建立之決策樹，而圖 3.15 則為區塊 Bt 的決策樹 DTt 經過案例 2 的調整後的決策樹 DTt+1,我們可以看到標明虛線部份為與前一個決策樹 DTt 不同之部份，而所代表之規則如下。(正確率如右：修建前：42.85；修建後：64.28；重建後：85.71)

		age			income			student		c_ratin	
		<=30	30..40	>40	High	Medium	Low	Yes	No	Fair	Excellent
Bt (初始為ID3)	YES	3	1	3	1	2	4	0	7	4	3
	NO	2	3	2	3	2	2	2	5	6	1
Bt+1	YES	1	4	3	4	2	2	2	6	4	4
	NO	1	0	5	2	2	2	3	3	2	4
卡方顯差		-3.78312	0.95854	-3.21467	-2.17479	-3.84146	-3.56368	-2.72146	-3.69001	-2.77479	-3.15574

圖 3.13 Case2 實例計次表

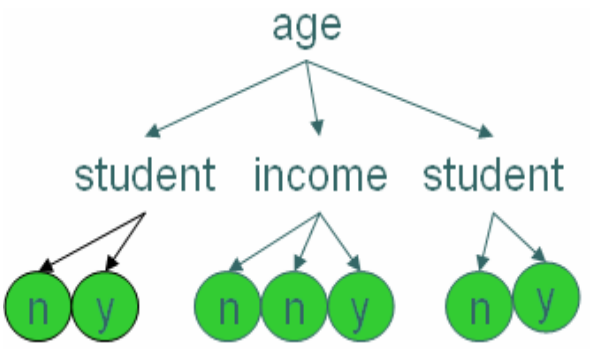


圖 3.14 原決策樹

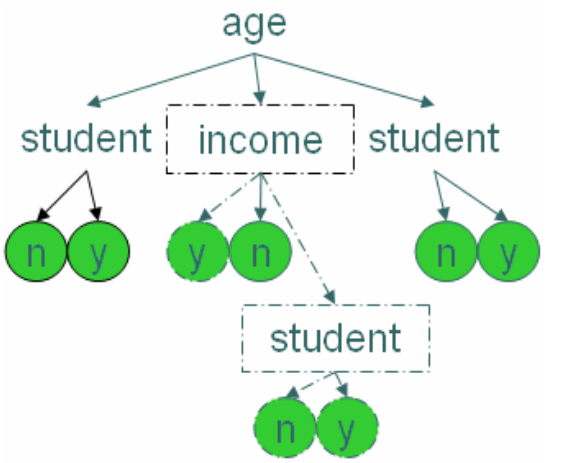


圖 3.15 Case2 修建樹

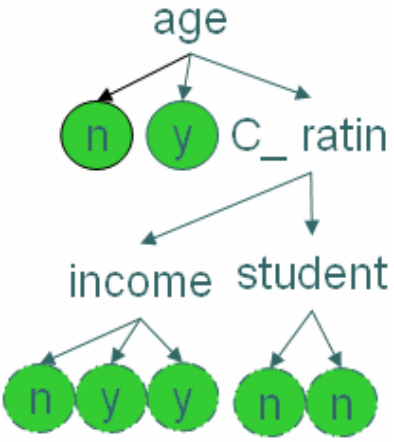


圖 3.16 重建樹

(一)、資料區塊 Bt(圖 3.14)所求得的規則為：

$age = "< 30" \cap student = "yes" \rightarrow class = no$

$age = "< 30" \cap student = "no" \rightarrow class = yes$

$age = "30..40" \cap income = "high" \rightarrow class = no$

$age = "30..40" \cap income = "medium" \rightarrow class = no$

$age = "30..40" \cap income = "low" \rightarrow class = yes$

$age = "> 40" \cap student = "yes" \rightarrow class = no$

$age = "> 40" \cap student = "no" \rightarrow class = yes$

(二)、資料區塊 Bt 根據 Bt+1 而修建所得求得的規則(圖 3.15)為：

$age = "< 30" \cap student = "yes" \rightarrow class = no$

$age = "< 30" \cap student = "no" \rightarrow class = yes$

$age = "30..40" \cap income = "high" \rightarrow class = yes$

$age = "30..40" \cap income = "medium" \rightarrow class = no$

$age = "30..40" \cap income = "low" \cap istudent = "yes" \rightarrow class = no$

$age = "30..40" \cap income = "low" \cap istudent = "no" \rightarrow class = yes$

$age = "> 40" \cap student = "yes" \rightarrow class = no$

$age = "> 40" \cap student = "no" \rightarrow class = yes$

(三)、資料區塊根據 Bt+1 重建決策樹所求得的規則(圖 3.16)為：

$age = "<= 30" \rightarrow class = no$

$$age = "30..40" \rightarrow class = yes$$

$$age = "> 40" \cap c\_ratin = "fair" \cap income = "high" \rightarrow class = no$$

$$age = "> 40" \cap c\_ratin = "fair" \cap income = "medium" \rightarrow class = yes$$

$$age = "> 40" \cap c\_ratin = "fair" \cap income = "low" \rightarrow class = yes$$

$$age = "> 40" \cap c\_ratin = "excellent" \cap student = "yes" \rightarrow class = no$$

$$age = "> 40" \cap c\_ratin = "excellent" \cap student = "no" \rightarrow class = no$$

參、案例 3：單屬性漂移（多範圍）

一、Case3 概念：

Step1：判斷是所有漂移的概念元皆對應至同一屬性：如表 3.11 表 3.12 所示，其陰影（漂移）的部份均屬於同一屬性之下。

Step2：判斷為原決策樹上之分裂屬性：如圖 3.17 所示，漂移的屬性為 a2，屬於原決策樹上之分裂節點。

Step3：判斷漂移的概念元之屬性值位置不完全相同：如表 3.11、表 3.12、圖 3.17 所示，漂移的屬性值不完全相同(並非皆在同一屬性值中)。

Step4：針對決策樹上所有該漂移屬性之節點再次分裂：如圖 3.18 在該漂移屬性再次分裂，以 bt+1（表 3.12）之資料重新在圖 3.17 的屬性 a2 部份進行分裂測試。

表 3.11 Bt

Bt

	a1		a2		
			v1	v2	v3
c1			1	1	2
c2			2	4	3

表 3.12 Bt+1

Bt+1

	a1		a2		
			v1	v2	v3
c1			3	4	2
C2			2	1	3

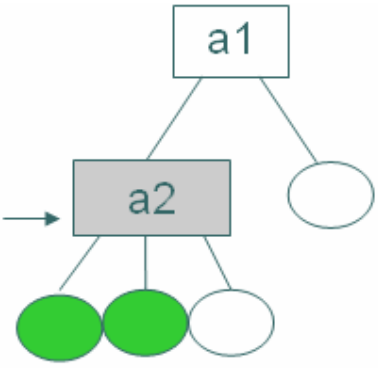


圖 3.17 原決策樹

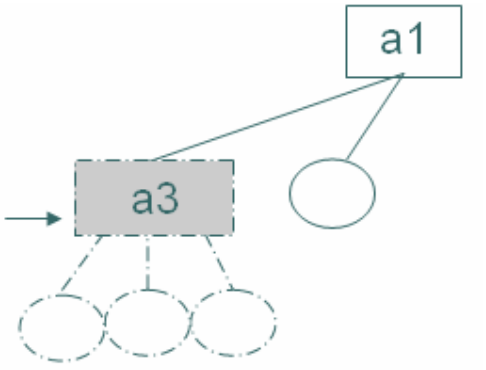


圖 3.18 Case3 修建樹

二、Case3 實例：

如圖 3.16 所示，在第四屬性 c_ratin 的屬性值 Fair 與 Excellent 部份有發生了漂移的現象（Bt 與 Bt+1 作卡方檢定有顯著差異），圖 3.17 為資料區塊 Bt 時所建立之決策樹，而圖 3.18 則為區塊 Bt 的決策樹 DTt 經過案例 3 的調整後的決策樹 DTt+1,我們可以看到標明虛線部份為與前一個決策樹 DTt 不同之部份，而所代表之規則如下。(正確率如右：修建前：35.71；修建後：57.14；重建後：71.42)

次數狀態顯示

		age			income			student		c_ratin	
		<=30	30..40	>40	High	Medium	Low	Yes	No	Fair	Excellent
Bt (初始爲ID3)	YES	0	5	1	3	0	3	3	3	5	1
	NO	2	4	2	5	1	2	3	5	0	8
Bt+1	YES	2	4	1	2	1	4	3	4	3	4
	NO	2	3	2	2	0	5	2	5	4	3
卡方顯差		-2.34146	-3.83742	-3.84146	-3.67003	-1.84146	-3.53034	-3.73146	-3.75713	0.444254	0.041945

圖 3.19 Case3 實例計次表

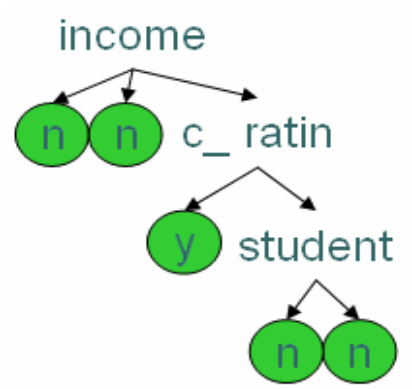


圖 3.20 原決策樹

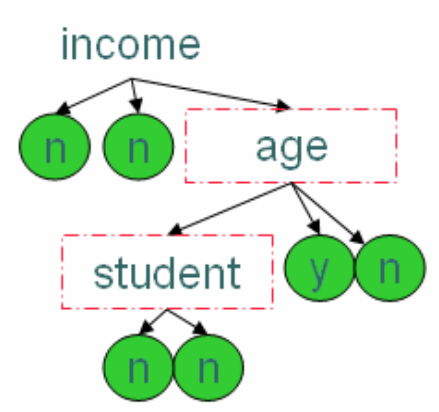


圖 3.21 Case3 修建樹

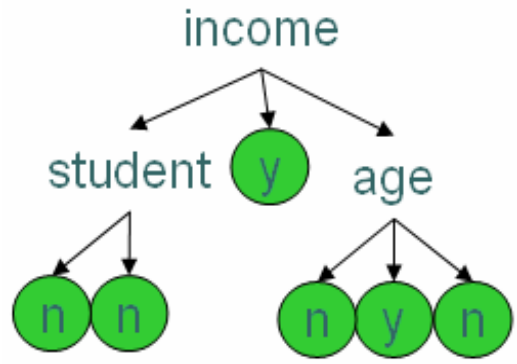


圖 3.22 重建樹

(一)、資料區塊 Bt(圖 3.20)所求得的規則為：

$income = "high" \rightarrow class = no$

$income = "medium" \rightarrow class = no$

$income = "low" \cap c\_ratin = "fair" \rightarrow class = yes$

$income = "low" \cap c\_ratin = "excellent" \cap student = "yes" \rightarrow class = no$

$income = "low" \cap c\_ratin = "excellent" \cap student = "no" \rightarrow class = no$

(二)、資料區塊 Bt 根據 Bt+1 而修建所得求得的規則(圖 3.21)為：

$income = "high" \rightarrow class = no$

$income = "medium" \rightarrow class = no$

$income = "low" \cap age = "<= 30" \cap student = "yes" \rightarrow class = no$

$income = "low" \cap age = "<= 30" \cap student = "no" \rightarrow class = no$

$income = "low" \cap age = "30..40" \rightarrow class = yes$

$income = "low" \cap age = "> 40" \rightarrow class = no$

(三)、資料區塊根據 Bt+1 重建決策樹所求得的規則(圖 3.22)為：

$income = "high" \cap student = "yes" \rightarrow class = no$

$income = "high" \cap student = "no" \rightarrow class = no$

$income = "mudium" \rightarrow class = yes$

$income = "low" \cap age = "< 30" \rightarrow class = no$

$income = "low" \cap age = "30..40" \rightarrow class = yes$

$income = "low" \cap age = "> 40" \rightarrow class = no$

肆、案例 4：多屬性（多範圍）

一、Case4 概念：

Step1：判斷所有漂移的概念元對應至一個以上之屬性，且至少有一屬性含一個以上之屬性值對應：

Step2：針對新資料重建決策樹：

表 3.13 Bt

Bt

	a1			a2		
	v1	v2	v3			
c1	1	4	2			
c2	4	1	3			

表 3.14 Bt+1

Bt+1

	a1			a2		
	v1	v2	v3			
c1	4	4	2			
C2	2	1	3			

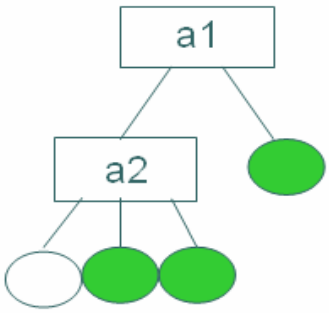


圖 3.23 原決策樹

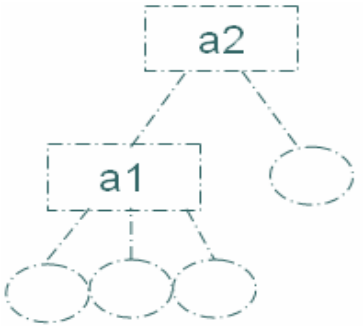


圖 3.24 Case4 修建樹（重建樹）

二、Case4 實例：

如圖 3.25 所示，在第多個屬性都出現漂移現象；而 age 屬性的屬性值“30..40”與屬性值“>40”也都出現了漂移。圖 3.26 為資料區塊 Bt 時所建立之決策樹 DTt，而圖 3.27 則為區塊 Bt 的決策樹 DTt 經過案例 4 的重建後的決策樹 DTt+1,我們可以看到標明虛線部份為與前一個決策樹 DTt 不同之部份，而所代表之規則如下。(正確率如右：修建前：35.71；修建後：92.85；重建後：92.85)

		age			income			student		c_ratin	
		<=30	30..40	>40	High	Medium	Low	Yes	No	Fair	Excellent
Bt (初始爲ID3)	YES	2	4	3	2	4	3	6	3	6	3
	NO	3	0	2	2	2	1	1	4	2	3
Bt+1	YES	2	0	0	0	0	2	2	0	0	2
	NO	1	4	7	3	2	7	8	4	7	5
卡方關差		-2.17220	5.294459	2.894459	-0.60554	-0.03887	0.553486	4.431761	-0.34839	6.044459	-2.07875

圖 3.25 Case4 實例計次表

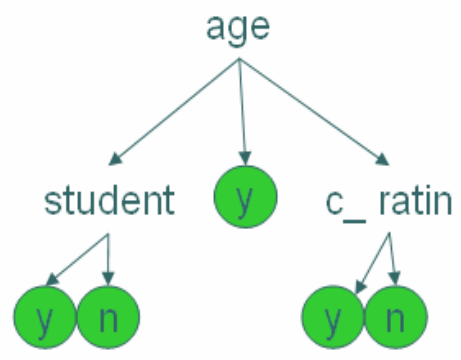


圖 3.26 原決策樹

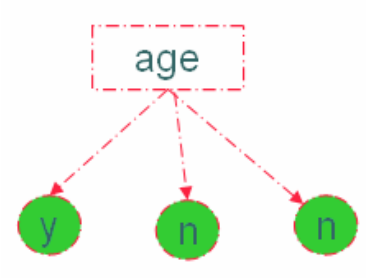


圖 3.27 Case4 修建樹(重建樹)

(一)、圖 3.26資料區塊 Bt 所求得的規則如下所示：

$age = "< 30" \cap student = "yes" \rightarrow class = yes$

$age = "< 30" \cap student = "no" \rightarrow class = no$

$age = "30..40" \rightarrow class = yes$

$age = "> 40" \cap c\_ratin = "fair" \rightarrow class = yes$

$age = "> 40" \cap c\_ratin = "excellent" \rightarrow class = no$

(二)、圖 3.27 則是使用資料區塊 Bt+1 所重建求得的規則如下所示：

$age = "< 30" \rightarrow class = yes$

$age = "30..40" \rightarrow class = no$

$age = "> 40" \rightarrow class = no$

伍、案例 5：多屬性單屬性值（固定範圍）

一、Case5 概念：

Step1：判斷所有漂移的概念元對應至一個以上之屬性，且對應到單一屬性內之漂移概念元<=1：

Step2：針對 bt+1 之資料對原決策樹上之漂移屬性節點作再分裂，若有漂移屬性再另一漂移屬性之子樹時，以含括對方者作為調整基準。

表 3.15 Bt

Bt

	a1			a2		
	v1	v2	v3			
c1	1	4	2			
c2	4	1	3			

表 3.16 Bt+1

Bt+1

	a1			a2		
	v1	v2	v3			
c1	4	4	2			
C2	2	1	3			

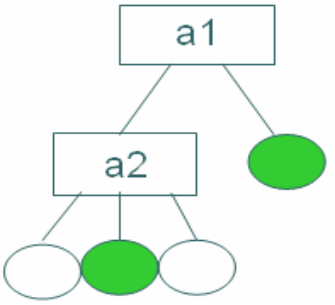


圖 3.28 原決策樹

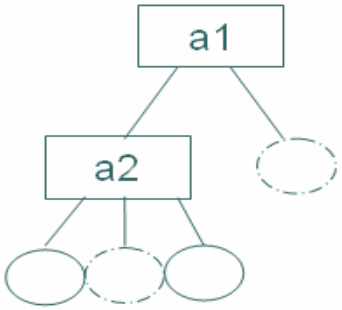


圖 3.29 Case5 修建樹

二、Case5 實例：

如圖 3.30 所示，在第一個屬性 age 的屬性值“30..40”與在第三個屬性 student 的屬性值“yes”部份有發生了漂移的現象（Bt 與 Bt+1 作卡方檢定有顯著差異），圖 3.31 為資料區塊 Bt 時所建立之決策樹，而圖 3.32 則為區塊 Bt 的決策樹 DTt 經過案例 4 的重建後的決策樹 DTt+1,我們可以看到標明虛線部份為與前一個決策樹 DTt 不同之部份，而所代表之規則如下。(正確率如右：修建前：50；修建後：85.71；重建後：92.85)

		age			income			student		c_ratin	
		<=30	30..40	>40	High	Medium	Low	Yes	No	Fair	Excellent
Bt (初始爲ID3)	YES	2	4	3	2	4	3	6	3	6	3
	NO	3	0	2	2	2	1	1	4	2	3
Bt+1	YES	1	2	0	0	1	2	1	2	2	1
	NO	5	1	5	4	5	2	6	5	8	3
	卡方顯差	-1.95693	-1.14998	1.580173	-0.03887	0.380173	-2.17220	4.437316	-2.39443	2.73946	-2.08054

圖 3.30 Case5 實例計次表

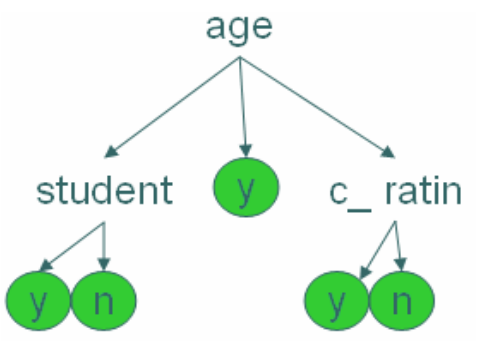


圖 3.31 原決策樹

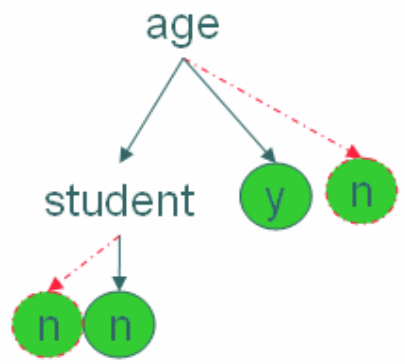


圖 3.32 Case5 修建樹

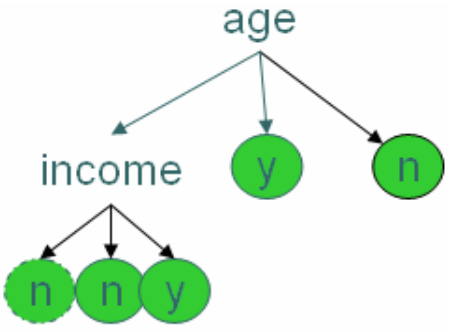


圖 3. 33 重建樹

(一)、圖 3.31 資料區塊 Bt 所求得的規則如下所示：

$age = "< 30" \cap student = "yes" \rightarrow class = yes$

$age = "< 30" \cap student = "no" \rightarrow class = no$

$age = "30..40" \rightarrow class = yes$

$age = "> 40" \cap c\_ratin = "fair" \rightarrow class = yes$

$age = "> 40" \cap c\_ratin = "excellent" \rightarrow class = no$

(二)、圖 3.32 則是使用資料區塊 Bt+1 所修建求得的規則如下所示：

$age = "< 30" \cap student = "yes" \rightarrow class = no$

$age = "< 30" \cap student = "no" \rightarrow class = no$

$age = "30..40" \rightarrow class = yes$

$age = "> 40" \rightarrow class = no$

(三)、圖 3.33 則是使用資料區塊 Bt+1 所重建求得的規則如下所示：

$age = "< 30" \cap income = "high" \rightarrow class = no$

$age = "< 30" \cap income = "medium" \rightarrow class = no$

$age = "< 30" \cap income = "low" \rightarrow class = yes$

$age = "30..40" \rightarrow class = yes$

$age = "> 40" \rightarrow class = no$

第四節　CDC 演算法/程式流程

本節將介紹 CDC 演算法，及程式流程相關區塊，並輔以第五節之完整例子來表達本論文演算法之精神與作法，以使未來若需相關實作時能有所依據。

壹、CDC 演算法：

演算法如圖 3.34 將以行號方式表達與程式流程相關之區塊，並在程式流程部份輔以實例來說明演算法的進行。

一、演算法說明：

（一）、第 1、2、3 行：輸入所需之參數，依次為檢定顯著水準、區塊數、資料筆數/每區塊（對照程式流程 1.）。

（二）、第 4 行：為建立一個初始資料區塊，其區塊大小視前述設定參數而定（對照程式流程 2.）。

(三)、第 5 行：產生一個 Bt 次數表，依屬性、屬性值、目標類別來歸類次數以作為檢定之用（對照程式流程 3.)。

(四)、第 6 行:使用 Bt 區塊建立一個初始決策樹(對照程式流程 4.)。

(五)、第 9 行：產生 Bt+1(Next Bt+1)資料（假定的下一個進入之資料區塊)。(對照程式流程 5.)

(六)、第 10 行：為產生 Bt+1 之次數表作為檢定時之用（對照程式流程 6.)。

(七)、第 11 行：計算並判斷 Bt 與 Bt+1 之卡方顯著差異（是否漂移）（對照程式流程 8、9.)。

(八)、第 12 行：若 11 行不成立則未漂移，則執行 12 行刪除掉 Bt+1 之次數表（對照程式流程 15.)。

(九)、第 14～18 行：若 11 行比較結果有顯著差異，則執行 15 行，呼叫執行 21 行～43 行針對漂移案例作區分與調整（對照程式流程 10.)。

(十)、第 16 行：於 21~43 行結束後跳回 16 行將此次漂移之屬性的 Count 取代原 Bt 次數表之屬性 count。(對照程式流程 14.)。

(十一)、第 8、19 行：為進行下一個區塊之檢定與處理。(對照程式流程 19.)

(十二)、第 25、27、30、37、40 行：為演算法之漂移案例分類處理。（對照程式流程 12.)

```
01 α                                    //設定檢定顯著水準；
02 BlockNum=250，；     //設定區塊數(BlockNum)。
03 DataCount=500；     //及每區塊內含之資料筆數(DataCount)。
04 Set Bt is FirstBlock；   //設 Bt 為第一個進入的區塊。
05 Make List of Count With Bt；   //產生 Bt 次數表。
06 Dim DTt as DecisionTree With Bt;   //令 DTt 為一個 Bt 所建立的決策樹。
07 //---------------------------------------------------------------------------
08 For (Block=1 to BlockNum){          //設終止條件為執行 BlockNum 次。
09   Set Bt+1 is NextBlock；     //設 Bt+1=下一個進入的區塊。
10   Make List of Count With Bt+1；          //產生 Bt+1 次數表。
11   If (χ²(Bt,Bt+1)<χ²(r−1)(c−1),α){//比較 Bt 與 Bt+1 之卡方檢定結果。
12        Delete CountList of Bt+1；//若小於則丟棄 Bt+1 的次數表。
13   }
14   Else{
15        CDCasePorcess()
16        To Replace CountList With Bt By Bt+1；
17        //將 Bt+1 漂移屬性之次數取代 Bt 之次數。
18   }
19 } Next Block   //進行下一進入區塊。
20 //---------------------------------------------------------------------------
21 CDCasePorcess(){   //漂移案例處理。
22   If (Attribute is Same) {     //如果屬性相同，表示為單屬性內之漂移現象。
23        If (Attribute =best Attribute) {   //屬性為最佳屬性，表示已在決策樹上。
24             If Attribute is Same //在樹上且屬性值相同，單屬性單屬性值漂移。
25             Case 2   //單屬性單屬性值漂移之調整案例 2。
26             Else                    //屬性值不同，單屬性多屬性值漂移。
27             Case 3   //單屬性多屬性值漂移之調整案例 3。
28        }
29        Else {   //不在樹上之屬性。
30             Case1   //未在樹上之單屬性漂移調整案例 1。
31        }
32   }
33   Else{   //一個以上之屬性漂移，多屬性漂移。
34
35        If (each value of Attribute is Same With Single Attribute){
36        //單一屬性下最多只有一個屬性值之概念元漂移
37             Case5   //多屬性單屬性值漂移調整案例 5。
38        }
39        Else{
40             Case4   //多屬性多屬性值漂移調整案例 4。
41        }
42   }
43}
```

圖 3.34 CDC 演算法

貳、CDC 演算法-程式流程圖：

程式流程圖 3.35 將說明在實作程式時所使用的驗証方式，與實作方法，並將輔以實例與演算法交互參照，以使整個演算法之方法、實作能夠配合，藉以說明演算概念。

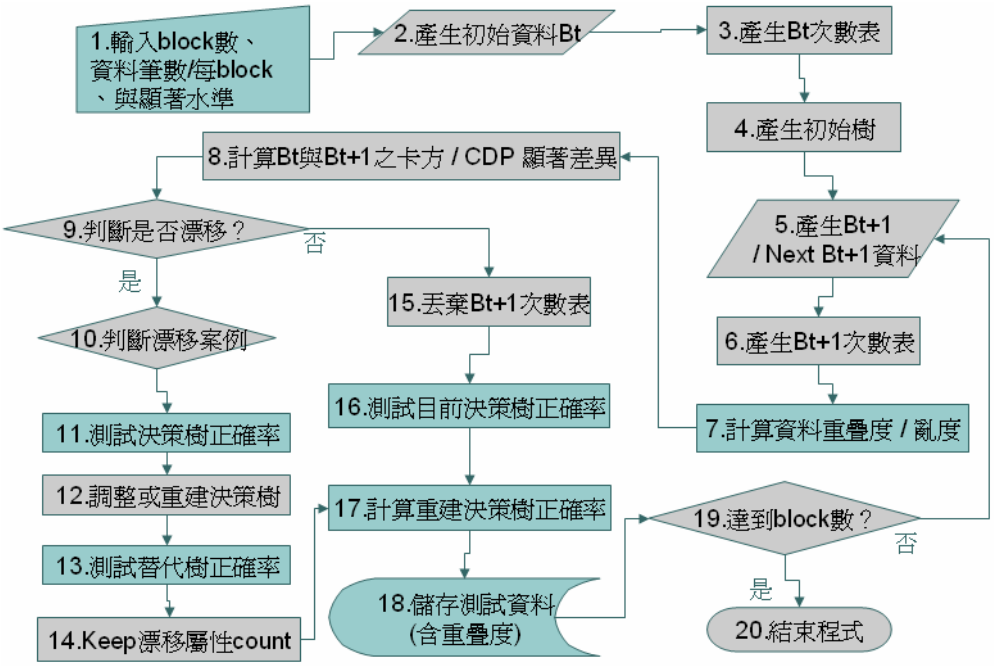


圖 3.35 CDC 演算法程式流程圖

一、程式流程圖說明：

以下將之程式流程圖之序號來說明其實例與作法，其中談到「無關演算法」部份為程式進行時為測試結果所作之動作，與演算法並無絕對關係。

程序1、輸入區塊數（block）、資料筆數/每區塊、顯著水準；例如輸

入之區塊數為 250 個，而每區塊共 14 筆資料，顯著水準為 0.5。

程序2、產生初始資料 Bt：設如圖 3.17 為初始資料表 Bt。

程序3、產生 Bt 次數表：如圖 3.18 為依據 3.17 所產生之次數表。

程序4、以 Bt 產生初始樹：依第五節，建立決策樹之步驟建立如圖 3.38 之初始樹。

程序5、產生 Bt+1(Next Bt+1)資料：設表 3.21 為新進資料集合 Bt+1。

程序6、產生 Bt+1 次數表：以表 3.21 資料建立如表 3.22 之次數表。

程序7、計算資料重疊度/亂度：無關演算法。

程序8、計算 Bt 與 Bt+1 之卡方顯著差異：藉以獲知兩區塊是否漂移；如表 3.23，及該節步驟可用以計算其卡方檢定結果。

程序9、判斷是否漂移：如表 3.23 所示，有三個屬性下有漂移（卡方檢定有顯著差異）。

程序10、判斷漂移案例：依照漂移案例區分，各別屬性之漂移概念元均對應到一個以下之屬性值，則此漂移案例為案例五。

程序11、測試決策樹調整前正確率：無關演算法；以 Bt+1 資料對調整前決策樹進行測試，當依屬性、屬性值區分之其分類無誤時為正確；以全部資料正確筆數除以全部資料筆數取得正確率。

程序12、調整或重建決策樹：依其漂移調整案例使用 Bt+1 資料對原決策樹進行調整，或重建的動作。

程序13、測試替代樹正確率：無關演算法；以 Bt+1 資料對替代樹進行測試，當依屬性、屬性值區分之其分類無誤時為正確；以全部資料正確筆數除以全部資料筆數取得正確率。

程序14、Keep 漂移屬性 Count：若有漂移則保留 Bt+1 相關漂移位置之計次資料。

程序15、丟棄 Bt+1 次數表：若 Bt 與 Bt+1 之資料區塊並無顯著差異，依經驗法則，可繼續使用 Bt 資料，故可丟棄 Bt+1 之次數表。

程序16、測試目前決策樹正確率：無關演算法；同前述。

程序17、計算重建決策樹正確率：無關演算法；同前述。

程序18、儲存測試資料：無關演算法。

程序19、達到 block 數：若所設定之區塊數目已達成，則跳出程式。

程序20、結束程式：無關演算法。

第五節　CDC 演算法實例

本演算法以卡方檢定來綜合比較類別之間的比例變化，並且增加一個漂移案例以修正在多屬性漂移中常見的單屬性值漂移產生之大量重建所造成的巨大計算成本，並一個完整連貫的例子來說明卡方漂移偵測演算法 CDC(Concept Drift Detection of Chi-Square)之檢定範例，以及案例五的調整方式；由於案例 1~4 在之前的章節已有介紹，相關可參考 CDP-Tree[1]此節不再贅述。

壹、決策樹建立準則：

本演算法決策樹依照第二章介紹之 Information Gain 公式進行分裂屬性之選擇依據，且輔以下列三種考量四項準則進行決策樹之建立。

一、純化：

先測試是否純化，若剩下之資料集合之所屬類別完全相同，則直接將此節點分支設成該相同所屬類別。

二、強制純化（為預防過度學習）：

(一)、測試資料筆數是否少於等於 3 筆，若成立則以目前之資料所屬類別眾數決定該節點類別。

(二)、為預防過度學習，測試資料筆數是否少於 3 筆且無眾數，若成立則亂數取出一個類別所以該節點的所屬類別。

三、分裂（未完成分裂，繼續）：

以上皆不成立，則再次使用各剩餘屬性進行分裂；(剩餘屬性係指，由該測試節點至根節點的路徑中尚未使用之屬性，可能依問題特性而決定是否可重覆使用同屬性進行分裂)。

貳、卡方漂移偵測演算法範例

以下依照 ID3 決策樹之電腦購買資料表作為範例，實作整個卡方漂移偵測演算法。包含初始決策樹建立、決策樹建立、概念漂移偵測、漂移案例調整方式之流程。

一、建立初始決策樹

設一個表 3.17 ID3 電腦購買資料表，該表為某段時間 t0~t1 所進入的資料（共 14 筆），如表 3.18 所示，共有四個屬性，Age、Income、Student、Credit_rating、每一屬性分別有兩至三個屬性值，而目標類別有兩個類別。

表 3.17 ID3 電腦購買資料表（Bt）

age	income	student	credit_rating	buy_computer
<=30	High	No	Fair	No
<=30	High	No	Excellent	No
30..40	High	No	Fair	Yes
>40	Medium	No	Fair	Yes
>40	Low	Yes	Fair	Yes
>40	Low	Yes	Excellent	No
30..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
30..40	Medium	No	Excellent	Yes
30..40	High	Yes	Fair	Yes
>40	Medium	No	Excellent	No

表 3.18 ID3 電腦購買資料計次表

屬性	age			income			student		c_rain	
屬性值	<=30	30..40	>40	High	Medium	Low	Yes	No	Fair	Excellent
Yes	2	4	3	2	4	3	6	3	6	3
No	3	0	2	2	2	1	1	4	2	3

(一)、我們將表 3.17 按屬性區分，依各屬性值之不同目標類別的分類計次如表 3.18。

(二)、並依據表 3.18，計算各屬性之 Information Gain 以選擇最佳分裂屬性。

(三)、依據決策樹建立準則依次測試 1. 純化、2. 強制純化、3. 分裂的動作，如需再分裂則依據公式 2.1 以計算並比較四屬性之 Information Gain 以挑選出此節點的最佳分裂屬性（決策樹之起始

節點為根節點)。計算過程如下所示：

1.依據 2.2 的公式我們可以表 3.18 之計次表來計算根節點測試前的資訊量如下：

$$I(5,9) = (-\frac{5}{14}\log\frac{5}{14}) + (-\frac{9}{14}\log\frac{9}{14}) = 0.283$$

2.依據公式 2.2 及 2.3 我們可以表 3.18 計次表來計算出各屬性的 E(a)(測試後的資訊量)如下：

$$E(age) = \frac{2+3}{14}I(2,3) + \frac{4+0}{14}I(4,0) + \frac{3+2}{14}I(3,2) = 0.2330472089$$
$$E(income) = \frac{2+2}{14}I(2,2) + \frac{4+2}{14}I(4,2) + \frac{3+1}{14}I(3,1) = 0.2516988352$$
$$E(student) = \frac{6+1}{14}I(6,1) + \frac{3+4}{14}I(3,4) = 0.2691886167$$
$$E(c\_rain) = \frac{6+2}{14}I(6,2) + \frac{3+3}{14}I(3,3) = 0.2819831554$$

3.依據 Information Gain 公式，測試前資訊量減掉測試後資訊量以計算各屬性作為分裂節點之優劣：

$$UseAgeInfoGain = I(5,9) - E(age) = 0.049952791$$
$$UseIncomeInfoGain = I(5,9) - E(income) = 0.031301164$$
$$UseStudentInfoGain = I(5,9) - E(student) = 0.013811383$$
$$UseC\_rainInfoGain = I(5,9) - E(c\_rain) = 0.0010168446$$

以上各屬性之資訊量以 age 屬性之 Information Gain 值為最高，故在此節點以 age 作為分裂屬性；建立如下圖之決策樹。

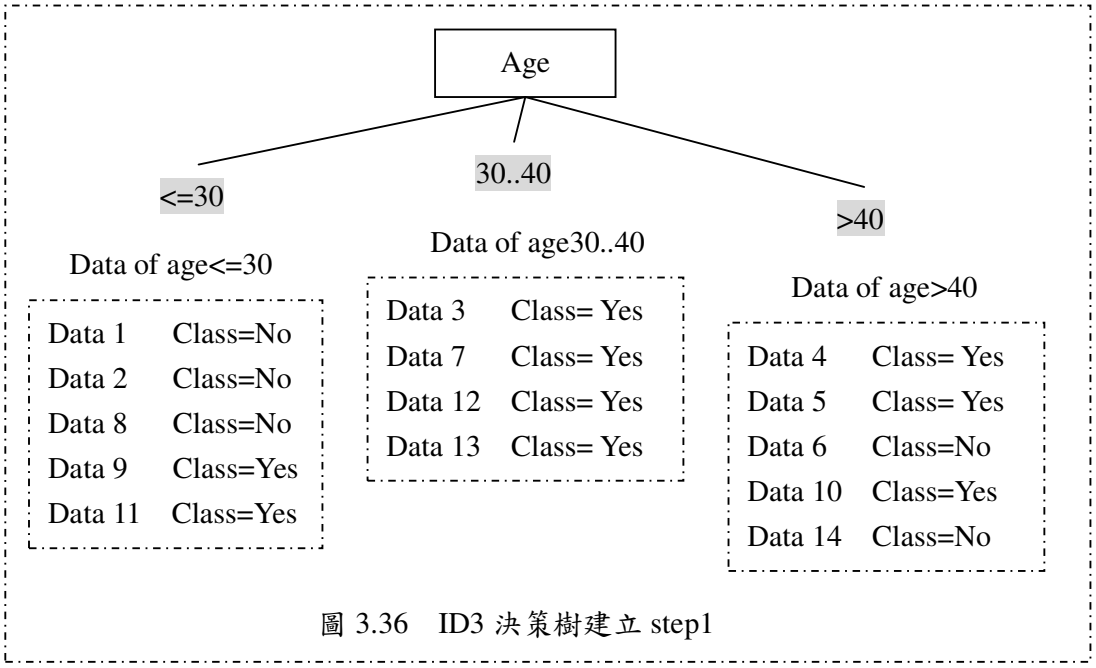


圖 3.36　ID3 決策樹建立 step1

如圖 3.36 所示，依照 age 屬性所區分的資料分為三個部份，再依決策樹建立之準則來對這三部份再次進行純化、強制純化、分裂之動作。

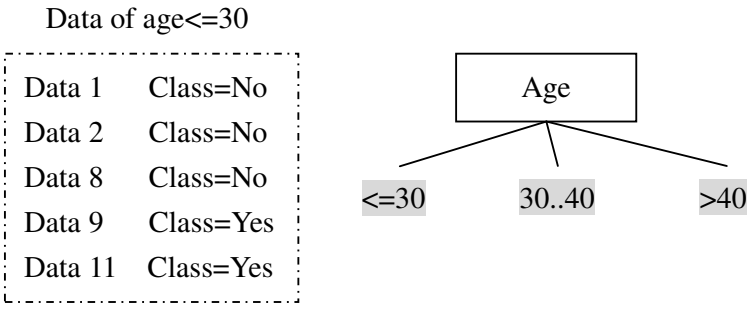


圖 3.37　ID3 決策樹建立 step2

如圖 3.37 為針對 age 屬性下<=30 之屬性值所區分資料再依據決策樹建立準則，依序進行 1. 純化、2. 強制純化、3. 分裂等測試。

表 3.19 ID3 電腦購買資料表(Age="<=30")

DataNo	income	student	credit_rating	buy_computer
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

表 3.20 ID3 電腦購買資料計次表(Age="<=30")

屬性	income			student		c_rain	
屬性值	High	Medium	Low	Yes	No	Fair	Excellent
Yes	0	1	1	2	0	1	1
No	2	1	0	0	3	2	1

4.依據 2.2 的公式我們可以計算表 3.19 age<=30 節點測試前的資訊量如下：

$$I(2,3) = (-\frac{2}{5}\log\frac{2}{5}) + (-\frac{3}{5}\log\frac{3}{5}) = 0.292$$

5.依據公式 2.2 及 2.3 我們可以計算出表 3.20 各剩餘屬性的 E(a)(測試後的資訊量)如下：

$$E(income) = \frac{0+2}{5} I(0,2) + \frac{1+1}{5} I(1,1) + \frac{1+0}{5} I(1,0) = 0.2034644023$$
$$E(student) = \frac{2+0}{5} I(2,0) + \frac{0+3}{5} I(0,3) = 0.1435359512$$
$$E(c\_rain) = \frac{1+2}{5} I(1,2) + \frac{1+1}{5} I(1,1) = 0.2912172033$$

6.依據 Information Gain 公式，測試前資訊量減掉測試後資訊量以計算各屬性作為分裂節點之優劣：

$$UseIncomeInfoGain = I(2,3) - E(income) = 0.088535597$$
$$UseStudentInfoGain = I(2,3) - E(student) = 0.148464048$$
$$UseC\_rainInfoGain = I(2,3) - E(c\_rain) = 0.0007827967$$

以上各屬性之資訊量以 Student 屬性之 Information Gain 值為最高，故在此節點以 Student 作為分裂屬性；建立如下圖之決策樹。

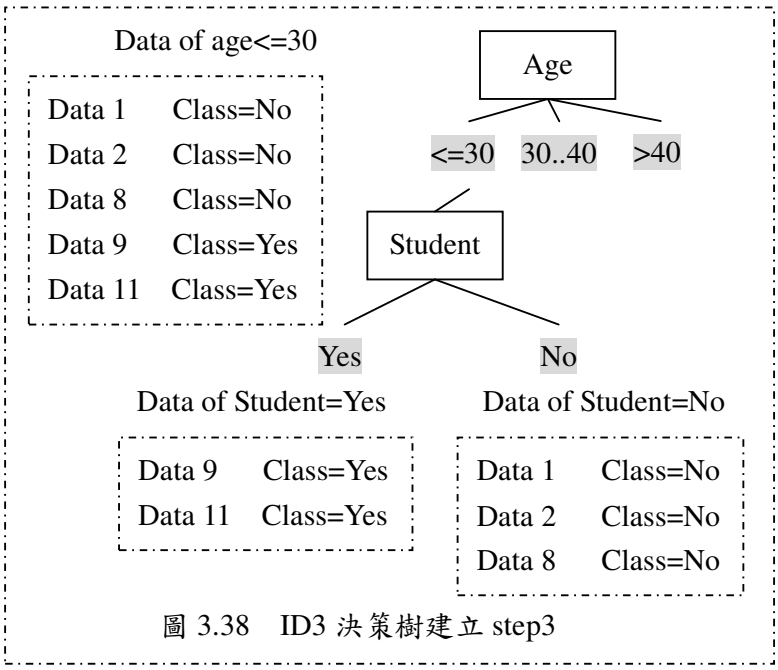


圖 3.38　ID3 決策樹建立 step3

如圖 3.38 我們依序處理 Student=Yes 屬性下之資料，依據決策樹建立準則依次測試 1. 純化、2. 強制純化、3. 分裂的動作，此時發現所區分出之資料裡已經純化，僅有單一目標類別之資料，故無須再進行分裂。

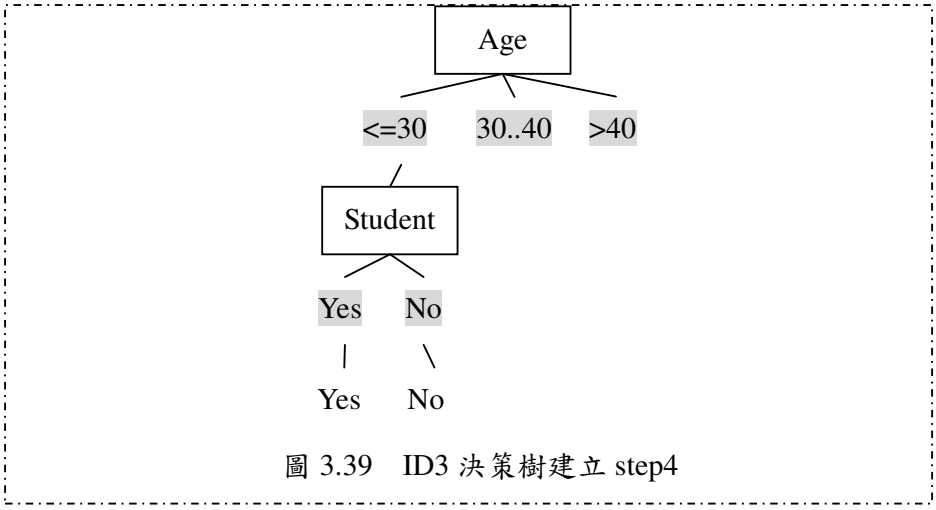


圖 3.39　ID3 決策樹建立 step4

如圖 3.39 至此我們完成了一個決策樹從根節點至葉節點之建立，其餘節點按此方法依序建立取得初始決策樹如圖 3.40。

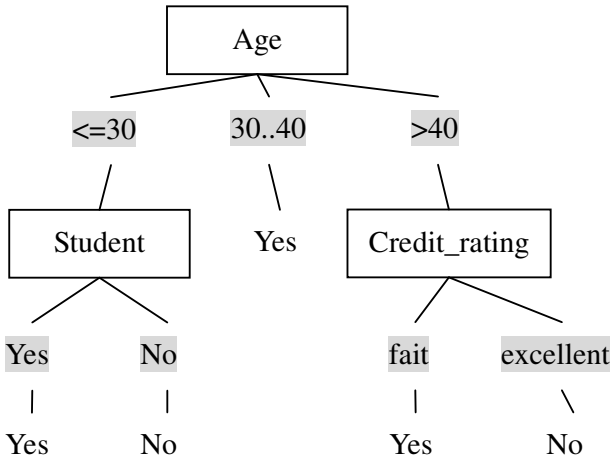


圖 3.40　ID3 初始決策樹

二、漂移案例五（多屬性單屬性值漂移）

設表 3.17 為 Bt，而下一個時間區段的資料，表 3.21 則為 Bt+1，演算法將取得 Bt+1 之計數資料，並執行卡方檢定，以判斷 Bt~Bt+1 是否產生漂移現象（檢定結果是否具顯著），且漂移案例為何（相對應之修建決策樹的方案）。

(一)、新進資料集合(Bt+1)

表 3.21 下一時間區段之電腦購買資料表（Bt+1）

age	income	student	credit_rating	buy_computer
<=30	Low	Yes	Excellent	Yes
30..40	High	Yes	Fair	No
30..40	Low	No	Excellent	Yes
<=30	Low	No	Excellent	No
<=30	Medium	No	Excellent	No
>40	Medium	Yes	Fair	No
>40	Medium	Yes	Fair	No
30..40	Low	No	Fair	Yes
30..40	Low	Yes	Fair	No
>40	Medium	Yes	Fair	No
<=30	Low	Yes	Fair	Yes
30..40	Low	No	Fair	Yes
<=30	High	Yes	Fair	No
<=30	High	Yes	Fair	Yes

(二)、資料集合的計數資料

表 3.22 下一時間區段之電腦購買資料計次表（Bt+1）

屬性	age			income			student		c_rain	
屬性值	<=30	30..40	>40	High	Medium	Low	Yes	No	Fair	Excellent
Yes	3	3	0	1	0	5	3	3	4	2
No	3	2	3	2	4	2	6	2	6	2

(三)、計數資料的卡方檢定

表 3.23 ID3 資料計次與新區塊資料計次表

Bt(ID3 資料計次)										
屬性	age			income			student		c_rain	
屬性值	<=30	30..40	>40	High	Medium	Low	Yes	No	Fair	Excellent
Yes	2	4	*3*	2	*4*	3	*6*	3	6	3
No	3	0	*2*	2	*2*	1	*1*	4	2	3
Bt+1(新區塊資料計次)										
屬性	age			income			student		c_rain	
屬性值	<=30	30..40	>40	High	Medium	Low	Yes	No	Fair	Excellent
Yes	3	3	*0*	1	*0*	5	*3*	3	4	2
No	3	2	*3*	2	*4*	2	*6*	2	6	2

$X^2_{D\to D'}(1,1) = -2.59554 < 0$ 無顯著差異。　$X^2_{D\to D'}(2,1) = -2.51109 < 0$ 無顯著差異。

$X^2_{D\to D'}(1,2) = -0.64839 < 0$ 無顯著差異。　$X^2_{D\to D'}(2,2) = 1.738903 > 0$ *有顯著差異*。

$X^2_{D\to D'}(1,3) = 0.174459 > 0$ *有顯著差異*。　$X^2_{D\to D'}(2,3) = -2.68917 < 0$ 無顯著差異。

$X^2_{D\to D'}(3,1) = 1.684482 > 0$ *有顯著差異*。　$X^2_{D\to D'}(4,1) = -0.50054 < 0$ 無顯著差異。

$X^2_{D\to D'}(3,2) = -2.36268 < 0$ 無顯著差異。　$X^2_{D\to D'}(4,2) = -2.70554 < 0$ 無顯著差異。

(四)、檢定結果的案例調整

此部份將依前項檢定結果來作案例調整，藉以達到減少重建之目的。如圖 3.41 灰色底色部份，經由計數資的卡方檢定已知灰色區塊部份產生漂移現象，此時應按漂移案例處理，案例五，藉由在原決策樹上之漂移位置。依照新的資料區塊 Bt+1 之資料對原決策樹的漂移子樹作修建動作。

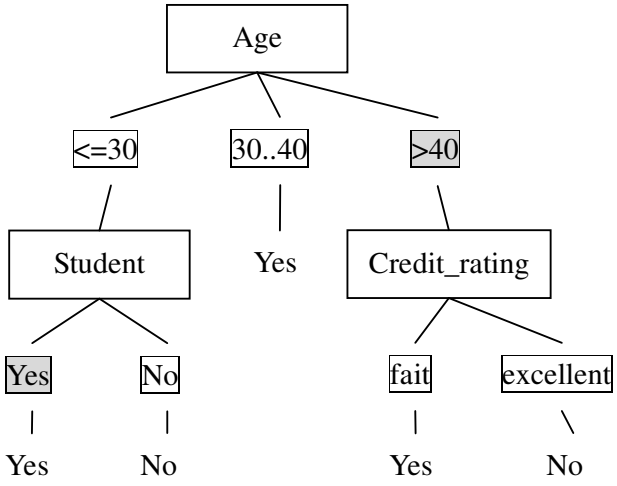


圖 3.41　ID3 初始決策樹（陰影部份為漂移部份）

表 3.23（Bt+1）資料表

age="<=30" & student="Yes"			
DataNo	income	credit_rating	buy_computer
1	Low	Excellent	Yes
11	Low	Fair	Yes
13	High	Fair	No
14	High	Fair	Yes

表 3.24（Bt+1） 計次表

age="<=30" & student="Yes"					
屬性	income			c_rain	
屬性值	High	Medium	Low	Fair	Excellent
Yes	1	0	1	2	1
No	1	0	1	1	0

如表 3.23 即為 Bt+1 在 age="<=30" & student="Yes"條件下之資料。而表 3.24 則為 age="<=30" & student="Yes"條件下之資料計次表。

表 3.25（Bt+1）

age=“>40”資料表				
DataNo	income	student	credit_rating	buy_computer
6	Medium	Yes	Fair	No
7	Medium	Yes	Fair	No
10	Medium	Yes	Fair	No

表 3.26（Bt+1）

age=“>40”計次表							
屬性	income			student		c_rain	
屬性值	High	Medium	Low	Yes	No	Fair	Excellent
Yes	0	0	0	0	0	0	0
No	0	3	0	3	0	3	0

如表 3.25 即為 Bt+1 在 age=“>40”條件下之資料。而表 3.26 則為 age=“>40”資料表條件下之資料計次表。決策樹需以這兩個表的資料進行漂移節點的重建工作。

修建步驟如下：

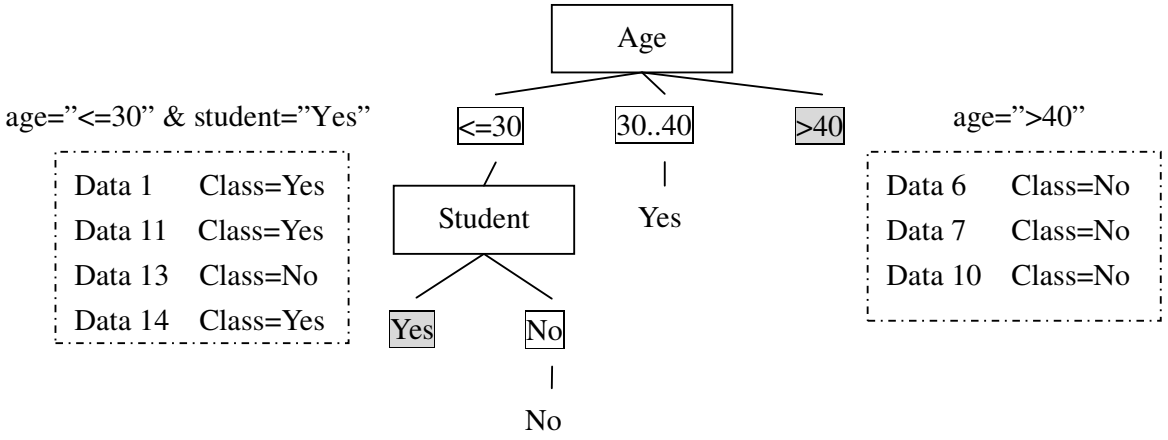


圖 3.42　決策樹 Case5 修建 step1

Step1:將產生漂移之資料區塊依照漂移之屬性區分在原決策樹之子節點，如圖 3.42。

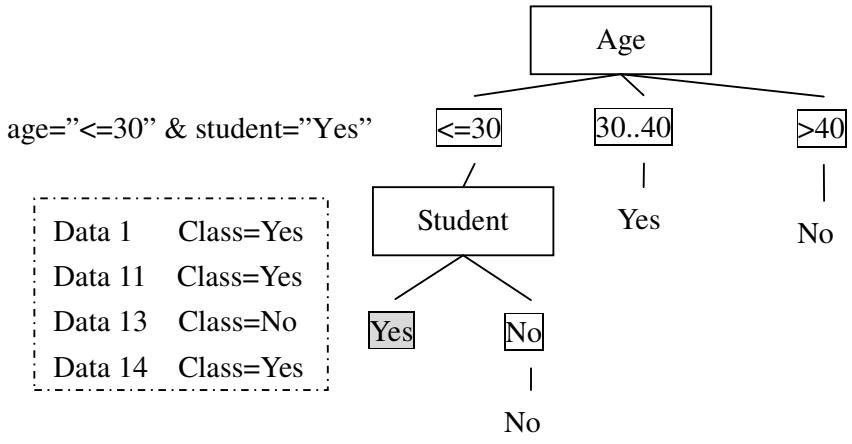


圖 3.43　決策樹 Case5 修建 step2

Step2:使用 Bt+1 資料並以決策樹建立準則：純化、強制純化、分裂來分別對各漂移屬性之子樹作調整動作，如圖 3.43 其符合 age=">40"的資料由於已經純化，故直接取得目標類別為"NO"。

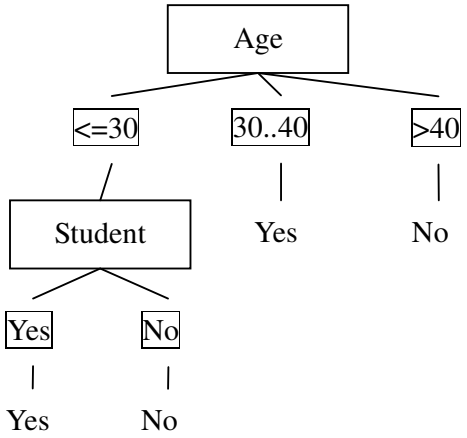


圖 3.44 調整後決策樹

Step3:使用 Bt+1 資料並以決策樹建立準則：純化、強制純化、分裂來分別對各漂移屬性之子樹作調整動作，如圖 3.44 其符合 age="<=30"&student="yes"的資料由於符合強制純化之原則，故直接取最大值之目標類別為"yes"，至此已無需調整之節點。

(五)、調整前、調整後與重建之決策樹

以下圖 3.45、圖 3.46、圖 3.47 分別為調整前、調整後、與重建之決策樹。由下圖可以看到在圖 3.45 的 age>40 的節點經由 Case5 調整後成為圖 3.46 age>40 的子節點與重建後決策樹之 age>40 部份相同;因此由圖中我們可以比較在調整後決策樹與重建之決策樹有相似提高的現象，顯示其調整具有一定的修正作用。

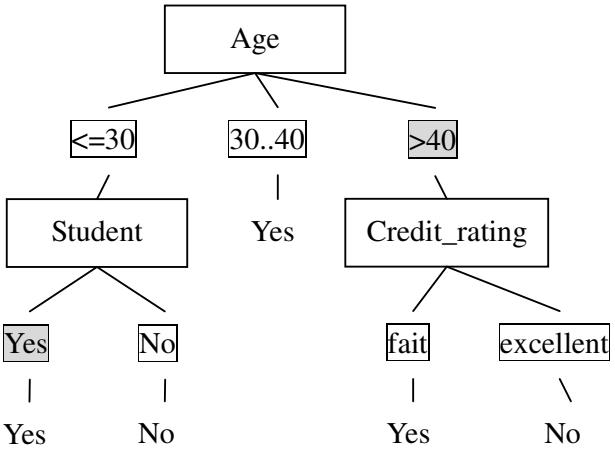


圖 3.45　調整前決策樹

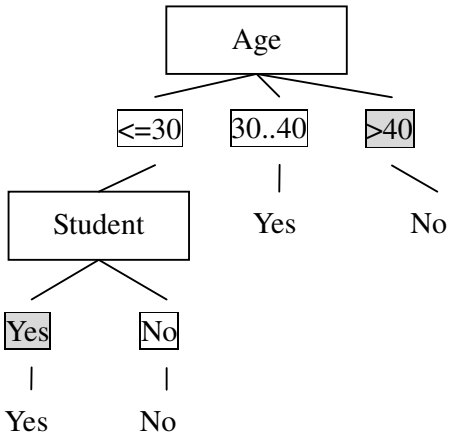


圖 3.46　調整後決策樹

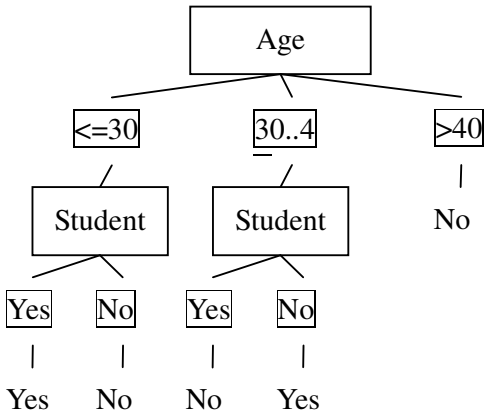


圖 3.47　重建決策樹

(六)、問題思考

本論文思考由概念元偵測之演算法可能產生的問題舉例如下：

設決策樹如圖 3.48，我們可以發現在 age>40 的部份包含了 130 筆資料，而若 age>40 以外之概念元有多屬性漂移現象，則漂移演算法即將該分類器作重建動作，然而這些漂移屬性影響分類器的程度卻不高，如此一來將造成某些不需要重建但仍重建的情況，故若能對屬性的重要性作加權動作，即可將不重要的屬性漂移現象視為相對不重要，而重要的屬性其漂移加權後得以正確反應出其影響分類器正確率的程度。

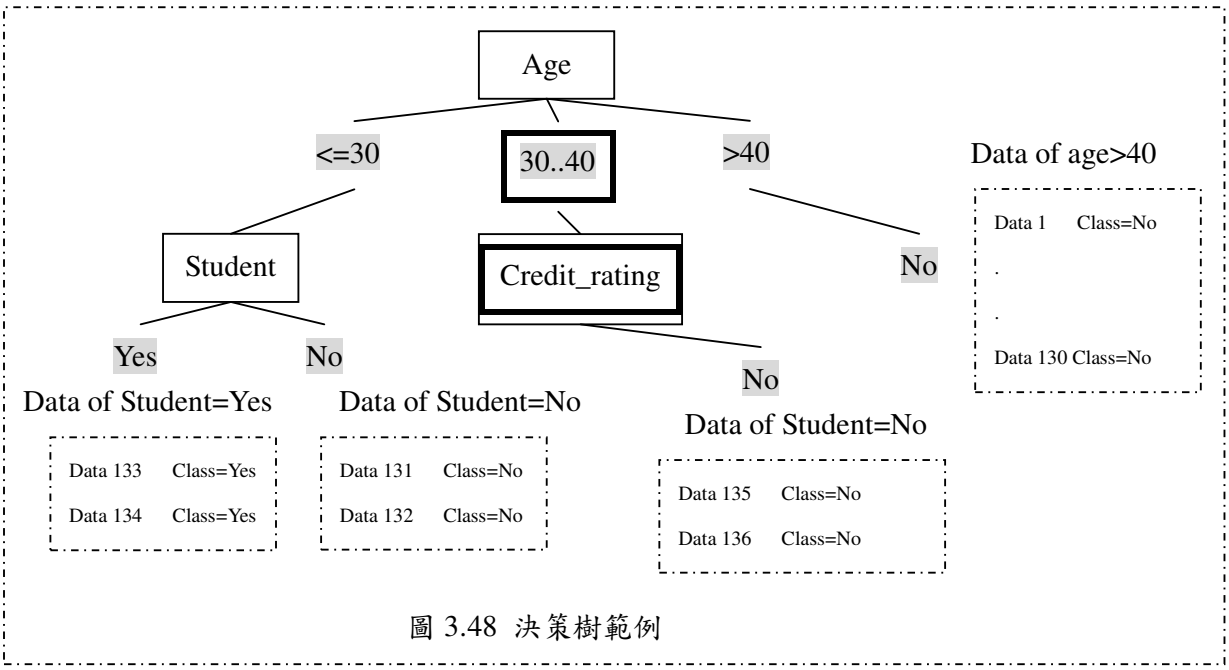


圖 3.48 決策樹範例

表 3.27 漂移屬性										
Bt										
屬性	age			income			student		c_rain	
屬性值	<=30	30..40	>40	High	Medium	Low	Yes	No	Fair	Excellent
Yes										
No										
Bt+1										
屬性	age			income			student		c_rain	
屬性值	<=30	30..40	>40	High	Medium	Low	Yes	No	Fair	Excellent
Yes										
No										
是否漂移	N	Yes	N	N	N	N	N	N	Yes	Yes

第四章、實驗分析

在本章中我們針對CDC與CDP-Tree（以下簡稱CDP）去作兩類別與四類別資料的分類分析之概念漂移偵測實驗，第一節為實驗資料，第二節為實驗設計，第三節為兩類別資料之概念漂移偵測實驗數據之分析，第四節為四類別資料之概念漂移偵測實驗數據之分析，第五節為漂移調整案例五之實驗數據分析。

第一節　實驗資料

壹、 規則制定概念：實驗資料共有 250 個區塊，且每區塊共含 500 筆資料，依此 250 個區塊區分為五個不同的區域分別代表四種不同的資料產生規則：

一、區域 1~2)區塊(1-50~51-100)：區域 1 任選適當規則作為起始資料產生規則，若規則數不足易導致分類不佳，故需視實際資料筆數與屬性而定;而區域 2 則為區域 1 的類別反相而定以此為驗證資料概念的轉換。

二、區域 2~3)區塊(51-100~101-150)：區域 3 採用改變自區域 2 之部份規則，並減少規則數以檢視規則數與規則變化對分類狀態的影響。

三、區域 3~4)區塊(101-150~151-200)：區域 4 採用改變自區域 3 的關係運算子，以觀察規則的關係運算子對分類狀態的影響。

四、區域 4~5)區塊(151-200~201-250)：區域 5 採用亂數決定其目標類別並與屬性無相關性，以檢視資料在這種情況下的分類狀態的影響；另需

注意的是由於實驗包含此一亂數區塊，故平均正確率拉低並非調整效果不佳所致，而為此區域之亂度太高所致之平均正確率下降。

貳、 規則表示方式：如表 4.1 區塊 1-50 的區間中其第一條資料為例，資料產生規則如右所示 $a1Value = a2Value \cap a1Value = 2 \rightarrow class = C1$ 式中 a1Value 表示第一個屬性之屬性值代號，而 a2Value 為第二個屬性之屬性值代號，而右箭號之右邊的 Class=C1 則表示當箭號左方之條件式成立時，則此時將產生一筆類別為 C1(Class1)之測試資料，其餘各規則皆依其表示方式進行。另在資料變異方法中則以規則產生目標類別，並 1%與 5%的機率來隨機產生目標類別，藉此作 1%與 5%的資料變異。

參、 兩類別實驗資料：依照表 4.1 之產生規則及 1%與 5%資料變異程度來產生，每次實驗之實驗資料產生 250 個資料區塊，而每區塊共有 500 筆資料，每筆資料有四個屬性，屬性 1、2 為 3 個屬性值，屬性 2、3 為 2 個屬性值，分為兩類別。

表4.1兩類別實驗資料產生規則

區塊區間	資料產生規則
1-50	$a1Value = a2Value \cap a1Value = 2 \rightarrow class = C1$ $a1Value <> a3Value \rightarrow class = C2$ $a2Value > a4Value \cap a2Value = 1 \rightarrow class = C2$ $a2Value > a4Value \cap a2Value <> 1 \rightarrow class = C1$ $a1Value < a3Value \rightarrow class = C2$ $a1Value > a3Value \rightarrow class = C1$
51-100	$a1Value = a2Value \cap a1Value = 2 \rightarrow class = C2$ $a1Value <> a3Value \rightarrow class = C1$ $a2Value > a4Value \cap a2Value = 1 \rightarrow class = C1$ $a2Value > a4Value \cap a2Value <> 1 \rightarrow class = C2$ $a1Value < a3Value \rightarrow class = C1$ $a1Value > a3Value \rightarrow class = C2$
101-150	$a1Value = a2Value \rightarrow class = C1$ $a1Value < a3Value \rightarrow class = C2$
151-200	$a1Value <> a2Value \rightarrow class = C1$ $a1Value > a3Value \rightarrow class = C2$
201-250	亂數取得class1 or class2

肆、 四類別實驗資料：依照表 4.2a 及表 4.2b 之產生規則及 1%與 5%資料變異程度來產生，每次實驗之實驗資料為 250 個資料區塊，而每區塊共有 500 筆資料，每筆資料有四個屬性，屬性 1、2 為 3 個屬性值，屬性 2、3 為 2 個屬性值，共可區分為四種類別。

(a)資料產生規則(1-50、51-100)

區塊區間	資料產生規則
1-50	$a1Value = a2Value \cap a1Value = 2 \rightarrow class = C1$ $a1Value <> a3Value \rightarrow class = C2$ $a2Value > a4Value \cap a2Value = 1 \rightarrow class = C2$ $a2Value > a4Value \cap a2Value <> 1 \rightarrow class = C1$ $a1Value < a3Value \rightarrow class = C2$ $a1Value > a3Value \rightarrow class = C1$ $a1Value <> a2Value \cap a1Value = 2 \rightarrow class = C3$ $a1Value = a3Value \rightarrow class = C4$ $a2Value < a4Value \cap a2Value = 1 \rightarrow class = C3$ $a2Value < a4Value \cap a2Value <> 1 \rightarrow class = C4$ $a1Value > a3Value \rightarrow class = C3$ $a1Value < a3Value \rightarrow class = C4$
51-100	$a1Value = a2Value \cap a1Value = 2 \rightarrow class = C2$ $a1Value <> a3Value \rightarrow class = C1$ $a2Value > a4Value \cap a2Value = 1 \rightarrow class = C1$ $a2Value > a4Value \cap a2Value <> 1 \rightarrow class = C2$ $a1Value < a3Value \rightarrow class = C1$ $a1Value > a3Value \rightarrow class = C2$ $a1Value <> a2Value \cap a1Value = 2 \rightarrow class = C3$ $a1Value = a3Value \rightarrow class = C4$ $a2Value < a4Value \cap a2Value <> 1 \rightarrow class = C3$ $a2Value < a4Value \cap a2Value = 1 \rightarrow class = C4$ $a1Value > a3Value \rightarrow class = C3$ $a1Value < a3Value \rightarrow class = C4$

(b)資料產生規則(101-150、151-200、201-250)

區塊區間	資料產生規則
101-150	$a1Value = a2Value \rightarrow class = C1$ $a1Value < a3Value \rightarrow class = C2$ $a1Value <> a2Value \rightarrow class = C3$ $a1Value > a3Value \rightarrow class = C4$
151-200	$a1Value <> a2Value \rightarrow class = C1$ $a1Value > a3Value \rightarrow class = C2$ $a1Value = a2Value \rightarrow class = C3$ $a1Value < a3Value \rightarrow class = C4$
201-250	亂數取得 class1 ~class4

表 4.2 四類別實驗資料產生規則

第二節　實驗設計

本研究給予資料某些規則性來產生所需之實驗資料，以每 1 區塊為概念漂移的區間而每 50 個區塊為一個概念轉換的區間，其產生數據的規則如第四章第一節所述。在第三節、第四節中，本研究設計讓兩方法(CDC&CDP)在由上述產生規則之兩類別資料與四類別資料在不同顯著水準與不同的資料變異下分別評估其「平均正確率」與「未調整比率」以及對個別之「完全重建平均正確率」作分析，以比較兩方法在兩類別資料與在四類別（多類別）分類分析之概念漂移偵測上是否有所差異（由於此處比較重點在偵測方法，故將漂移案例五於第五節中另外分析）。並以三種不同的信心水準來比較兩方法(CDC&CDP)對不同信心水準下是否有漸近或漸遠於重建正確率的情況。另第五節中將以多屬性漂移下之兩個調整案例來分析，其案例五與案例四在多屬性漂移下之比重。第五節中就案例五的調整方法來分析其各資料變異下之正確率分佈，藉以驗證其案例五之可用性。

壹、公式 4.1：為本研究為評估兩方法在不同因素下之綜合評估而設計之公式，其分子為每次實驗內之 250 個區塊的實際漂移案例乘上各別案例實際平均正確率再除以全部共 250 個區塊之平均每區塊正確率。

貳、公式 4.2：分子為案例 0 的區塊次數，而分母為總區塊數；其為本研究為評估在兩方法在一定的平均正確率下其未漂移的區塊數目佔用全部區塊的比例，藉以初步評估兩方法在調整時其正確率與重整修建的相互關係，其值在相同平均正確率下未調整比率愈高愈好，代表耗用重整修建的成本相對較低。

$$\text{平均正確率} = \frac{\sum_{i=0}^{4} Counti * Correcti\%}{BT\_Count} \qquad \text{公式 4.1}$$

$$\text{未調整率} = \frac{Count0}{Bt\_Count} \qquad \text{公式 4.2}$$

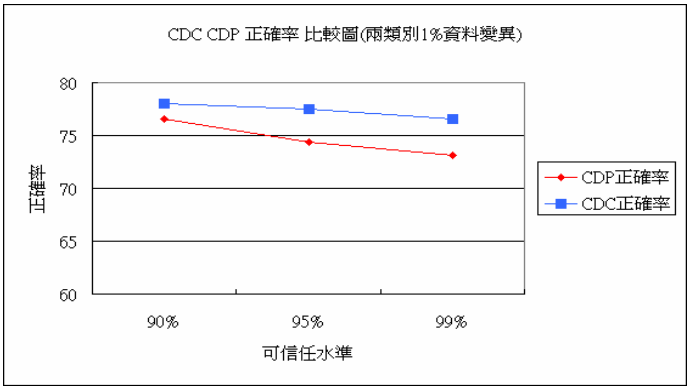
第三節 兩類別資料之概念漂移偵測

在第三節中我們在三種信任水準下(90%、95%、99%)分別對 CDC 與 CDP 進行在正確率、未調整比率、完全重建正確率之綜合實驗。如圖 4.1ab 我們在 CDC 與 CDP 的每一信任水準下各作 10 次實驗，並平均 10 次實驗數據；而實驗資料依前述產生規則及資料變異程度來產生，每次實驗之實驗資料為 250 個資料區塊，而每區塊共有 500 筆資料，每筆資料有四個屬性，屬性 1、2 為 3 個屬性值，屬性 2、3 為 2 個屬性值，共可區分為兩種類別。

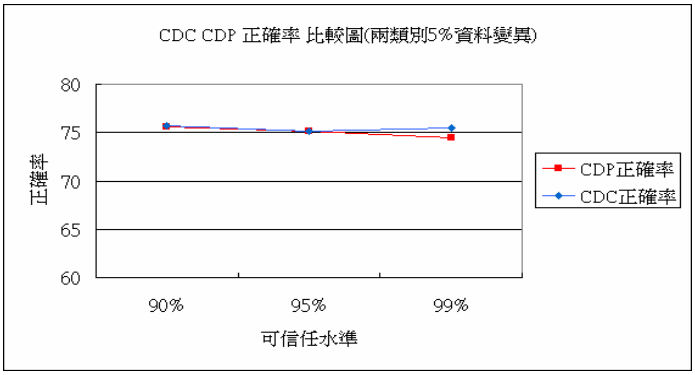
壹、兩類別資料正確率與未調整率分析

一、兩類別 CDC 與 CDP 之正確率比較：

如圖 4.1a 與圖 4.1b 為 CDC 與 CDP 依不同資料變異下分別在不同顯著水準下之正確率的關係，在圖 4.1a 中 1%資料變異中 CDC 之正確率均在 CDP 之上，而 4.1b 之 5%資料變異則 CDC 隨顯著水準提升而正確率有成長傾向，而 CDP 則略微下降，依以上所述我們可以發現在兩類別資料下當資料變異變大時，CDC 並沒有足夠的證據證明其正確率高於 CDP，但 CDC 有隨著可信任水準提高而正確率隨之提高或持平的現象。



(a)CDC/CDP 正確率比較圖(兩類別 1%資料變異)

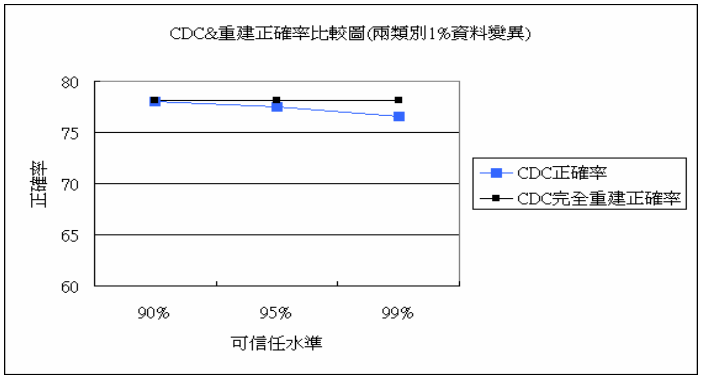


(b)CDC/CDP 正確率比較圖(兩類別 5%資料變異)

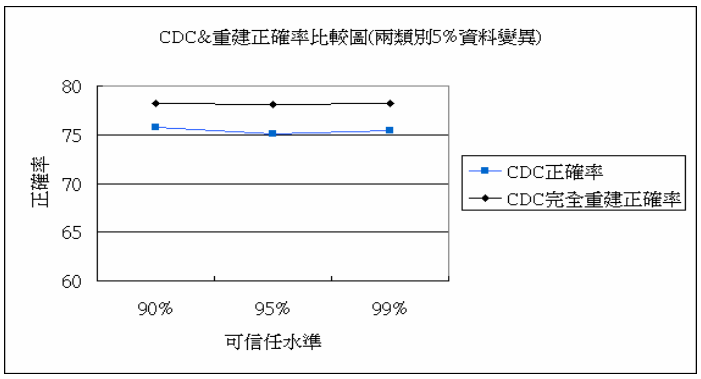
圖 4.1.為 CDC/CDP 正確率比較圖(分別在 1%與 5%資料變異下)

二、兩類別 CDC 與重建之正確率比較：

如圖 4.2a 顯示出在兩類別 1%資料變異下 CDC 之正確率有向下的傾向但比較其圖 4.3a 時我們卻發現 CDC 其正確率下降的傾向較圖 4.3a 為緩，顯示 CDC 在這種情況下有較佳之抗壓性，而在圖 4.2b 中 5%的資料變異下 CDC 有隨可信任水準提高而正確率隨之提高之傾向。



(a) CDC/重建 正確率比較圖(兩類別 1%資料變異)

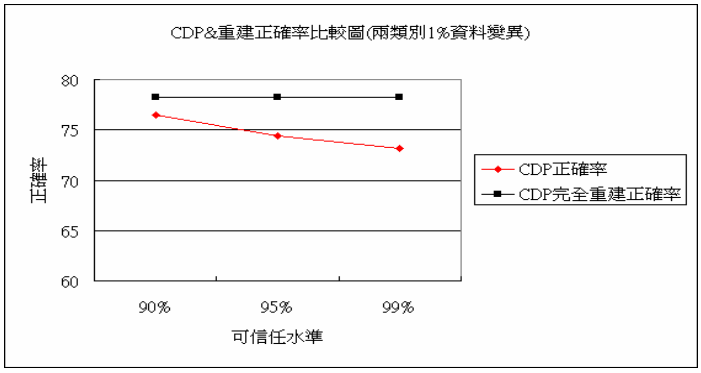


(b) CDC/重建 正確率比較圖(兩類別 5%資料變異)

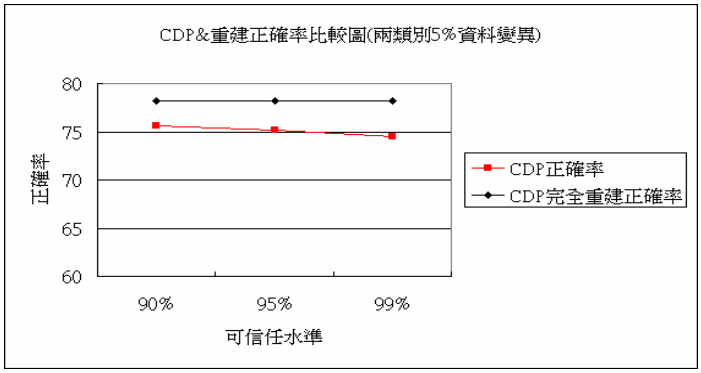
圖 4.2.為 CDC/重建正確率比較圖(分別在 1%與 5%資料變異下)

三、兩類別 CDP 與重建之正確率比較：

圖 4.3a 與圖 4.3b 均顯示出 CDP 在兩類別的資料 1%或 5%資料變異下，其正確率均隨著可信水準提高而導致正確率下降，且我們再比較圖 4.3a 與圖 4.2a 可以觀察到 CDC 之正確率下降的速度較 CDP 為緩，顯示 CDC 有較佳的適用度，另比較圖 4.3b 與 4.2b 可以發現在可信任水準提高到 99%時 CDP 之正確率有下降的傾向，而 CDC 則為上升或持平的傾向。



(a) CDP/重建 正確率比較圖(兩類別 1%資料變異)

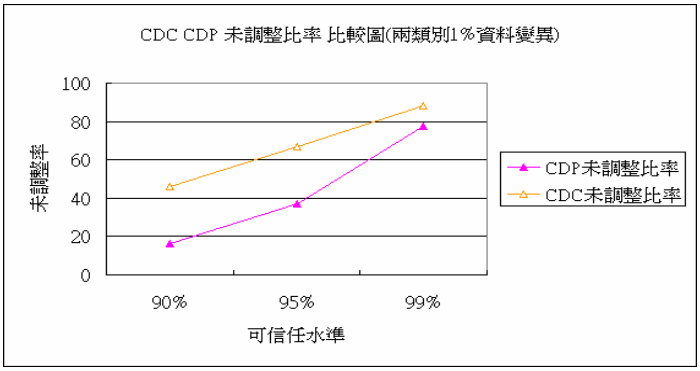


(b) CDP/重建 正確率比較圖(兩類別 5%資料變異)

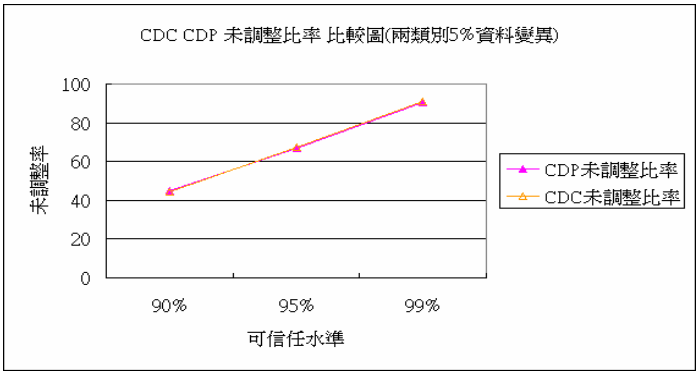
圖 4.3.為 CDP/重建正確率比較圖(分別在 1%與 5%資料變異下)

四、兩類別 CDC 與 CDP 之未調整比率比較：

另如圖 4.4a 我們可以看到在 1%顯著水準下 CDC 相較於 CDP 有較多的未漂移案例，而 CDP 則較常作調整重建等機制。而圖 4.4b 則顯示兩者的未調整率並沒有顯著不同，此時我們可以再去作更細部的分析其案例分佈，以實驗兩演算方式之異同。



(a) CDC/CDP 未調整率比較圖(兩類別 1%資料變異)

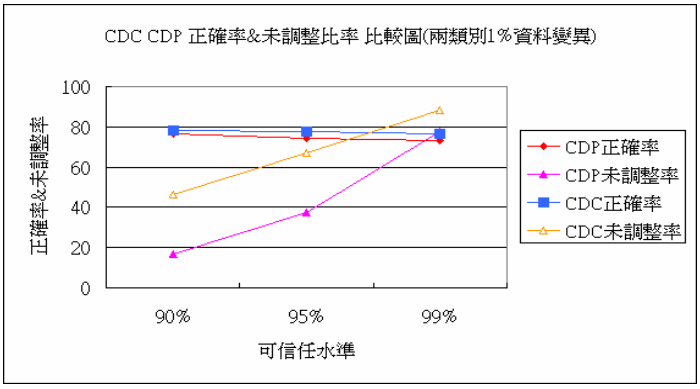


(b) CDC/CDP 未調整率比較圖(兩類別 5%資料變異)

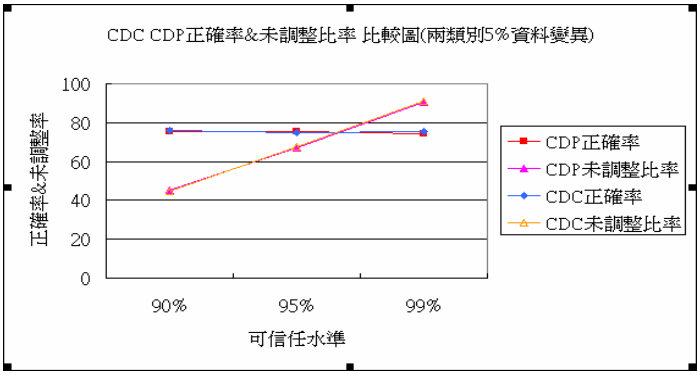
圖 4.4.為 CDC/CDP 未調整率比較圖(分別在 1%與 5%資料變異下)

五、兩類別 CDC 與 CDP 正確率與未調整比率之比較：

如圖 4.5a 顯示出在 1%資料變異時 CDC 比 CDP 在相似的正確率下有更多的未調整案例，意即其花費在修建、重建之成本較 CDP 為少；而圖 4.5b 無論正確率亦或之未調整率在兩方法中均相似，意即兩方法在 5%資料變異下沒有顯著的差異，但可再探究其實際調整案例的調整耗用成本大小。



(a) CDC/CDP 正確率/未調整率比較圖(兩類別 1%資料變異)



(b) CDC/CDP 正確率/未調整率比較圖(兩類別 5%資料變異)

圖 4.5.為 CDC/CDP 正確率/未調整率比較圖(分別在 1%與 5%資料變異下)

六、綜合以上：

（一）、可發現兩類別資料與 1%資料變異下 CDC 的正確率與未調整率優於 CDP。

（二）、而在兩類別資料 5%變異下 CDC 的正確率或未調整比率與 CDP 並沒有顯著的不同。

故在兩類別資料下 CDC 並沒有較 CDP 有優勢，但其實際仍可以進一步作調整案例的分析來比較兩方法所耗用的調整成本（圖 4.5 為 CDC/CDP 正確率與未調整率的綜合比較，圖 4.5 所顯示的意義為在三種可信水準下，CDC/CDP 兩者並沒有顯著差異，且隨著可信水準的提升，可有效的降低訓練成本，而不致使正確率有明顯下降）。

貳、兩類別資料案例個數分析

經由前述兩類別資料之正確率與未調整率分析，我們可以歸納出在兩類別資料中 CDC 與 CDP 這兩種方法之正確率並沒有顯著差異，因此我們再作案例數分析以分析在訓練成本方面是否有顯著差異；如表 4.3(a)、4.3(b)、4.3(c)、表 4.4 (a)、4.4(b)、4.4(c)分解如下所示，可以發現以下各點特性：

一、隨顯著水準之提升，Case0(未漂移案例，故不需耗用修建成本)增多而 Case4 變少（多屬性漂移案例少，表示重建成本少），顯示我們可以隨訓練成本的提高而使用更高之顯著水準來降低訓練所花費的總成本。如表 4.3 Case0 在區域 1-50、51-100、101-150、151-200、201-250

均呈現出隨顯著水準提高而提高，表格內的數字表示每區域 50 個區塊判定為 Case0 的個數。(愈多依照先前的正確率分析，可得在相同的正確率下，可以較節省修建成本)

表 4.3 兩類別資料 5%變異(平均數)之 Case0 案例表

兩類別資料5%變異之漂移案例數(平均數)

區域區間		1-50		51-100		101-150		151-200		201-250	
偵測方法		CDC	CDP	CDC	CDP	CDC	CDP	CDC	CDP	CDC	CDP
CASE0	90%	22.8	22.7	23.6	20.9	22.5	21.6	23.8	23.5	23	23.6
	95%	35	37.7	30.7	31.5	34.9	32.2	32.6	32.8	36	33
	99%	48	46.9	42.8	43.8	47	45.9	42.8	45.5	48	45

二、如表 4.3 由於區域 5（201-250）為亂數產生之資料，然而兩種演算法之漂移案例並沒有因此產生劇烈的變化，顯示漂移案例分配與資料重疊度（資料亂度）沒有直接的關係（資料亂度影響的是正確率）。

三、如表 4.3 所示 CDC 與 CDP 在五個漂移規則的區域之中無特定案例突增或突降的情況發生，顯示兩種方法的偵測並沒有因為不同的資料規則而有不穩定的情況。

四、在兩類別實驗中 Case1、Case3 發生的最少，按直覺判斷為實驗屬性 4 種，而屬性值至多 3 種所引起；屬性愈少，則 Case1(未在決策樹上之屬性漂移亦愈少)；同一屬性之屬性值愈少就愈難出現 Case3(單屬性多屬性值漂移)。

表 4.4 兩類別資料 5%變異(平均數)之區域 201-250 案例表

案例漂移														區域區間	
CASE4			CASE3			CASE2			CASE1			CASE0			偵測方法	
99%	95%	90%	99%	95%	90%	99%	95%	90%	99%	95%	90%	99%	95%	90%		
1	4.9	13.8	0	0.6	1	1	8.5	11.8	0	0	0.4	48	36	23	CDC	201-250
1.5	5.6	12.7	0	0.3	0.6	3.5	11.1	12.9	0	0	0.2	45	33	23.6	CDP	

五、綜言之在 3 種顯著水準下之評估之結果顯示，雖在 1%資料變異下平均數、最大數、最小數中 CDC 之 Case0（未漂移案例）有稍高於 CDP 之情形，但以 5%變異資料來說兩方法並無差異，顯示這兩種方法在兩類別的漂移偵測上測出之案例(調整成本)並沒有較大差異性。

(a)兩類別資料 1%變異之漂移案例數(平均數)

兩類別資料1%變異之漂移案例數(平均數)

區域區間			1-50		51-100		101-150		151-200		201-250	
偵測方法			CDC	CDP	CDC	CDP	CDC	CDP	CDC	CDP	CDC	CDP
漂移案例	CASE0	90%	23.1	22	21.3	20.7	24.2	19.8	23.3	25.3	22.1	20.7
		95%	33	32.4	33.9	32.9	31.4	30.3	36.2	36.1	33.2	32.2
		99%	45	46.8	43.8	45.6	43.3	43.3	42.6	46.4	45.5	44.5
	CASE1	90%	1.3	2.4	1.3	1.8	0	0	0	0	0	0
		95%	1.4	0.9	0.4	0.8	0	0	0	0.2	0	0
		99%	0.4	0.6	0.2	0	0	0	0.2	0	0	0
	CASE2	90%	14.1	12.7	13.8	10.8	11.4	14.4	9.7	10.5	14.2	14.5
		95%	8.8	11.4	10.1	9.6	11.3	11.8	7.7	6.7	10.5	11.6
		99%	3.3	2.3	4	3.3	4.6	4.5	4.1	2	3.1	3.9
	CASE3	90%	0.9	1	2.3	1.6	1.2	1.7	0.7	1	1.5	1.3
		95%	0.2	0.1	0.7	0.7	0.4	0.6	0.2	0.4	0.2	0.4
		99%	0.1	0	0.2	0	0	0	0.1	0	0	0
	CASE4	90%	10.6	11.9	11.3	15.1	13.2	14.1	16.3	13.2	12.2	13.5
		95%	6.6	5.2	4.9	6	6.9	7.3	5.9	6.6	6.1	5.8
		99%	1.2	0.3	1.8	1.1	2.1	2.2	3	1.6	1.4	1.6

(b)兩類別資料 1%變異之漂移案例數(最大數)

兩類別資料1%變異之漂移案例數(最大數)

區域區間			1-50		51-100		101-150		151-200		201-250	
偵測方法			CDC	CDP	CDC	CDP	CDC	CDP	CDC	CDP	CDC	CDP
漂移案例	CASE0	90%	31	30	30	29	29	29	31	29	33	28
		95%	44	47	45	40	44	42	44	44	45	45
		99%	50	49	48	49	47	49	49	49	49	49
	CASE1	90%	6	6	6	5	0	0	0	0	0	0
		95%	3	4	1	2	0	0	0	2	0	0
		99%	1	2	1	0	0	0	2	0	0	0
	CASE2	90%	19	19	20	15	15	24	14	20	19	21
		95%	16	22	24	13	15	17	15	12	17	22
		99%	11	4	11	14	9	12	12	6	9	8
	CASE3	90%	2	3	6	5	3	5	3	3	5	4
		95%	1	1	3	2	2	2	1	1	2	1
		99%	1	0	1	0	0	0	1	0	0	0
	CASE4	90%	16	18	17	23	20	20	22	19	17	19
		95%	10	11	8	11	11	11	11	12	10	11
		99%	5	2	4	2	6	7	6	4	5	4

(c)兩類別資料 1%變異之漂移案例數(最小數)

兩類別資料1%變異之漂移案例數(最小數)

區域區間			1-50		51-100		101-150		151-200		201-250	
偵測方法			CDC	CDP	CDC	CDP	CDC	CDP	CDC	CDP	CDC	CDP
漂移案例	CASE0	90%	18	15	14	16	17	11	19	21	13	16
		95%	23	22	17	28	28	25	30	29	24	22
		99%	38	44	36	34	36	35	35	40	40	38
	CASE1	90%	0	0	0	0	0	0	0	0	0	0
		95%	0	0	0	0	0	0	0	0	0	0
		99%	0	0	0	0	0	0	0	0	0	0
	CASE2	90%	10	8	8	5	7	9	7	6	5	11
		95%	3	3	2	6	5	7	3	4	4	4
		99%	0	0	1	0	2	0	0	0	0	0
	CASE3	90%	0	0	0	0	0	0	0	0	0	0
		95%	0	0	0	0	0	0	0	0	0	0
		99%	0	0	0	0	0	0	0	0	0	0
	CASE4	90%	7	3	4	8	7	5	9	5	9	10
		95%	2	0	1	3	1	1	1	1	1	1
		99%	0	0	1	1	1	1	1	1	1	1

表 4.5 兩類別資料 1%變異之漂移案例數(平均、最大、最小數)

(a)兩類別資料 5%變異之漂移案例數(平均數)

兩類別資料5%變異之漂移案例數(平均數)

區域區間			1-50		51-100		101-150		151-200		201-250	
偵測方法			CDC	CDP	CDC	CDP	CDC	CDP	CDC	CDP	CDC	CDP
漂移案例	CASE0	90%	22.8	22.7	23.6	20.9	22.5	21.6	23.8	23.5	23	23.6
		95%	35	37.7	30.7	31.5	34.9	32.2	32.6	32.8	36	33
		99%	48	46.9	42.8	43.8	47	45.9	42.8	45.5	48	45
	CASE1	90%	2.1	2.1	1.4	0.7	0.4	0.2	0.4	0.2	0.4	0.2
		95%	0.4	0.6	0.7	0.6	0	0	0	0.3	0	0
		99%	0	0.1	0	0.3	0	0	0	0	0	0
	CASE2	90%	12.6	11.2	12.1	14.9	12.4	13.8	12.1	11	11.8	12.9
		95%	9.6	8.9	9.4	11.5	8.1	11.2	10.4	9.8	8.5	11.1
		99%	2	2.6	0	4.3	2	2.8	0	2.3	1	3.5
	CASE3	90%	0.9	1	1.3	1.5	1.3	1.6	0.6	0.9	1	0.6
		95%	0.7	0.4	0.2	0.2	0.6	0.7	0.3	0.1	0.6	0.3
		99%	0	0	0	0	0	0	0	0	0	0
	CASE4	90%	11.6	13	11.6	12	13.4	12.8	13.1	14.4	13.8	12.7
		95%	4.3	2.4	9	6.2	6.4	5.9	6.7	7	4.9	5.6
		99%	0	0.3	0	1.6	1	1.3	0	2.2	1	1.5

(b)兩類別資料 5%變異之漂移案例數(最大數)

兩類別資料5%變異之漂移案例數(最大數)

區域區間			1-50		51-100		101-150		151-200		201-250	
偵測方法			CDC	CDP	CDC	CDP	CDC	CDP	CDC	CDP	CDC	CDP
漂移案例	CASE0	90%	30	33	30	24	30	35	30	31	30	30
		95%	45	48	38	43	47	42	39	37	43	43
		99%	50	50	49	49	49	49	48	48	49	48
	CASE1	90%	8	6	2	2	2	2	2	2	2	2
		95%	1	6	3	4	0	0	0	2	0	0
		99%	0	1	1	2	0	0	0	0	0	0
	CASE2	90%	21	15	20	20	18	18	18	16	16	17
		95%	16	20	14	21	16	20	18	14	19	17
		99%	6	11	12	11	6	9	7	7	11	8
	CASE3	90%	2	2	4	3	4	8	3	3	2	1
		95%	2	1	1	1	2	3	1	1	2	1
		99%	0	0	0	0	0	0	0	0	0	0
	CASE4	90%	16	20	16	20	16	20	16	20	17	20
		95%	11	6	14	15	10	10	13	10	11	10
		99%	2	2	5	4	5	3	3	6	4	4

(c)兩類別資料 5%變異之漂移案例數(最小數)

兩類別資料5%變異之漂移案例數(最小數)

區域區間			1-50		51-100		101-150		151-200		201-250	
偵測方法			CDC	CDP	CDC	CDP	CDC	CDP	CDC	CDP	CDC	CDP
漂移案例	CASE0	90%	16	16	17	16	14	15	20	13	16	19
		95%	27	24	23	22	24	23	22	28	24	24
		99%	42	37	34	35	39	39	40	37	35	39
	CASE1	90%	0	0	0	0	0	0	0	0	0	0
		95%	0	0	0	0	0	0	0	0	0	0
		99%	0	0	0	0	0	0	0	0	0	0
	CASE2	90%	7	7	7	8	7	7	7	8	7	8
		95%	3	1	6	5	2	4	4	5	2	5
		99%	0	0	0	0	0	0	1	0	0	1
	CASE3	90%	0	0	0	0	0	0	0	0	0	0
		95%	0	0	0	0	0	0	0	0	0	0
		99%	0	0	0	0	0	0	0	0	0	0
	CASE4	90%	7	7	7	5	11	7	9	5	9	7
		95%	0	0	4	2	1	1	2	4	1	1
		99%	0	0	1	0	1	1	1	1	1	1

表 4.6 兩類別資料 5%變異之漂移案例數(平均、最大、最小數)

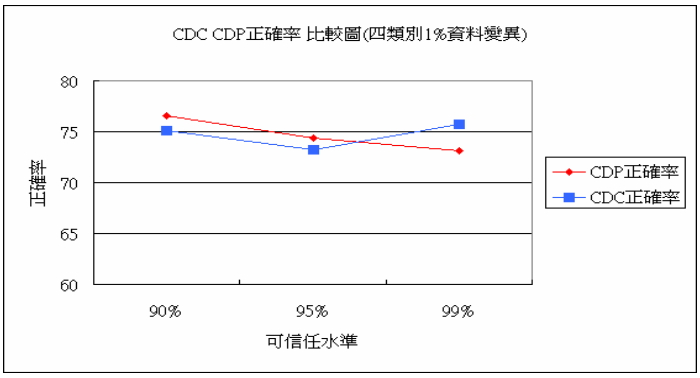
第四節 四類別資料之概念漂移偵測

在第四節中我們在三種可信任水準下（90%、95%、99%）分別對CDC與CDP進行在平均正確率、未調整比率、完全重建平均正確率之綜合實驗。如圖4.6a與4.6b裡，我們在CDC與CDP的每一顯著水準下各作10次實驗，並平均10次實驗數據；而實驗資料依前述產生規則及資料變異程度來產生，每次實驗之實驗資料為250個資料區塊，而每區塊共有500筆資料，每筆資料有四個屬性，屬性1、2為3個屬性值，屬性2、3為2個屬性值，共可區分為四種類別。

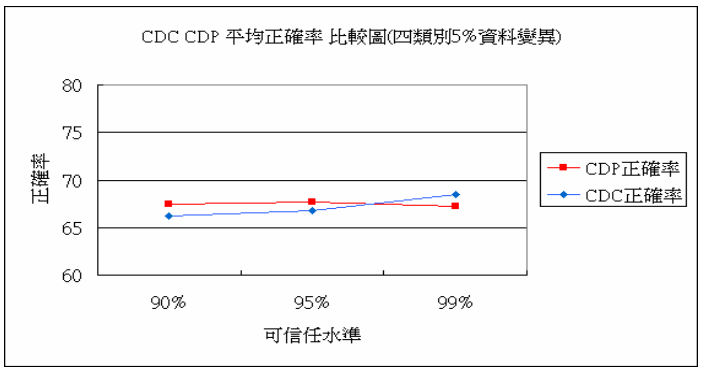
壹、四類別資料正確率與未調整率分析

一、四類別CDC與CDP之正確率比較：

如圖4.6a與圖4.6b，圖中無論在1%或5%資料變異下，CDC均隨著顯著水準之上升而平均正確率也隨之上升，反之CDP卻有向下趨勢，故我們發現CDC在99%顯著水準下其正確率比CDP高，且在高可信任水準下有較適用於多類別分類分析之概念漂移偵測上之現象。



(a)CDC/CDP 正確率比較圖(四類別 1%資料變異)

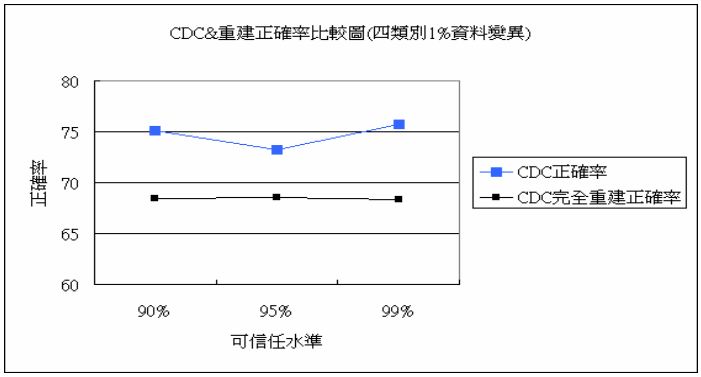


(b)CDC/CDP 正確率比較圖(四類別 5%資料變異)

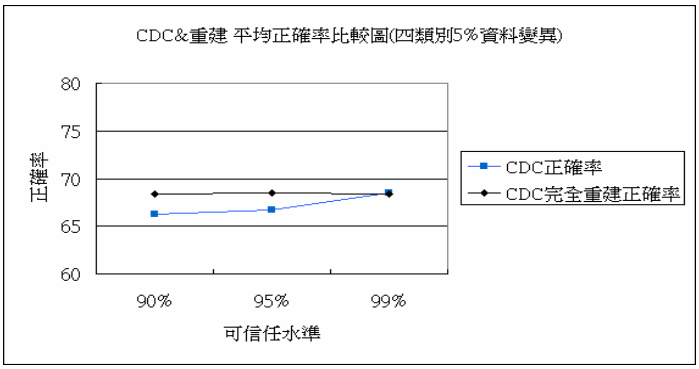
圖4.6.為CDC/CDP正確率比較圖(分別在1%與5%資料變異下)

二、四類別CDC與重建之正確率比較

在圖 4.7a 中由於資料變異程度相當低僅有 1%，故已經建立之決策樹不需要常變動就已經能應付後續資料的分類需求，因此其平均正確率相當高；而在圖 4.7b 中所示 CDC 在資料變異度 5%情形下之平均正確率隨著顯著水準的提高 CDC 之平均正確率亦隨之提高而接近「完全重建之正確率」，故我們可推估 CDC 在多類別分類分析之概念漂移偵測上之可用性。



(a) CDC/重建 正確率比較圖(四類別 1%資料變異)

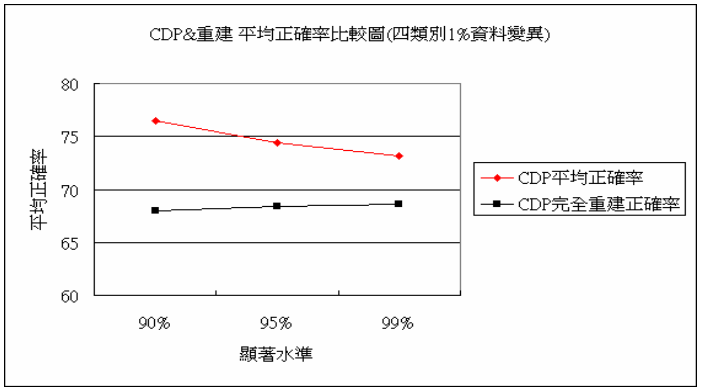


(b) CDC/重建 正確率比較圖(四類別 5%資料變異)

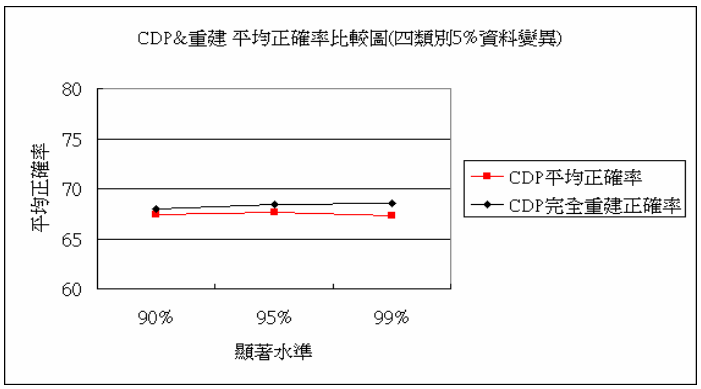
圖 4.7.為 CDC/重建正確率比較圖(分別在 1%與 5%資料變異下)

三、四類別 CDP 與重建之正確率比較：

如圖 4.8a 隨著顯著水準的提高，平均正確率卻有隨之下降的傾向，且如圖 4.8b 為 CDP 在 5%資料變異度下隨著顯著水準的提高，而平均正確率卻隨之下降。



(a) CDP/重建 正確率比較圖(四類別 1%資料變異)

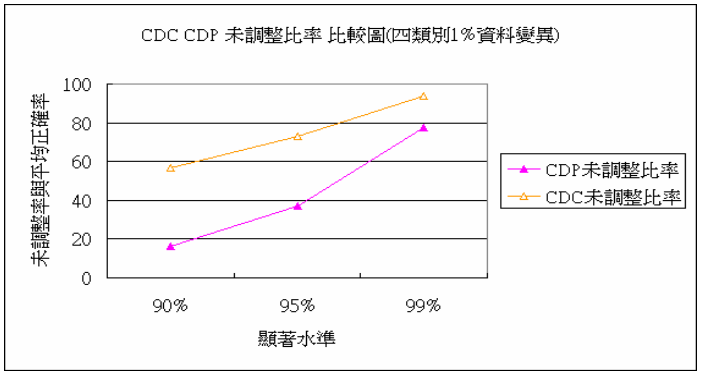


(b) CDP/重建 正確率比較圖(四類別 5%資料變異)

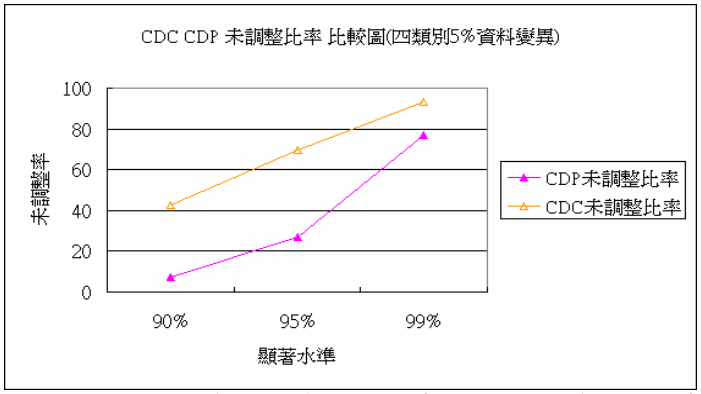
圖 4.8.為 CDP/重建正確率比較圖(分別在 1%與 5%資料變異下)

四、四類別 CDC 與 CDP 之未調整比率比較：

如圖 4.9a 所示我們可以得知在不同的顯著水準下，CDC 的未調整比率均比 CDP 之未調整比率為大，而在圖 4.9b 之 5%資料變異下的 CDC 未調整比率亦比 CDP 還來高，故可推估 CDC 較 CDP 使用的重建成本為低。



(a) CDC/CDP 未調整率比較圖(四類別 1%資料變異)

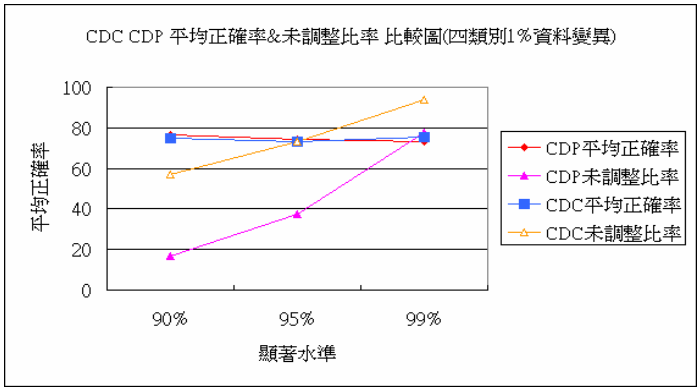


(b) CDC/CDP 未調整率比較圖(四類別 5%資料變異)

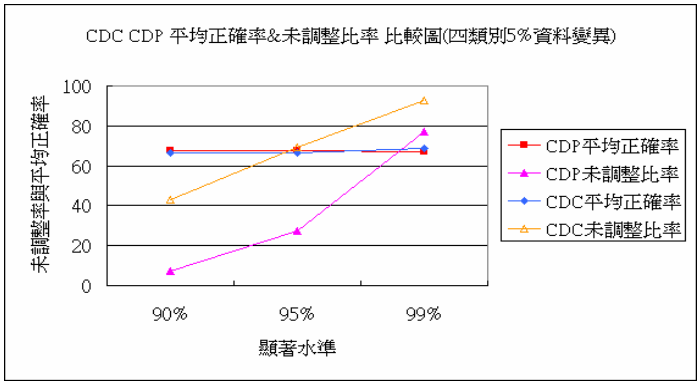
圖 4.9.為 CDC/CDP 未調整率比較圖(分別在 1%與 5%資料變異下)

五、四類別 CDC 與 CDP 正確率與未調整比率之比較：

如圖 4.10a 所示為 CDC 與 CDP 在 1%資料變異下，未調整比率與平均正確率之綜合分析，在圖中我們可以得到在每一顯著水準下 CDC 與 CDP 之平均正確率均相去不遠，且 CDC 之未調整比率遠大於 CDP 。圖 4.10b 亦是相同的情況，由此可證在多類別分類分析之概念漂移偵測上，CDC 在幾近相同甚或更佳之平均正確率而能保持調整成本的最小化。故我們推估 CDC 比 CDP 更適用於多類別的分類分析之概念漂移偵測上。



(a) CDC/CDP 正確率/未調整率比較圖(四類別 1%資料



(b) CDC/CDP 正確率/未調整率比較圖(四類別 5%資料

圖 4.10.為 CDC/CDP 正確率/未調整率比較圖(分別在 1%與 5%資料變異下)

六、綜合以上：

可發現四類別資料與 1%及 5%資料變異下 CDC 的正確率雖未優於 CDP，但其未調整率遠遠高於 CDP，顯著在多類別分類時 CDC 有較佳的概念保留性（即使用較少的修建成本），仍能維持相當之正確率；故在四類別資料下 CDC 較 CDP 具有優勢。

貳、四類別資料案例個數分析

經由前述四類別資料之正確率與未調整率分析，我們可以歸納出在四類別資料中 CDC 與 CDP 這兩種方法之正確率並沒有顯著差異，因此我們再作案例數分析以分析在訓練成本方面是否有顯著差異；如表 4.6 abc 與表 4.7 abc 所示，可以發現以下各點特性：

一、隨顯著水準之提升，未漂移案例增多而多屬性漂移案例變少（重建案例少），顯示我們可以隨訓練成本的提高而使用更高之顯著水準來降低訓練所花費的總成本（同兩類別之歸納結果）。

二、由於區域 5（201-250）為亂數產生之資料，然而兩種演算法之漂移案例並沒有因此產生劇烈的變化，顯示漂移案例分配與資料重疊度（資料亂度）沒有直接的關係（資料亂度影響的是正確率，由前面實驗可得）（同兩類別之歸納結果）。

三、CDC 與 CDP 在五個漂移規則的區域之中無特定案例突增或降的情況發生，顯示兩種方法的偵測並沒有因為不同的資料規則而有不穩定的情況（同兩類別之歸納結果）。

四、在四類別實驗中 Case1、Case3 發生的最少，按直覺判斷為實驗屬性 4 種，而屬性值至多 3 種所引起；屬性愈少，則 Case1(未在決策樹上之屬性漂移亦愈少)；同一屬性之屬性值愈少就愈難出現 Case3(單屬性多屬性值漂移)(同兩類別之歸納結果）。

五、在 3 種顯著水準下之評估之結果顯示，在 1%資料變異下平均數、最大數、最小數中 CDC 之 Case0（未漂移案例）有較高於 CDP 之情形，

且以 5%變異資料來說亦同樣有較高於 CDP 之情況，顯示這兩種方法在四類別的漂移偵測上測出之案例有明顯的差異性（不同於兩類別之結果）。

(a)四類別資料 1%變異之漂移案例數(平均數)

四類別資料1%變異之漂移案例數(平均數)

區域區間			1-50		51-100		101-150		151-200		201-250	
偵測方法			CDC	CDP	CDC	CDP	CDC	CDP	CDC	CDP	CDC	CDP
漂移案例	CASE0	90%	31.8	8.5	28.3	8.9	28.5	11.2	32	10.3	20.3	2.4
		95%	35.3	21.1	39	17.7	39.4	20	38.4	22.3	30.2	12.1
		99%	47.3	39.9	47.6	38.7	46.8	42.8	47.1	41	45.7	31.9
	CASE1	90%	0	0.2	0.3	0.2	0.1	0	0.4	0.6	0	0
		95%	0.1	0.1	0.1	0	0	0	0.2	0	0	0
		99%	0	0	0	0	0	0	0	0	0	0
	CASE2	90%	10.3	10.7	11.7	11.6	13.6	13.8	10.7	11.5	16.2	6
		95%	9.3	12.6	7.7	14.3	7.3	16.2	7.1	13.4	13.2	15
		99%	1.8	7.2	1.4	8	2	5.6	1.4	5.6	3.2	13.2
	CASE3	90%	0.2	1.6	0.2	1.4	0.4	1.2	0.5	2.3	1	2.4
		95%	0.1	0.5	0.1	1	0.2	0.8	0.1	0.8	0.4	1.8
		99%	0	0.2	0	0	0	0.1	0	0.1	0.1	0.4
	CASE4	90%	7.7	29	9.5	27.9	7.4	23.8	6.5	25.3	12.5	39.2
		95%	5.2	15.7	3.1	17	3.1	13	4.2	13.5	6.2	21.1
		99%	0.9	2.7	1	3.3	1.2	1.5	1.5	3.3	1	4.5

(b)四類別資料 1%變異之漂移案例數(最大數)

四類別資料1%變異之漂移案例數(最大數)

區域區間			1-50		51-100		101-150		151-200		201-250	
偵測方法			CDC	CDP	CDC	CDP	CDC	CDP	CDC	CDP	CDC	CDP
漂移案例	CASE0	90%	37	12	33	12	35	14	37	18	25	4
		95%	42	29	44	23	44	25	43	32	37	16
		99%	50	47	49	46	49	47	49	46	49	38
	CASE1	90%	0	1	1	1	1	0	1	2	0	0
		95%	1	1	1	0	0	0	1	0	0	0
		99%	0	0	0	0	0	0	0	0	0	0
	CASE2	90%	15	15	16	16	17	20	15	14	27	12
		95%	14	16	15	24	14	21	10	16	19	17
		99%	5	11	5	16	5	11	5	10	7	21
	CASE3	90%	1	3	1	3	1	3	2	5	2	5
		95%	1	1	1	5	1	2	1	3	2	3
		99%	0	1	0	0	0	1	0	1	1	1
	CASE4	90%	11	36	15	35	14	29	10	33	20	46
		95%	8	20	7	23	8	19	9	18	13	31
		99%	2	6	1	7	2	3	4	8	1	9

(c)四類別資料 1%變異之漂移案例數(最小數)

四類別資料1%變異之漂移案例數(最小數)

區域區間			1-50		51-100		101-150		151-200		201-250	
偵測方法			CDC	CDP	CDC	CDP	CDC	CDP	CDC	CDP	CDC	CDP
漂移案例	CASE0	90%	27	5	21	5	25	8	28	7	14	0
		95%	29	15	30	14	27	12	33	16	26	6
		99%	44	35	44	28	44	37	44	36	42	26
	CASE1	90%	0	0	0	0	0	0	0	0	0	0
		95%	0	0	0	0	0	0	0	0	0	0
		99%	0	0	0	0	0	0	0	0	0	0
	CASE2	90%	5	6	6	8	10	8	8	7	13	2
		95%	4	8	5	7	4	10	6	6	9	11
		99%	0	3	0	2	0	2	0	2	0	8
	CASE3	90%	0	0	0	0	0	0	0	0	0	1
		95%	0	0	0	0	0	0	0	0	0	0
		99%	0	0	0	0	0	0	0	0	0	0
	CASE4	90%	4	23	7	20	2	18	1	19	8	33
		95%	2	10	1	11	1	7	1	9	2	16
		99%	0	0	1	1	1	1	1	1	1	1

表 4.7 四類別資料 1%變異之漂移案例數(平均、最大、最小數)

(a)四類別資料 5%變異之漂移案例數(平均數)

四類別資料5%變異之漂移案例數(平均數)

區域區間			1-50		51-100		101-150		151-200		201-250	
偵測方法			CDC	CDP	CDC	CDP	CDC	CDP	CDC	CDP	CDC	CDP
漂移案例	CASE0	90%	24.2	3.2	22.9	3.6	21.3	2.9	22	4.5	21.5	4
		95%	35.2	14.4	33.6	13.1	38.7	12.9	32	15.8	34.2	11.6
		99%	46.6	37.8	46.4	38.2	47.6	40.1	47.7	40.2	45.2	36.8
	CASE1	90%	0.4	0	0.4	0	0.4	0	0.5	0.1	0.4	0
		95%	0	0.2	0	0	0	0	0.2	0.3	0	0
		99%	0	0	0	0	0	0	0	0.1	0	0
	CASE2	90%	12.1	7.5	14.7	7.2	14.4	8.7	13.4	8.2	13.6	7.9
		95%	10.3	13.8	10.8	14.4	8.2	17	12	14.5	11	14.3
		99%	2.3	9.4	2.44	9.4	1.44	7.1	1.22	7.9	3.44	11.8
	CASE3	90%	0.2	2.3	0.6	3	0.9	2.5	0.3	1.1	1.1	2.9
		95%	0.3	1.5	0.4	1.7	0	1.7	0.1	1.3	0.3	2.5
		99%	0	0.2	0	0.1	0	0.1	0	0	0	0.5
	CASE4	90%	13.1	37	11.4	34.6	13	35.9	13.8	36.1	13.4	35.2
		95%	4.2	20.1	5.2	20.8	3.1	18.4	5.7	18.1	4.5	21.6
		99%	1.1	3.8	1.11	2.4	1	2.2	1.11	3	1.33	3

(b)四類別資料 5%變異之漂移案例數(最大數)

四類別資料5%變異之漂移案例數(最大數)

區域區間			1-50		51-100		101-150		151-200		201-250	
偵測方法			CDC	CDP	CDC	CDP	CDC	CDP	CDC	CDP	CDC	CDP
漂移案例	CASE0	90%	30	7	30	6	30	6	30	9	30	9
		95%	41	19	49	17	44	18	42	24	45	17
		99%	49	47	49	44	49	48	49	46	49	49
	CASE1	90%	2	0	2	0	2	0	2	1	2	0
		95%	0	1	0	0	0	0	2	1	0	0
		99%	0	0	0	0	0	0	0	1	0	0
	CASE2	90%	17	12	22	10	19	13	20	14	19	11
		95%	14	17	18	17	13	23	19	19	18	19
		99%	7	16	7	14	8	16	4	13	7	20
	CASE3	90%	2	4	3	5	3	4	2	3	4	5
		95%	1	3	2	4	0	3	1	3	2	4
		99%	0	1	0	1	0	1	0	0	0	1
	CASE4	90%	17	43	16	47	16	40	20	44	18	39
		95%	7	24	12	24	7	23	13	25	9	31
		99%	2	8	2	7	1	6	2	4	3	7

(c)四類別資料 5%變異之漂移案例數(最小數)

四類別資料5%變異之漂移案例數(最小數)

區域區間			1-50		51-100		101-150		151-200		201-250	
偵測方法			CDC	CDP	CDC	CDP	CDC	CDP	CDC	CDP	CDC	CDP
漂移案例	CASE0	90%	20	1	15	1	14	0	16	1	13	1
		95%	29	9	25	9	34	9	27	9	27	5
		99%	42	29	42	29	41	28	45	33	42	28
	CASE1	90%	0	0	0	0	0	0	0	0	0	0
		95%	0	0	0	0	0	0	0	0	0	0
		99%	0	0	0	0	0	0	0	0	0	0
	CASE2	90%	7	4	7	0	7	5	7	4	7	5
		95%	7	8	0	10	4	11	7	10	4	11
		99%	0	4	0	4	0	1	0	2	0	5
	CASE3	90%	0	1	0	1	0	1	0	0	0	0
		95%	0	0	0	1	0	1	0	0	0	1
		99%	0	0	0	0	0	0	0	0	0	0
	CASE4	90%	9	31	6	20	8	30	10	29	9	28
		95%	1	17	1	15	1	13	1	11	1	15
		99%	0	1	1	1	1	1	1	1	1	1

表 4.8 四類別資料 5%變異之漂移案例數(平均、最大、最小數)

第五節　案例五(Case5)實驗數據分析

本節將會以案例五之漂移處理，調整前、調整後、重建後之正確率作比較，並輔以 1%、5%的資料變異來分析在多屬性漂移裡，CASE5(多屬性單屬性值漂移)與 CASE4（多屬性，多屬性值漂移）發生之比例分析，藉以改善在 CDP-Tree 中案例四（多屬性漂移處理）完全使用重建，而未考慮到更細部區分多屬性漂移案例，所造成的重建次數增加。(實驗數據為 10 次實驗平均值)。

壹、多屬性漂移調整案例之機率分析

表 4.9 為四類別資料，以「CDC 演算法」設定 95%可信水準下偵測之「多屬性漂移」；由表 4.9 可以發現隨著資料變異程度最大，而多屬性漂移為多屬性單屬性值之情形愈多。本論文假定在「多屬性漂移」下可區分為「多屬性值」與「單屬性值」這兩種現象，且由前述提出「多屬性單屬性值」之調整方法，並將之歸類為「案例五」之調整方法。且由此表可發現「案例五」有一定程度的發生機會，若能使多屬性值內之單屬性值漂移獲得調整，以取得不錯的正確率，則可避免將「多屬性漂移」完全視為重建案例之成本浪費。

表 4.9 多屬性漂移內 Case4 與 Case5 頻率比較

多屬性漂移			
		Case4 多屬性值漂移	Case5 單屬性值漂移
資料變異	1%	0.820278	0.179722
	5%	0.64881	0.35119

貳、案例五之正確率分析

由表 4.9 可知，進行多屬性之單屬性值漂移（Case5）的調整是具有一定意義的，因為它在多屬性漂移中佔有一定程度的發生機率。如表 4.10 所示，以 Case5 在 1%、5%資料變異下的分類器調整前、調整後與重建時所進行之測試正確率作分析。

表 4.10 案例五　正確率分佈

	調整前正確率		調整後正確率		重建後正確率	
資料變異	**1%**	**5%**	**1%**	**5%**	**1%**	**5%**
平均值	77.22	74.17	80.38	79.6	90.67	86.93
最大值	87.14	89.6	88.56	89.8	93.92	89.4
最小值	51.76	53.4	68.48	66.2	85.4	83

一、在這兩種資料變異下，我們可以發現在調整後之正確率，有一定程度的提升，且平均正確率約可保持在 80 上下，且調整後之正確率平均值隨資料變異度上升，並沒有顯著下降的情況。

二、且調整後之最小正確率均有明顯的提升，顯示 Case5 可以有效地調整正確率過低的部份，縮小正確率高低相差過巨的情況。

第五章、結論與未來發展

第一節　結論

在本研究的實驗中我們驗證在兩類別的資料中 CDC 與 CDP-Tree 這兩種方法在 5%資料變異下並沒有顯著不同，意即其 CDP-Tree 同樣之分類問題在 CDC 底下仍能作不錯的分類，而多類別的資料裡我們提供了一些實驗來驗證我們的適用性；本論文考量在多屬性漂移處理上缺乏彈性，進而提出一個新的概念漂移案例 Case5，其研究結論列出如下：

壹、偵測的優點：CDC 保留了 CDP-Tree 之優點，即「偵測」的方式來對隨時間變動的資料流作檢定。

貳、適用多類別資料：CDC 改善了 CDP-Tree 之檢定方式，使之能以更低的重建成本（減少案例四之處理）而保持相同的正確率。

參、調整案例修正：CDC 改善了 CDP-Tree 在調整案例區分上過於粗略的問題，利用案例五來對多屬性漂移情況作調整，以節省重建成本，且可維持一定的正確率。

肆、適用各種資料：漂移案例之相對比例不因資料產生方式不同而劇變。

第二節　未來發展

壹、依重要子節點樹作調整：依決策樹特性愈上層節點愈具意義，而相對包含愈多資料筆數的子節點樹亦較具意義。故我們可以為資料區塊之決策樹評估其決策樹路徑組合的重要性，並且依據決策樹路徑組合重要性的漂移狀態來評估哪些部份漂移是更具意義的。且針對漂移檢定依資料筆數加權重要性，藉以針對相關漂移的調整方法作改良。

貳、更有效的概念元檢定方法：由於這個概念漂移偵測問題轉換為一個檢定問題。故我們可以利用替換檢定方法的方式來使漂移的偵測上可以有所改善或適用不同的情況。

參、調整方法的改善：由於目前調整使用到的概念元僅用到屬性值層次資料。若有能在調整時進一步使用到屬性值之類別檢定結果，以作為調整策略上之用時，則可將檢定放在更細部的「屬性值－類別」層下。

參　考　文　獻

一、中文部份

[1]謝千慧, “一個適用於概念漂移資料串流探勘法之研究”, 國立台南師範學院，碩士論文，2004。

二、西文部份

[2] J. R. Quinlan, 1993 “C4.5: Program for Machine Learning,” Morgen Kaufmann Publisher, San Mateo, Ca.

[3] Domingos P. and Hulten G. (2000) Mining High-Speed Data Streams. In Proceedings of the Association for Computing Machinery Sixth International Conference on Knowledge Discovery and Data Mining

[4] Hulten G., Spencer L., and Domingos P. (2001) Mining Time-Changing Data Streams. ACM SIGKDD Conference.

[5] Wang H., Fan W. Yu P. and Han J. (2003) Mining Concept-Drifting Data Streams using Ensemble Classifiers, in the 9th ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD), Washington DC, USA.

[6] Fan W.(2004) Systematic data selection to mine concept-drifting data streams. ACM KDD Conference, pp. 128.137.

[7] Aggarwal C., Han J., WANG j., Yu P. S., (2003) A Framework for Clustering Evolving Data Streams, Proc. 2003 Int. Conf. on Very Large Data Bases (VLDB ’03), Berlin, Germany, Sept. 2003.

[8] Aggarwal C., Han J., Wang J., Yu P. S.,(2004) On Demand Classification of Data Streams, Proc. 2004 Int. Conf. on Knowledge Discovery and Data Mining (KDD’04), Seattle, WA.

[9] Last M. (2002) Online Classification of Nonstationary Data Streams, Intelligent Data Analysis, Vol. 6, No. 2, pp. 129-147.

[10] Law Y., Zaniolo C. (2005) An Adaptive Nearest Neighbor Classification Algorithm for Data Streams, Proceedings of the 9th European Conference on the Principals and Practice of Knowledge Discovery in Databases, springer Verlag, Porto, Portugal.

[11] Ferrer-Troyano F. J., Aguilar-Ruiz J. S. and Riquelme J. C. (2004) Discovering Dceision Rules from Numerical Data Streams, ACM Symposium on Applied Computing, pp. 649-653.

[12] Gaber, M, M., Krishnaswamy, S., and Zaslavsky, A., (2005). On-board Mining of Data Streams in Sensor Networks, Accepted as a chapter in the forthcoming book Advance Methods of Knowledge Disvcovery from complex Data,(Eds.) Sanghamitra Badhyopadhyay, Ujjwal Maulik, Lawrence Holder and Diane cook, Springer Verlag, to appear.

[13] G. Hulten, L. Spencer, and P. Ddmingos, "Mining Time-Changing Data Streams, " In Proc. 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA., pp. 97-106, Aug. 2001.

[14] R. Klinkenberg and I. Renz, " Adaptive Information Filtering：Learning in The Presence of Concept Drifts, "In M. Sahami, M. Craven, T. Joachims, and A. McCallum, editors, Workshop Notes of the ICML-98 Workshop on Learning for Text Categorization, pp.33-40, Menlo Park, CA., AAAI Press, 1998.

[15] J. R. Quinlan, "Learning Efficient Classification Procedures and Their Application to Chess End Games, " Machine Learning : An Artificial Intelligence Approach , Michalski et. All (EDS), Tioga Publishing, Palo Alto, 1983.

[16] J. R. Quinlan, "Induction of Decision Trees, " Machine Learning, Vol. 1, No. 1, pp. 81-106, 1986.

[17] J. R. Quinlan, "C4.5:Program for Machine Learning, " Morgen Kaufmann Publisher, San Mateo, CA, 1993.

[18] P. Domingos and G. Hulten, "mining High-Speed Data Streams, "In Proc. Association for Computing Machinery 6th International Conference on Knowledge Discovery and Data Mining, Bostion, MA., pp. 71-80, Aug. 2000.

[19] Quinlan, J.R., 1986. Induction of Decision Trees. Machine Learning, 1, 1, pp.81-106

[20] Lewis, R.J., M.D., Ph.D., 2000. An Introduction to Classification and Regression Tree (CART) Analysis. The Annual Meeting of the Society for Academic Emergency Medicine, Francisco, California.

[21] Hand D.J., Mannila H., and Smyth P. (2001) Principles of data mining, MIT Press.

[22]Hastie T., Tibshirani R., Friedman J. (2001) The elements of statistical learning: data mining, inference, and prediction, New York: Springer.

[23]M. Maloof, "Incremental Rule Learning with Partial Instance Memory for Changing Concepts, "In Proc.s of the international Joint Conference on Neural Networks, Los alamitos, CA: IEEE Press, Jul. 2003.

[24]G. Widmer and M. Kubat, "Learning in The Presence of Concept Drift and Hidden Contexts, " Machine Learning, Vol. 23, No. 1, pp. 69-101, 1996.