

植基於模糊切割與資料方塊之多方位關聯法則挖掘法

A Multi-aspect Association Rule Miner Based on Fuzzy Partition and Data Cubes

邱宏彬助理教授¹

蘇建源研究生²

康書萍研究生³

hpchiu@mail.nhu.edu.tw jienyien@mail2000.com.tw spkang@sun.phnvs.cy.edu.tw

南華大學資訊管理系碩士班⁴

摘要

在現今競爭激烈與變化快速的環境下，對於資訊的脈動往往不容易有完整的掌握，再加上資料庫的資料愈來愈多且複雜的同時，資料庫的管理與應用也逐漸被企業所重視。利用資料挖掘技術可以幫助企業從龐大且複雜的資料庫中找到隱藏的資訊，特別是在關聯法則技術方面的探討。如何才能有效地推導出關聯法則，在過去已經有許多的方法相繼被提出，但大部份的演算法都有其限制，對於(a)商品項目的多樣化、(b)決策者所需的大方向性的資訊、(c)商品數量與價格等數值型資料、(d)線上挖掘、與(e)資料庫經常性的更新等問題無法有效地同時加以處理。因此，本研究將利用商品項目的階層分類樹(taxonomy)、模糊切割的觀念與多維度資料模式的資料方塊提出一漸進式多階層關聯法則挖掘法，以滿足使用者多方位的挖掘需求。關鍵詞：個人化介面、忠誠度、差異化行銷、RFM model

關鍵詞：階層分類樹、多階層關聯法則、模糊切割、資料方塊、線上挖掘、漸進式挖掘

¹ Hung Pin Chiu, Assistant Professor

² Chien Yuan Su, Postgraduate

³ Su Ping Kang, Postgraduate

⁴ Department of Information Management, NAN HUA University

Abstract

Data mining is the exploration and analysis of large quantities of data in order to discover meaningful patterns and rules. It is an important discipline, which has widely applied in fields ranging from customer relationship management to marketing, and medicine. Mining association rules from transaction databases is most commonly seen in data mining. However, most conventional algorithms can not simultaneously and effectively satisfy the following requirements: (a) the strong associations between the great diversity of products, (b) the desired rules at cross level of concept, (c) the relationships among transactions using numeric values, (d) on-line mining, and (e) incremental mining. In this paper, we integrate the taxonomy, fuzzy partition method, and data cubes technique to propose a sophisticated multi-level association rules miner to achieve the multi-aspect mining requirements for the users.

Keyword: Taxonomy, Multiple-level association rules, Fuzzy partition, Data cube, Online mining, Incremental mining

壹、導論

在現今競爭激烈、資訊科技快速成長、網際網路的興起與顧客至上的環境下，瞭解與滿足顧客的需求是企業獲利的關鍵因素，這樣的轉變使得企業必須要非常瞭解顧客的特性以及其消費行為模式等相關資訊，才能進一步推出滿足顧客且能夠替企業創造價值的行銷策略。在企業的交易資料庫中隱藏了許多顧客與產品的一些資訊，但隨著資料庫愈趨龐大且複雜時，如何有效地從中找到有用的資訊與知識將是非常重要的關鍵。

資料挖掘是一個可從龐大且複雜的資料庫中找到隱藏資訊的技術，其中又以關聯法則是目前所有資料挖掘技術研究中最

常探討的技術之一[1-8,11-13,16]。透過關聯法則，我們可以從資料庫中找出某些商品項目間彼此的關聯性來幫助使用者發覺隱含的知識。如何有效率地從大量的資料中搜尋出隱含的資訊與有用的規則，一直是關聯法則探勘研究領域中十分重視的課題[1,4-8,11]。在關聯法則的挖掘上，最具代表性的方法是由 Agrawal et al.所提出的 Apriori 演算法[13]。Apriori 演算法中主要包含了以下兩個主要的步驟：

- (1)反覆的產生候選項目組 (candidate itemsets) 並搜尋整個資料庫，直到找出所有的大項目組(large itemsets)。
- (2)利用(1)所找出的大項目組，推導出所

有的關聯法則。

在步驟(1)中，候選項目組的支持度必須大於或是等於使用者所設定的最小支持度門檻值，才能成為大項目組。同樣的，在步驟(2)中所產生的關聯法則期可靠度也必須達到使用者最初設定的最小可靠度門檻值。因為 Apriori 演算法牽涉到多次資料庫掃描以及可能產生過多的候選項目集，因此是一件耗時的工作。有許多改善的方法被提出，如 DHP 演算法[11]利用減少候選 2-itemsets 的個數來減少計算量以增進效能，而 DIC [16]、Sampling [8]等演算法則是減少資料庫的掃描次數的方式來改善效能。這些演算法皆是在處理單一商品層次間的關聯法則以及資料型態是屬於二值屬性的資料。傳統關聯法則挖掘法的整理可參見[6]。

最小支持度門檻值的設定並不是一件容易的事。藉由線上調整門檻值的功能，使用者可機動的挖掘出對應的關聯法則以作為適當決策的依據，此即所謂的線上探勘(On-line mining)。Apriori-like 演算法並不適合線上探勘，因為每次調整新的門檻值時，這些演算法都必須重新掃描原始資料庫 [4]。

現實生活中，企業的交易是隨時進行的，而資料庫中的資料也必須隨著交易的新增而動態的記錄新的資料，因此產生了在動態資料庫進行漸進式探勘(incremental mining)的需求 [1,4,5,7]。Apriori-like 演算法也不適合漸進式探勘，因為每次新的交

易資料進入資料庫時，這些演算法必須重新掃描更動後的整個資料庫而不能只考慮新增記錄集。Cheung et al.所提出的 FUP(Fast Update)的演算法[7]可算是漸進式探勘方法的代表範例，當資料庫更動時，藉由比較原始資料庫的高頻項目組(large itemsets)與新增記錄集中的高頻項目組來減少其掃描原始資料庫的次數。但 FUP 不支援線上探勘。許志豪[4]利用縮減項目集絡(Reduced Itemsets Lattice)的觀念，在動態資料庫中做線上關聯法則探勘。該演算法利用 Apriori principle 將必要的項目集資訊儲存在絡(Lattice)的架構中，以方便關聯法則的探勘。在資料庫更新或最小支持度門檻值調整時，必須同步將絡中所儲存的資訊做必要的調整，若儲存的資訊越詳細，探勘時就越能減少必須到資料庫做確認的情況。

隨著商品項目的多樣化，找出的關聯法則數目可能會變的較少，更多隱藏的知識便不能被挖掘出來。再者，上層的決策者往往沒時間處理瑣碎的訊息，他們所需要的是一個大方向性的資訊來進行決策，所以，提供一個能挖掘多層次關聯法則的演算法便能滿足各方的需求。另外，交易資料庫中也包含商品數量與價格等數值型資料，這些資料也為商品項目間的關聯性挖掘提供重要的資訊，所以能夠處理數值型資料的方法也就非常重要。但是，傳統關聯法則挖掘法幾乎都在處理單一商品層次間的關聯法則以及資料型態是屬於二值

屬性的資料，因此，過去的方法將很難有效同時處理這樣複雜的狀況。

綜合上述，傳統關聯法則挖掘法對於(a)商品項目的多樣化、(b)決策者所需的大方向性的資訊、(c)商品數量與價格等數值型資料、(d)線上挖掘、與(e)資料庫經常性的變動等問題無法同時有效地處理，這是一個十分複雜、困難但卻有實際需要的問題。因此，本研究將提出一多功能的關聯法則的挖掘法來彌補過去演算法的不足。該方法將利用多階層關聯法則來解決商品項目的多樣化、決策者需要大方向性的資訊的問題，再利用模糊切割(Fuzzy Partition)的觀念來解決資料庫資料型態多樣性的問題，最後再利用資料方塊(Data Cubes)來達成漸進式挖掘以解決資料庫經常性變動所造成的問題。

在本文第貳節將說明研究方法與相關文獻探討，包括多階層關聯法則、模糊切割技術與資料方塊；接下來，第參節將描述本研究所提出的可滿足多方位需求之多階層關聯法則挖掘法的流程與作法；在第四節我們用一範例說明之；最後是結論與未來展望。

貳、研究方法與相關文獻探討

本研究將整合商品項目的階層分類樹(taxonomy)、模糊切割的觀念與資料方塊提出一漸進式多階層關聯法則挖掘法，以滿足使用者多方位的挖掘需求。底下我們探

討相關研究的限制，並闡述本研究方法的理由與目的。

(一)商品項目的階層分類樹(taxonomy)與多階層關聯法則

傳統關聯法則挖掘法幾乎都在分析單一商品層次間的關聯，這些方法並沒有考慮商品項目的階層關係。隨著商品項目的多樣化，有時候會很難去找出具有強烈關聯性(strong association)的關聯法則，亦即可能導致很多商品項目組無法滿足我們所定的最小支持度門檻值，使得找出來的關聯法則數目可能會變少，甚至產生許多隱藏的知識便無法被挖掘出來的問題。真實世界的商品通常以階層結構的方式來組織[10,15,19]，若將資料分析的層次提升，可以容易找出較多且更多樣性的關聯法則以滿足不同使用者的需求，此即多層次的關聯法則挖掘(multiple-level association rules mining)。這些較高層次的關聯法則可以提供大方向的關聯性，讓決策者了解大環境的趨勢，以下達決策。

為了達成多層次關聯法則的搜尋，我們必須先對商品進行歸納的動作。我們可以將商品分出數種不同的類別並且建立商品項目的階層分類樹。如圖 1 所示，根節點(root node)代表著所有產品的總稱，而葉節點(left node)則代表產品完整的名稱。現階段在多層次資料挖掘的研究上，多傾向於利用現有單層次的演算法去實作。其中最著名的是 J. Han et al.所提出的 ML_T2L1 演算法[10]與 R. Srikant et al.所提出的

Cumulate 演算法[15]。

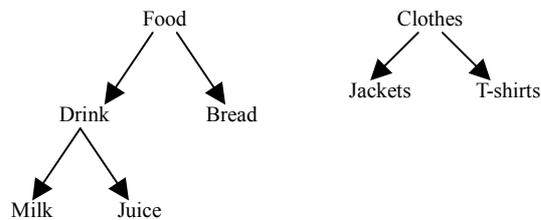


圖 1. 商品之多階層分類樹

J. Han et al. 根據 Apriori principle 提出一個由上而下逐層深入的方法 (top-down progressive deepening method) 來挖掘多層關聯法則。每一層所訂定的最小支持度門檻值可以不相同，典型的狀況是由最上層逐層遞減門檻值。首先，交易資料庫的每一產品項目都根據其在階層分類樹的層次資訊加以編碼。例如，令根節點為第一層，Food 的編碼是 1，則 Drink 的編碼是 11，Bread 的編碼是 12，Milk 的編碼是 111，Juice 的編碼是 112。接著，利用 Apriori-like algorithm [10]，由階層 1 開始找出其高頻 k -項目組 (k -itemsets)。然後，只考慮其高頻 1-項目組在第二層的子代產品，同樣地，利用 Apriori-like algorithm 找出該層的高頻 k -項目組。以此類推，逐層挖掘出各層的高頻項目組。此方法必須進一步修改才能進行跨階層 (cross-level) 的關聯法則挖掘。R. Srikant et al. 提出一跨層之關聯法則挖

掘法，該方法並不將項目編碼，而是將每一交易紀錄中的各個項目的祖先加入該交易紀錄中，如此即可應用傳統的 Apriori algorithm 在這延伸的交易紀錄中挖掘出可跨階層的關聯法則。在這基本方法的基礎上，Cumulate 演算法加入三個最佳化來改善效率：(1) 只將出現在候選項目組中的祖先加入交易紀錄中，而非全部項目的祖先都加入、(2) 事先找出所有項目的祖先，便不需再循著 Taxonomy 架構來找出其祖先、(3) 去除同時包含商品項目及其祖先的商品項目組。

ML_T2L1 演算法與 Cumulate 演算法所挖掘的多(跨)階層關聯法則可以提供更多樣性與大方向的關聯性以滿足不同使用者的決策需求。然而這兩種方法有許多缺失與限制：(a) 並未考慮到各個項目(組)在每一筆交易所購買的數量或是購買的金額等數值型資料的分析處理，(b) 因為是屬於

Apriori-like 演算法，所以無法滿足漸進式探勘與線上探勘的需求。

(二) 模糊切割與數量型關聯法則問題 (Quantitative Association Rules Problem)

在傳統集合上，一個元素 (Object) 不是屬於就是不屬於某個集合，其歸屬關係 (Membership) 是二元性的。例如，一個歸類為”高”的人不能同時是”矮”的人，此即 Crisp set。用來表示一個元素 x 是否屬於集合 S 的特性函數 (Characteristic function) 定義如下：

$$m_s(x) = \begin{cases} 1 & \text{if } x \text{ is an element of } S \\ 0 & \text{otherwise} \end{cases} \quad \dots\dots\dots (公式1)$$

雖然傳統集合是有用的，但其有許多限制。例如當我們定義身高超過 175 公分的人是”高”，則一個 174 公分的人不能被歸類為”高”。在真實情況中，人們時常採用比較彈性的作法，例如，一個 165 公分的人可同時歸類為不同的程度的”高”和”矮”。

容許部份歸屬關係的集合即為模糊集合，這是由 Zadeh 在 1965 年所提出的[20]。一個模糊集合的歸屬函數將集合的所有元素對應至[0,1]區間，此對應值稱為歸屬度 (Membership grade)。歸屬度反應一個元素屬於集合的程度。在宇集合 U 上的模糊集合 A 可由歸屬函數 $\mu_A(x)$ 來定義：

$$\mu_A(x): U \rightarrow [0,1] \quad (公式2)$$

其中

$$\begin{aligned} \mu_A(x) &= 1 && \text{if } x \text{ 完全屬於 } A \\ \mu_A(x) &= 0 && \text{if } x \text{ 完全不屬於 } A \\ 0 < \mu_A(x) < 1 && \text{if } x \text{ 部份屬於 } A \end{aligned}$$

根據文獻我們可以發現歸屬度函數有相當多種，最常被應用的有三角形、梯形、L-R、指數函數、S 函數等[20]。如何找出一個適當的歸屬度函數是目前模糊理論研究中相當實際且重要的問題，一般來說並沒有通用的定理或公式，通常是依據經驗或統計方法來加以確定，很難具客觀性。許多研究學者希望能夠透過系統化的方式來找出比較客觀的歸屬度函數，最常見的做法是先建立粗略的歸屬度函數，然後藉由學習與不斷地實驗，逐步進行修正和調整使得歸屬度函數更加客觀。最簡單的歸屬度函數 - 三角形歸屬度函數，其定義如下：

$$\mu_A(a,b,c,x) = \begin{cases} 0, x < a \\ (x-a)/(b-a), a \leq x \leq b \\ (c-x)/(c-b), b \leq x \leq c \\ 0, x > c \end{cases} \quad (公式3)$$

其圖形如下圖 2 所示：

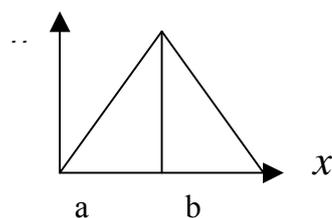


圖 2. 三角形歸屬度函數圖

相較於傳統集合，模糊集合表示法並沒有使用單一的門檻值來定義集合的歸屬情況而是在歸類上允許某個範圍的灰色區域。例如，圖 3 表示身高”高”和”矮”的歸屬函數，我們可以看到 170 公分的人可分別以不同的程度歸類為”高”或”矮”的人。

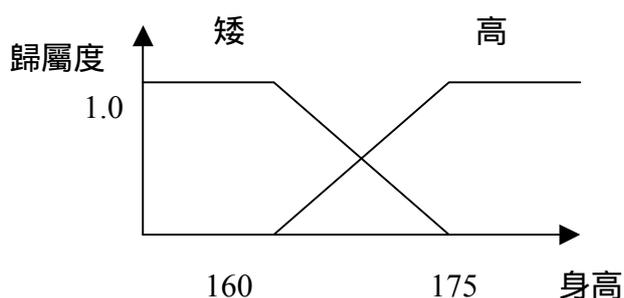


圖 3. 身高歸屬函數

在許多交易資料中，商品數量與價格是隱含重要關聯的數值型資料。為了分析商品數量(或價格)之間的關聯性，我們必須先將數值資料轉換成類別型資料。以區間為基礎的分割(interval-base partition)是最常用的歸類方式，落在某一個特定區間內的數值即歸屬為該區間所代表的類別[14]。如此，可將數量型關聯法則問題轉換成傳統的布林型關聯法則問題[14]。但是在區間邊界的資料在歸類的過程會造成資訊遺失而降低挖掘出的關聯法則的有效性。模糊集合在歸類上允許某個範圍的灰色區域，因此，我們將利用模糊切割在數值屬性上定義多個模糊項(集合)來進行數值資料的歸類，例如，在 X、Y、Z 上都分割成 Low(L), Medium(M),和 High(H)三部分，分

別表示為 X.L、X.M、X.H、Y.L、Y.M、Y.H、Z.L、Z.M、Z.H 等項目，而將問題轉換成模糊關聯法則問題[21]。其挖掘出來的模糊關聯法則是以語文詞(linguistic term)來表示，將更自然且容易被人們所瞭解。

模糊切割(Fuzzy partition)是在模糊 if-then 法則中相當重要的概念。不同的分割方式可得到不同組合的 Fuzzy if-then rules，其分割愈多則產生的法則也愈多，可區分較複雜的資料分類，但其推論的效率相對地較差。一般而言，依據資料的複雜度及分佈情況有三種分割方式可供選擇即方格分割、樹狀分割與散佈分割，在實務上最常使用的模糊分割法是方格分割[20]。在方格分割部份以 Ishibuchi 在 1992 年[9]提出的 Simple fuzzy partition 的概念最為著名。本研究主要是利用 Simple fuzzy partition 的概念來建立多階層模糊關聯法則。

(三) 以資料方塊為基礎之線上與漸進式模糊關聯分析

隨著資料庫技術不斷地發展及資料庫管理系統的廣泛應用，資料庫的資料量和規模也急劇成長，而資料倉儲(data warehouse)與線上資料分析(On-Line Analytical Processing, OLAP)也因此快速地發展中。資料倉儲與線上資料分析主要應用在多維度的資料模式中，而這個多維度的資料模式在搜尋、檢視、表示部份，資料方塊扮演著重要的角色。透過資料方塊，我們可以在多維度的龐大資料中快速

地找到我們要的資料。三維的資料方塊如下圖 4：

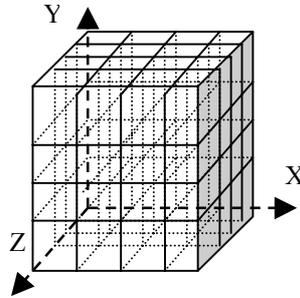


圖 4. 三維資料方塊表示圖

若以 Apriori-like 的演算法來進行模糊關聯法則挖掘時，同樣會有線上探勘及漸進式探勘的問題。我們將商品的數值屬性視為 Data cube 的維度，屬性上分割的每個模糊項對應到該維上的不同座標，如此，我們將可使用多維資料模式與 OLAP 的概念來處理這兩個問題。舉例來說，假設將商品 X、Y、Z 的數量分割成 K_x, K_y, K_z 個模糊項(集合)，則資料方塊的大小為 $(K_x + 1)(K_y + 1)(K_z + 1)$ 。換句話說，其 Subcube 的座標可由 $(0, 0, 0)$ 到 (K_x, K_y, K_z) 。我們掃描資料庫以累計不同項目組的模糊支持度，並儲存在對應的 subcube 中。令 $1 \leq x \leq K_x, 1 \leq y \leq K_y, 1 \leq z \leq K_z$ 則座標 $(x, 0, 0), (0, y, 0), (0, 0, z)$ 儲存的是 1-itemsets 的模糊支持度值 (fuzzy support); 座標 $(x, y, 0), (z, 0, z), (0, y, z)$ 儲存的是 2-itemsets 的模糊支持度值; 而座標 (x, y, z) 儲存的是 3-itemsets 的模糊支持

度值。因為資料方塊預先儲存了所有項目組的支持度，如此不必掃描資料庫即可快速求出關聯法則，達成線上挖掘 (On-line mining) 的功能[5]，而當新增紀錄時，亦可即時加以處理並更新資料方塊 (Data cube) 中對應項目組的支持度，以達到漸進式挖掘(Incremental mining)的目的[5]。

綜合言之，我們利用階層分類樹及多階層關聯法則來滿足商品項目的多樣化、決策者需要大方向性的資訊等需求。利用模糊切割的觀念來處理商品數量等數值型資料以分析其關聯性，最後，為了提升效能，我們利用資料方塊來達成線上探勘及漸進式探勘的目標。

參、多方位關聯法則挖掘法

本研究整合商品項目的階層分類樹、模糊切割技術與資料方塊以建構一個滿足多方位需求的多階層關聯法則挖掘法，其

系統流程如圖 5 所示。本節詳述其作法。

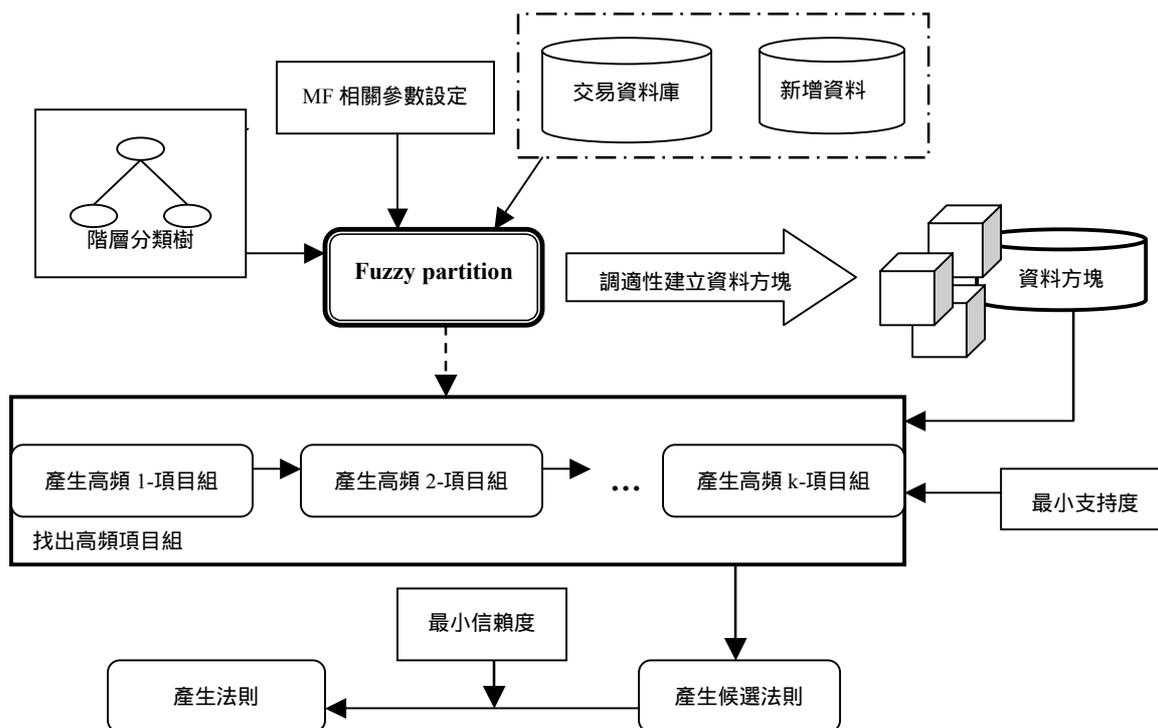


圖 5. 模糊切割之漸進式多階層關聯法則流程圖

(一) 模糊切割

我們利用模糊切割在數值屬性上定義多個模糊項(集合)來進行數值資料的歸類，以便分析商品數量等數值型資料的關聯性。簡單的三角形歸屬函數可使系統具有較佳的執行效率，且已成功地應用在很多問題上[20]，因此，在不失一般性的前提下，本研究先利用三角形歸屬度函數進行探討。假如在維度 D 上建立 K 個模糊集

合，如圖 6 所示，則第 i 個集合的模糊歸屬函數定義如下：

$$\mu_i^D(x) = \begin{cases} 0, & x < a_i \\ (x - a_i)/(b_i - a_i), & a_i \leq x \leq b_i \\ (c_i - x)/(c_i - b_i), & b_i \leq x \leq c_i \\ 0, & x > c_i \end{cases} \dots\dots\dots (\text{公式4})$$

其中 $1 \leq i \leq K$

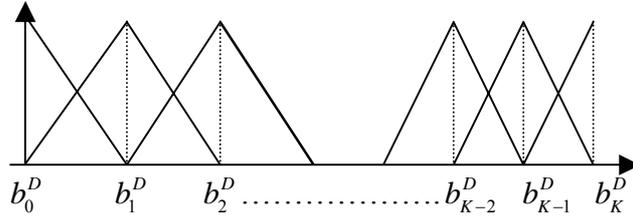


圖 6. 以三角形歸屬度函數建立的 Simple fuzzy partition

一般而言，均分且對稱的三角形模糊隸屬函數即可有不錯的效果[20]。因此我們可以求得 a , b , c 的間距為：

$$S = \frac{(x_{\max}^D - x_{\min}^D)}{(K-1)} \quad (\text{公式 5})$$

其中 x_{\max}^D 與 x_{\min}^D 分別為維度 D 資料的最大值與最小值，則第 i 個模糊歸屬函數的參數分別為

$$\begin{aligned} b_i &= x_{\min}^D + S \cdot (i - 1) \\ a_i &= b_i - S \\ c_i &= b_i + S \end{aligned}$$

我們將各維度所分割之模糊集合的相關參數記錄起來做為後續處理使用。接著，我們可由一維模糊分割進一步建立多維的模糊分割[20-21]。假設各維分割的模糊集合 (Fuzzy sets) 數目分別為 K_1, K_2, \dots, K_n 且 $x = (x_1, x_2, \dots, x_n)$ 表示一筆 n 維的資料，則第 i 維資料值在其第 t_i 個模糊集合的歸屬度為 $\mu_{t_i}(x_i)$ 其中 $1 \leq t_i \leq K_i$ ，如此，我們可以計算 x 在模糊集合 t_1, t_2, \dots, t_n 所組合之關係的歸屬度為

$$\mu_{t_1 \times t_2 \times \dots \times t_n}^{D_1 \times D_2 \times \dots \times D_n}(x_1, x_2, \dots, x_i, \dots, x_n) = \mu_{t_1}^{D_1}(x_1) \otimes \mu_{t_2}^{D_2}(x_2) \otimes \dots \otimes \mu_{t_i}^{D_i}(x_i) \otimes \dots \otimes \mu_{t_n}^{D_n}(x_n) \quad (\text{公式 6})$$

⊗ 為 Fuzzy conjunction operation，若取 Product operation 作為 Fuzzy conjunction operation 則

$$\mu_{t_1 \times t_2 \times \dots \times t_n}^{D_1 \times D_2 \times \dots \times D_n}(x_1, x_2, \dots, x_i, \dots, x_n) = \mu_{t_1}^{D_1}(x_1) \times \mu_{t_2}^{D_2}(x_2) \times \dots \times \mu_{t_i}^{D_i}(x_i) \times \dots \times \mu_{t_n}^{D_n}(x_n) \quad (\text{公式 7})$$

例一：資料表中有二維資料分別為商品 1 與商品 2 之商品數量，共 8 筆資料如下表 1：

表 1. 範例資料表

	商品 1 數量	商品 2 數量
r_1	10	1
r_2	20	15
r_3	15	5
r_4	0	0
r_5	20	10
r_6	10	2
r_7	60	40
r_8	35	50

我們在商品 1 數量與商品 2 數量上各定義三個集合與五個集合，分別為少 (A_1^1) 中 (A_2^1) 多 (A_3^1)，以及最少 (A_1^2) 次少

(A_2^2) 中 (A_3^2) 次多 (A_4^2) 最多 (A_5^2)，

則 $S^1 = \frac{(60-0)}{(3-1)} = 30$ 和

$S^2 = \frac{(50-0)}{(5-1)} = 12.5$ ，商品 1 數量的三角

模糊隸屬函數的參數分別為：

$(a_1, b_1, c_1) = (-30, 0, 30)$, $(a_2, b_2, c_2) = (0, 30, 60)$,

及 $(a_3, b_3, c_3) = (30, 60, 90)$ ，而商品 2 數量的

三角模糊隸屬函數的參數分別為：

$(a_1, b_1, c_1) = (-12.5, 0, 12.5)$,

$(a_2, b_2, c_2) = (0, 12.5, 25)$,

$(a_3, b_3, c_3) = (12.5, 25, 37.5)$,

$(a_4, b_4, c_4) = (25, 37.5, 50)$ ，及

$(a_5, b_5, c_5) = (37.5, 50, 62.5)$ 。以第一筆為

例，商品 1 數量的值為 10，其歸屬度計算為： $\mu_1^1(10) = 0.667$, $\mu_2^1(10) = 0.333$, $\mu_3^1(10) = 0$ ，而商品 2 數量的歸屬度計算為

$\mu_1^2(1) = 0.92$, $\mu_2^2(1) = 0.08$, $\mu_3^2(1) = 0$,

$\mu_4^2(1) = 0$ 及 $\mu_5^2(1) = 0$ 。因此，可以算出第

一筆紀錄在二維模糊分割的每一個子方塊 (subcube) 的歸屬度分別為：

$$\mu_{1 \times 1}^{1 \times 2}(10, 1) = \mu_1^1(10) \times \mu_1^2(1) = 0.667 \times 0.92 = 0.614,$$

$$\mu_{1 \times 2}^{1 \times 2}(10, 1) = \mu_1^1(10) \times \mu_2^2(1) = 0.053,$$

$$\mu_{2 \times 1}^{1 \times 2}(10, 1) = \mu_2^1(10) \times \mu_1^2(1) = 0.306,$$

$$\mu_{2 \times 2}^{1 \times 2}(10, 1) = \mu_2^1(10) \times \mu_2^2(1) = 0.027,$$

其他的歸屬度皆為 0，依此可類推算出其他 7 筆的歸屬度。

□

(二) 資料方塊的建立

如第貳節所述，當我們進行模糊分割而建立各商品項目數量維度的模糊集合之後，可以將各模糊集合視為一項目 (Item)，如此，可以利用 Apriori-like 演算法擷取項

目之間的模糊關聯法則[2]。但當資料庫中的資料量很龐大時，Apriori 演算法是相當費時的。我們可以事先計算這些項目組(itemset)的模糊支持度(Fuzzy support)值並儲存在資料方塊之中，如此，不必掃描資料庫即可快速求出關聯法則，達成線上挖掘的功能，而當新增紀錄時，亦可即時加以處理並更新資料方塊中對應項目組的支持度，以達到漸進式挖掘的目的。假設資料庫中共有 n 筆交易紀錄，商品分類樹中共有 m 個節點，亦即共有 m 個商品(含商品祖先)。資料方塊建構方式如下所示：

$$FS(A_1^{D_1} \times A_2^{D_2} \times \dots \times A_k^{D_k}) = \sum_{i=1}^n \mu_{x_1 \times x_2 \times \dots \times x_k}^{D_1 \times D_2 \times \dots \times D_k}(x_1, x_2, \dots, x_k) = \sum_{i=1}^n \mu_{x_1}^{D_1}(x_1) \times \mu_{x_2}^{D_2}(x_2) \times \dots \times \mu_{x_k}^{D_k}(x_k) \quad (\text{公式 8})$$

其中 x_i 為第 i 筆資料之維度 D_k 的值，

$1 \leq k \leq m$ 。值得注意的是，此處我們是以模糊計數(fuzzy count)當作模糊支持度，實際上，除以總筆數才是真正的模糊支持度。

(三) 多方位關聯法則挖掘法的計算步驟

因為所有 itemsets 的模糊支持度值已經存在 Data cube 中，所以，我們可以快速掃描 Data cube 並與最小模糊支持度門檻值比較來產生高頻項目組 (K -itemsets)，再由高頻項目組推導出關聯法則，並與最小模糊信賴度門檻值比較來產生符合使用者的法則。假設在商品與商品祖先的數量上之模糊切割的模糊集合個數分別為 K_1, K_2, \dots, K_n ；使用者制訂的門檻值分別為：最小模糊支持度 $minfs$ 與最小信賴度 $minfc$ ，則漸進式模糊多階層關聯法則，其

步驟一、從資料庫擷取出各商品的數量資料，並計算對應之商品祖先的數量

步驟二、使用者設定參數 K_1, K_2, \dots, K_m 以便系統由資料庫求出所有歸屬函數的三個參數

步驟三、利用公式 (7) 計算每一筆紀錄在 K -itemsets 的歸屬度值，並累加至資料方塊所對應之 Subcube 的模糊支持度中。 K -itemsets 的模糊支持度定義如下：

流程步驟如下：

Step1. 透過商品分類樹產生交易資料轉換表

Step2. 由交易資料轉換表建立資料方塊

Step3. 找出各維度之高頻項目組

(a) 掃描資料方塊中，對應 1-itemsets 的 Subcubes，若其儲存的模糊支持度大於 $minfs$ ，則加入高頻 1-itemsets 的集合 L_1

(b) 產生高頻 2-itemsets，有兩種方式可用：

A. 合併 L_1 產生 C_2 ，再到 Data cube 中檢查 C_2 中每一個候選 2-itemsets 的模糊支持度，若大於 $minfs$ ，則加入集合 L_2 中。

B. 直接掃描 Data cube 中，對

應 2-itemsets 的 Subcubes 之模糊支持度，若大於 $minfs$ ，則加入 L_2 中。

(c) 同理，可產生 $L_3, L_4, \dots, L_k, k \leq n$ 。

Step4. 產生法則

由各維度之高頻項目組產生候選法則，計算每一個候選法則之模糊信賴度（公式9），並與最小模糊信賴度 $minfc$ 比對，大於或是等於 $minfc$ 的法則為使用者所要之法則。令法則 R 的通式如下：

$$\text{Rule R: } A_{i_1}^{D_1} \times A_{i_2}^{D_2} \times \dots \times A_{i_i}^{D_i} \Rightarrow A_{i_{i+1}}^{D_{i+1}} \times \dots \times A_{i_k}^{D_k}$$

，其中 $1 \leq i \leq k \leq n$ ，則

$$FC(R) = \frac{FS(A_{i_1}^{D_1} \times A_{i_2}^{D_2} \times \dots \times A_{i_k}^{D_k})}{FS(A_{i_1}^{D_1} \times A_{i_2}^{D_2} \times \dots \times A_{i_i}^{D_i})} \quad (\text{公式9})$$

若 $FC(R) \geq minfc$ ，則 R 為所要之有效的關聯法則。

再者，透過資料方塊的預處理，我們可以達到線上挖掘與漸進式挖掘的功能：

(a) 線上挖掘

當 $minfs$ 改變時，重新執行 Step3 和 step4。

(b) 漸進式挖掘

若有資料新增時，可以及時或批次的方式，計算新增紀錄對應之商品祖先的數量，接著，利用公式（7）計算每一筆

紀錄在 K -itemsets 的歸屬度值，並累加至資料方塊所對應之 Subcube 的模糊支持度中。

肆、多方位關聯法則挖掘法的範例說明

假設交易資料如下表 2 所示；各維度分別分割成 L、M、H 三個模糊項，其模糊歸屬函數如圖 7 所示；商品分類樹如圖 8 所示；最小模糊支持度是 1.6、最小信賴度是 0.7。

表 2. 交易資料表

交易 ID	項目
1	(milk,3)(bread,4)(cookie,2)
2	(juice,3)(bread,7)(cookie,7)
3	(milk,2)(juice,10)(cookie,5)
4	(milk,7)(juice,2)(bread,8)(cookie,8)
5	(juice,2)(bread,8)(cookie,10)

圖 7. 模糊歸屬函數圖

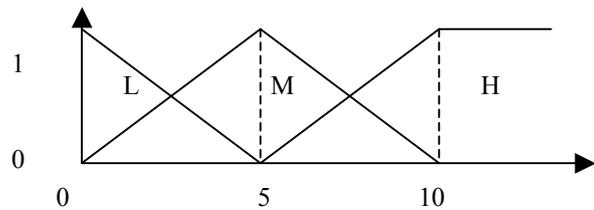
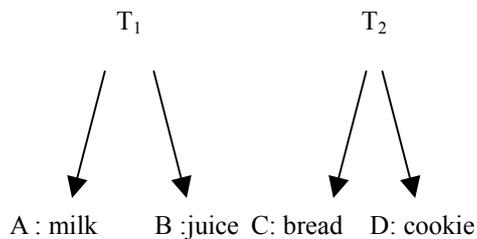


圖 8. 商品分類樹範例



Step1. 透過商品分類樹產生交易資料轉換表

增加商品祖先並且計算各祖先項目的數量。就 T_1 而言，其數量為 A 的數量加上 B 的數量，而 T_2 的數量為 C 的數量和 D 的數量的加總。依此類推，結果如下表 3：

表 3. 交易資料轉換表

交易 ID	項 目
1	(A,3)(C,4)(D,1)($T_1,3$)($T_2,5$)
2	(B,3)(C,7)(D,7)($T_1,3$)($T_2,14$)
3	(A,2)(B,10)(D,5)($T_1,12$)($T_2,5$)
4	(A,7)(B,2)(C,8)(D,8)($T_1,9$)($T_2,16$)
5	(B,2)(C,8)(D,10)($T_1,2$)($T_2,18$)

Step2. 由交易資料轉換表建立資料方塊

利用公式 (7) 計算每一筆紀錄在 K -itemsets 的歸屬度值，並累加至資料方塊所對應之 Subcube 的模糊支持度中。

- (a) 計算 1-itemsets 的模糊支持度：
- 我們將各 1-itemset 在每一筆紀錄之歸屬度算出，累加之後儲存至對應之 Subcube 中。我們以 1-itemsets 之模糊方格表來表示，結果如表 4 所示。以項目 A.L 為例，分別計算出其 5 筆的歸屬度為 0.4、0.0、0.6、0.0、0.0，累加之後即為 A.L 的模糊支持度。

表 4. 1-itemsets 之模糊方格表

	1	2	3	4	5	FS
A.L	0.4	0.0	0.6	0.0	0.0	1.0
A.M	0.6	0.0	0.4	0.6	0.0	1.6
A.H	0.0	0.0	0.0	0.4	0.0	0.4
B.L	0.0	0.4	0.0	0.6	0.6	1.6
B.M	0.0	0.6	0.0	0.4	0.4	1.4
B.H	0.0	0.0	1.0	0.0	0.0	1.0
C.L	0.2	0.0	0.0	0.0	0.0	0.2
C.M	0.8	0.6	0.0	0.4	0.4	2.2
C.H	0.0	0.4	0.0	0.6	0.6	1.6
D.L	0.8	0.0	0.0	0.0	0.0	0.8
D.M	0.2	0.6	1.0	0.4	0.0	2.2
D.H	0.0	0.4	0.0	0.6	1.0	2.0
T_1 .L	0.4	0.4	0.0	0.0	0.6	1.4
T_1 .M	0.6	0.6	0.0	0.2	0.4	1.8
T_1 .H	0.0	0.0	1.0	0.8	0.0	1.8
T_2 .L	0.0	0.0	0.0	0.0	0.0	0.0
T_2 .M	1.0	0.0	1.0	0.0	0.0	2.0
T_2 .H	0.0	1.0	0.0	1.0	1.0	3.0

- (b) 計算 2-itemsets 的模糊支持度：

由 1-itemsets 合併成 2-itemsets 並依據 (公式 8) 計算出每一項目組的支持度。以 A.M×B.L 為例，其模糊支持度的計算如表 5 所示。在第一筆時， $\mu(A.M \times B.L) = \mu(A.M) \times \mu(B.L) = 0.6 \times 0.0 = 0.0$ ，依此類推，算出其在每一筆中的模糊歸屬度，最後再將所有五筆的模糊歸屬度加總，即為其模糊支持度，並儲存至對應的資料子方塊中。所有 2-itemsets 的模糊支持度如表 6 所示。

表 5. A.MxB.L 之模糊支持度的計算

交易ID	A.M	B.L	A.MxB.L
1	0.6	0.0	0.0
2	0.0	0.4	0.0
3	0.4	0.0	0.0
4	0.6	0.6	0.36
5	0.0	0.6	0.0
FS			0.36

(c) 計算其餘 k -itemsets 的模糊支持度：

因為本資料表共有 6 種商品項目 (含商品祖先)，所以，同理，計算其餘 k -itemsets 的模糊支持度，直到 6-itemsets 產生為止。

表 6. 2-itemsets 之模糊方格表

項目組	FS	項目組	FS	項目組	FS	項目組	FS
A.L*B.L	0.00	A.H*C.M	0.16	B.M* T ₂ .H	1.40	C.H* T ₁ .L	0.40
A.L*B.M	.32	A.H*C.H	0.24	B.H*C.L	0.00	C.H* T ₁ .M	0.72
A.L*B.H	.60	A.H*D.L	0.00	B.H*C.M	0.00	C.H* T ₁ .H	0.48
A.L*C.L	.08	A.H*D.M	0.16	B.H*C.H	0.00	C.H* T ₂ .L	0.00
A.L*C.M	.32	A.H*D.H	0.24	B.H*D.L	0.00	C.H* T ₂ .M	.00
A.L*C.H	.00	A.H* T ₁ .L	0.00	B.H*D.M	1.00	C.H* T ₂ .H	.60
A.L*D.L	.32	A.H* T ₁ .M	0.08	B.H*D.H	0.00	D.L* T ₁ .L	0.32
A.L*D.M	.68	A.H* T ₁ .H	0.32	B.H* T ₁ .L	0.00	D.L* T ₁ .M	0.48
A.L*D.H	.00	A.H* T ₂ .L	0.00	B.H* T ₁ .M	0.00	D.L* T ₁ .H	0.00
A.L* T ₁ .L	.16	A.H* T ₂ .M	0.00	B.H* T ₁ .H	1.00	D.L* T ₂ .L	0.00
A.L* T ₁ .M	.24	A.H* T ₂ .H	0.40	B.H* T ₂ .L	0.00	D.L* T ₂ .M	0.80
A.L* T ₁ .H	.60	B.L*C.L	0.00	B.H* T ₂ .M	1.00	D.L* T ₂ .H	0.00
A.L* T ₂ .L	0.00	B.L*C.M	0.72	B.H* T ₂ .H	0.00	D.M* T ₁ .L	0.32
A.L* T ₂ .M	1.00	B.L*C.H	0.88	C.L*D.L	0.16	D.M* T ₁ .M	0.56
A.L* T ₂ .H	0.00	B.L*D.L	0.00	C.L*D.M	0.04	D.M* T ₁ .H	1.32
A.M*B.L	0.36	B.L*D.M	0.48	C.L*D.H	0.00	D.M* T ₂ .L	0.00
A.M*B.M	0.24	B.L*D.H	1.12	C.L* T ₁ .L	0.08	D.M* T ₂ .M	1.20
A.M*B.H	0.40	B.L* T ₁ .L	0.40	C.L* T ₁ .M	0.12	D.M* T ₂ .H	1.00
A.M*C.L	0.12	B.L* T ₁ .M	0.72	C.L* T ₁ .H	0.00	D.H* T ₁ .L	0.56
A.M*C.M	0.72	B.L* T ₁ .H	0.48	C.L* T ₂ .L	0.00	D.H* T ₁ .M	0.84
A.M*C.H	0.36	B.L* T ₂ .L	0.00	C.L* T ₂ .M	0.20	D.H* T ₁ .H	0.48
A.M*D.L	0.48	B.L* T ₂ .M	.00	C.L* T ₂ .H	0.00	D.H* T ₂ .L	0.00
A.M*D.M	0.76	B.L* T ₂ .H	.60	C.M*D.L	0.64	D.H* T ₂ .M	.00
A.M*D.H	0.36	B.M*C.L	0.00	C.M*D.M	0.68	D.H* T ₂ .H	.00
A.M* T ₁ .L	0.24	B.M*C.M	0.52	C.M*D.H	0.88	T ₁ .L* T ₂ .L	0.00
A.M* T ₁ .M	0.48	B.M*C.H	0.72	C.M* T ₁ .L	0.72	T ₁ .L* T ₂ .M	0.40
A.M* T ₁ .H	0.88	B.M*D.L	0.00	C.M* T ₁ .M	1.16	T ₁ .L* T ₂ .H	0.80
A.M* T ₂ .L	0.00	B.M*D.M	0.52	C.M* T ₁ .H	0.32	T ₁ .M* T ₂ .L	0.00
A.M* T ₂ .M	1.00	B.M*D.H	0.88	C.M* T ₂ .L	0.00	T ₁ .M* T ₂ .M	0.40
A.M* T ₂ .H	0.60	B.M* T ₁ .L	0.40	C.M* T ₂ .M	0.80	T ₁ .M* T ₂ .H	0.40
A.H*B.L	0.24	B.M* T ₁ .M	0.68	C.M* T ₂ .H	1.40	T ₁ .H* T ₂ .L	0.00
A.H*B.M	0.16	B.M* T ₁ .H	0.32	C.H*D.L	0.00	T ₁ .H* T ₂ .M	1.00
A.H*B.H	0.00	B.M* T ₂ .L	0.00	C.H*D.M	0.48	T ₁ .H* T ₂ .H	0.80
A.H*C.L	0.00	B.M* T ₂ .M	0.00	C.H*D.H	1.12		

Step3. 找出各維度之高頻項目組

透過 $minfs = 1.6$ ，去搜尋資料方塊中符合 $minfs$ 之項目組，即為各維度之高頻項目組。以 1-itemsets 為例，符合的高頻 1-itemsets 如表 7。符合 $minfs$ 的高頻 2-itemsets 如表 8。依此類推找出每一維的高頻項目組，此範例從 3 維至 6 維中無高頻項目組產生。

表 7. 高頻 1-itemsets

項目組	FS
A.M	1.6
B.L	1.6
C.M	2.2
C.H	1.6
D.M	2.2
D.H	2.0
T ₁ .M	1.8
T ₁ .H	1.8
T ₂ .M	2.0
T ₂ .H	3.0

表 8. 高頻 2-itemsets

項目組	FS
B.L* T ₂ .H	1.60
C.H* T ₂ .H	1.60
D.H* T ₂ .H	2.00

Step4. 產生法則

由各維度之高頻 k -項目組產生候選法則並計算其信賴度，再與使用者制定的最小模糊信賴度 $minfc = 0.70$ 比對，符合 $minfc$ 的法則即為使用者想要的法則。以此範例為例，其結果如表 9 所示。B.L→T₂.H 的信

賴度為 $1.6 / 1.6 = 1$ ；T₂.H→B.L 的信賴度為 $1.6 / 3.0 = 0.53$ ；C.H→T₂.H 的信賴度為 $1.6 / 1.6 = 1$ ；依此類推。其中符合 $minfc$ 的法則為 B.L→T₂.H、C.H→T₂.H、D.H→T₂.H 與 T₂.H→D.H，即為產生的法則。

表 9. 關聯法則表

法則	FC
B.L→T₂.H	1
T ₂ .H→B.L	0.53
C.H→T₂.H	1
T ₂ .H→C.H	0.53
D.H→T₂.H	1
T₂.H→D.H	1

在漸進式挖掘的部分，假設新增表 10 的 3 筆交易紀錄，則我們可以即時更新資料方塊中對應的項目組的模糊支持度。以 1-itemsets 為例，其更新的動作列表如表 11。此時，若 $minfs$ 變成 2.56，則滿足 $minfs$ 的高頻 1-itemsets 如表 12 所示。其他項目組的模糊支持度也依此類推來進行更新，並找出高頻項目組以產生法則。

表 10. 新增資料表

交易ID	項 目
6	(A,3)(C,4)(D,1)(T ₁ ,3)(T ₂ ,5)
7	(B,3)(C,7)(D,7)(T ₁ ,3)(T ₂ ,14)
8	(A,2)(B,10)(D,5)(T ₁ ,12)(T ₂ ,5)

表 11. 1-itemsets 之模糊支持度更新

項目組	新增資料之 FS	新增前之 FS	新增後之 FS
A.L	1.0	1.0	2.0
A.M	1.0	1.6	2.6
A.H	0.0	0.4	0.4
B.L	0.4	1.6	2.0
B.M	0.6	1.4	2.0
B.H	1.0	1.0	2.0
C.L	0.2	0.2	0.4
C.M	1.4	2.2	3.6
C.H	0.4	1.6	2.0
D.L	0.8	0.8	1.6
D.M	0.8	2.2	3.0
D.H	0.4	2.0	2.4
T ₁ .L	0.8	1.4	2.2
T ₁ .M	1.2	1.8	2.0
T ₁ .H	1.0	1.8	2.8
T ₂ .L	0.0	0.0	0.0
T ₂ .M	2.0	2.0	4.0
T ₂ .H	1.0	3.0	3.0

表 12. 新增後的高頻 1-itemsets

項目組	FS
A.M	2.6
C.M	3.6
D.M	3.0
T ₁ .H	2.8
T ₂ .M	4.0
T ₂ .H	3.0

伍、結論與未來展望

在現今競爭激烈與資訊科技快速成長的環境下，資料庫的管理與應用也逐漸被企業所重視，利用資料挖掘技術可以幫助企業從龐大且複雜的資料庫中找到隱藏的資訊，特別是關聯法則，透過最小支持度

及最小可靠度的設定，我們可以從龐大的資料量中，找出商品項目間的關聯性。

然而，面對多樣化的商品項目時，往往很難求出能滿足最小支持度的關聯法則，有時候會將一些具有高價值性的資訊遺漏而不自知。為避免這種遺憾發生，便有研究利用歸納的方式，將商品項目進行階層化分類的動作。透過不同層次的分類，我們便可以對不同層次做關聯法則的求取。但過去傳統的多階層的關聯法則大部分適用在二值屬性的資料，對於其他類型的資料就很難去進行處理。加上，現今的資料庫的資料隨時在變動，過去的演算法無法有效地在運用，導致當資料變更時必須重新做一次關聯法則挖掘。因此，本研究利用模糊分割的技術來解決多階層的關聯法則在多類型的資料處理上的問題以便讓企業界的決策者在面對如此快速變化且競爭的環境下，能夠有效地掌握住產品供需的大方向，確認發展的目標，以下達正確的指令。另外，結合資料方塊的預先儲存資料以及多維資料模式表示的技術，能夠快速因應資料庫變動而即時更新內容，進而達到漸進式挖掘的優勢。

本研究利用模糊切割以及資料方塊的技術來挖掘漸進式多階層關聯法則，未來的研究與應用可朝著以下的方向進行：

- (一) 就模糊切割而言：(a)在模糊理論中有相當多種模糊歸屬度函數，不同的模糊隸屬函數可能會產生不同影響；(b)各維所切割的數量 K 要各設多

少會有比較好的鑑別度並且產生最有效的法則。未來我們將實際深入探討這些層面的影響。

- (二) 就多階層關聯法則而言：我們可以探討商品項目之階層化架構的深度對多階層關聯法則挖掘的效率與品質之影響。
- (三) 資料方塊的記憶體配置與管理：當商品項目愈來愈多時，資料方塊的維度愈來愈複雜且龐大，這也導致建構與執行資料方塊的效率問題，甚至產生不可行的現象。為了解決這樣子的問題，本研究是採取類似ROLAP(Relational OLAP)的方式來配置與管理資料方塊的記憶體。未來，我們可以結合分散式平行處理的技術把資料方塊分成主資料方塊與暫存資料方塊，主資料方塊主要是儲存大項目組而暫存資料方塊主要是儲存非大項目組。當資料變動時，若原本非大項目組變更為大項目組時，此項目會移至主資料方塊；相反地，若原本為大項目組變更為非大項目組時，此項目會移至暫存資料方塊中。
- (四) 應用方面：本研究所提的多階層關聯法則挖掘的結果可以實際應用在很多電子商務系統的經營與行銷上，作為制定行銷策略的參考。未來，我們將結合其他技術以便廣泛地應用到電子商務系統的個人化行

銷、線上推薦與增進顧客價值等功能上，例如，(a)藉由所建立之專屬會員制度，透過本研究的漸進式模糊多階層關聯法則，從龐大的會員交易資料中挖掘消費特性，以實現個人化之服務，有效區隔市場與訂定行銷策略[6]；(b)透過本研究的漸進式模糊關聯法則來探討RFM(最近購買時間(Recency)購買頻率(Frequency)購買金額(Monetary))三個因素彼此之間的關聯性，以便提供更具彈性且有效的資訊，作為企業在制定客製化行銷策略的正確決策上的一個有力的參考依據[17]。

參考文獻

- [1] 王慶堯，民國 89 年，利用準大項目集之漸進式挖掘，義守大學資訊工程研究所碩士論文。
- [2] 林桂英、洪宗貝，民國 89 年，從數值型資料挖掘模糊多層及關聯規則及模糊順序性樣式，義守大學資訊工程學系碩士論文。
- [3] 武家慶、楊東麟，民國 90 年，適用於挖掘多層次聯結規則之架構，逢甲大學資訊工程學系碩士論文。
- [4] 許智豪，民國 89 年，在動態資料庫中作線上挖掘關聯式法則，國立中興大學資訊科學研究所碩士論文。

- [5] 楊昇樺、毛立人、邱宏彬，民國 91 年，『關聯法則之多層更新挖掘法及其應用』，刊於第三屆產業資訊管理學術暨新興科技實務研討會論文集(下集)，pp. 549-557.
- [6] 蔡玉娟、張簡雅文、黃彥文，民國 92 年，快速反向關聯法則與調整緊密規則-促銷商品組合之應用，資訊管理學報，第十卷，第一期，pp. 181-204.
- [7] D. W. Cheung, S. D. Lee, and B. Kao, "A general incremental technique for maintaining discovered association rules", in Proceedings of Database Systems for Advanced Applications, DASFAA'97, Melbourne, Australia, pp. 185-194, 1997.
- [8] H. Toivonen, "Sampling Large Database for association Rule", VLDB, pp. 134-145, 1996.
- [9] H. Ishibuchi, K. Nozaki, and H. Tanaka, "Distributed representation of fuzzy rules and its application to pattern classification," *Journal of Fuzzy Sets and Systems*, Vol.52 (1), pp.21-32, 1992.
- [10] Jiawei Han and Youngjian Fu, "Discovery of multiple-level association rules from large database," *The International Conference on Very Large Databases*, 1995.
- [11] J. S. Park, M. S. Chen, and P. S. Yu, "An Effective Hash Based Algorithm for Mining Association Rules", Proc. ACM SIGMOD, pp. 175-186, 1995.
- [12] M. S. Chen, J. Han, and P.S. Yu, "Data Mining: An Overview from a database Perspective", IEEE Transaction on Knowledge and Data Engineering, Vol.8, No. 6, December 1996.
- [13] R. Agrawal and R. Srikant, "Fast Algorithm for Mining Association Rule in Large Databases", In Proc. 1994 Int'l Conf. VLDB, pp. 487-499, Santiago, Chile, Sep. 1994.
- [14] R. Agrawal and R. Srikant, "Mining Quantitative Association Rules in Large Relational Tables," *In proc. of ACM SIGMOD*, PP.1-12, Moutreal Canada, 1995.
- [15] Ramakrishnan Srikant and Rakesh Agrawal, "Mining generalized association rules," *Proc. of the 21st VLDB*, pp. 407-419, 1997.
- [16] S. Brin, R. Motwani, J. D. Ullman, and S. Tsur, "Dynamic Itemset Counting and Implication Rules for Marketing Basket Data", 1997 ACM SIGMOD Conference on Management of Data, pp. 255-264, 1997.
- [17] Sung Ho Ha and Sang Chan Park, "Application of data mining tools to hotel data mart on the Intranet for database marketing," *Journal of Expert Systems with Applications*, Vol.15, pp.1-31, 1998.
- [18] Tzung-Pei Hong, C. S. Kuo and S. C. Chi, "A data mining algorithm for transaction data with

- quantitative values,” *Intelligent Data Analysis*, Vol.3, No.5, pp.363-376, 1999.
- [19] Tzung-Pei Hong, Kuei-Ying Lin, and Shyue-Liang Wang, ”Fuzzy data mining for interesting generalized association rules,” *Fuzzy sets and systems* ,Vol. 138, pp.255-269, 2003.
- [20] Yen John and Reza Langari, *Fuzzy Logic: Intelligence ,Control, and Information*, Prentice-Hall Inc., pp.70-71, 1999.
- [21] Yi-Chung Hu, Ruey-Shun Chen, and Gwo-Hshiung Tzeng, ”Discovering fuzzy association rules using fuzzy partition methods,” *Journal of Knowledge-Based Systems*, Vol.16, pp.137-147, 2003.