

運用二次叢集法來對數值、類別混合型資料探勘

施明毅 鄭佳文* 賴聯福
彰化師範大學資訊工程學系
彰化、台灣

{myshih, lfai}@cc.ncue.edu.tw, *m95612021@mail.ncue.edu.tw

摘要

近年來每日資料的產生量成爆炸性的成長，面對如此巨量的資料，資料探勘中的叢集法(clustering)，可以將資料分成數個子集合，再從這些子集合中發現一些隱藏的知識，已廣泛被運用到各領域的資料。然而目前大部份傳統的叢集法都將焦點擺在純數值型資料或純類別型資料，但由真實世界所收集的資料大都混合數值型與類別型屬性，若想直接將傳統叢集演算法應用在混合型資料是困難的。因此本文提出一個運用二次叢集新方法，結合階層式叢集方法和分割式叢集方法來對混合屬性資料作叢集運算。此方法根據 co-occurrence 理論找出資料中類別型屬性值間共同出現的關係，再利用這些關係把類別型屬性轉換為數值型屬性，當屬性已全部轉為數值型，資料就可被運用到距離型叢集演算法中了。然而目前發展出來的叢集演算法都有其限制或缺點，因此本文結合兩種叢集演算法並將其應用到數值和類別屬性混合的資料集中。此方法不僅保留了屬性值間的關係，而且改善單一階段叢集演算法的缺陷，實驗數值顯示此方法可在叢集數值、類別混合資料時，得到良好的品質。

關鍵字: 資料探勘、叢集、混合型資料

I. 前言

隨著資訊發展，所能取得的資料越來越多，再加上電腦科技應用日益廣泛，資料的型態也越來越複雜，這對資料的瞭解和進一步應用是一大挑戰。其最大挑戰就是在有限的資源中(i.e., 電腦記憶體和執行時間)對大量且複雜的資料運算，來找出隱藏在資料中的知識。毫無疑問的，資料只有當它可以被找出有用的資訊時才有意義，因此利用資料探勘技術來獲取資料中的 patterns，最近幾年大受歡迎。資料探勘是一種整合科技，其包含資料庫與資料倉儲技術、統計學、機器學習、人工智慧、高速計算、樣本分析、類神經網路、資料視覺化、資訊檢索、影像與信號處理和空間與時間性資料分析，其所應用領域相當廣泛。

叢集法(clustering)是資料探勘中常見且重要的技巧。主要目的是將資料集中的資料分群成數個叢集(cluster)，使得每個叢集內部資料相似度越高越好；而叢集間資料的差異性越高越好[1]。每個叢集中包含相似

的物件，此時分析這些物件就可推論出群組的特徵。叢集法已經被應用在許多不同領域，例如在商業行銷上，將顧客依據其購物習性加以分群，來瞭解其購物行為，幫助制定行銷策略。在網際網路上，將使用者的依瀏覽網頁的資料分群，可以發現使用者對網頁的使用行為，進而了解提升所需提供的網頁服務內容。在生物學上，將生物基因依功能進行分類，來幫助瞭解其功能。在文件探勘上，文件依出現的字彙分群，可以幫助資訊檢索。另外對其演算法而言，叢集法不需要事先定義分類別，可對未知類別的資料分群，再根據分群結果預測出相似的資訊。

現以有許多叢集演算法[1]-[11]被發展出來並應用在統計學、機器學習、圖形辨識和資料庫系統等領域上。對於大多數的叢集演算法而言，一個物件通常被視為多維空間中的一點。選取 d 個屬性以 d 維向量 (x_1, \dots, x_d) 來表示各物件；其中 x_i 代表所選取的第 i 個屬性值， x_i 的值可以是數值型或類別型。但是這些叢集演算法大部份都將焦點擺在純數值型資料或純類別型資料，但真實世界收集的資料大都混合數值型與類別型屬性，若想直接將傳統叢集法應用在數值與類別混合的資料是困難的。早期最直接的作法是將數值型屬性劃分出數個區段來轉換成為類別型屬性，或任意分配一數值給類別型屬性，來造成屬性值的單一性，但此類做法會因其誤差大而影響叢集品質。

K-prototype[12] 是最知名的對混合型資料叢集先驅者之一，此演算法對於數值型屬性的部份仍延用歐基里德距離公式來計算；而類別型屬性則是判斷類別屬性值相同與否，若不相同即擴大兩資料的距離。其最大缺點可規納為下列三點：

1. 叢集中心的類別屬性值選取瑕疵：k-prototype 選取叢集中出現最多次的類別值來代表叢集中心，但其它出現較少次的類別值可能對叢集中心也很重要。
2. 只有兩種類別屬性值距離：k-prototypes 在判斷資料物件與叢集中心的類別屬性距離時，直接比較類別值是否相同，若相同則其距離為零，否則距離為壹。但很明顯的，不同的類別值間距離應該有所不同，而不是都為零或壹。例如：「大」與「中」的距離應該比「大」與「小」的距離來得近。
3. 沿襲了 k-means 的缺點：因為 k-prototypes 是以 k-means 為基礎，再增加判斷類別屬性值的規則，因此也繼承了 k-means 易受 noise 影響及初始點選取會干擾叢集結果的缺點。

後來此類演算法的發展大都針對上述缺點來加以改進。Yin et al. [13] 和 Ahmad et al. [14] 更改叢集中心的表示法，不再採用單一類別屬性值作為叢集中心，而是列出中所有類別屬性值各別出現的比例，因此不論類別屬性值出現次數多或少，都將被考慮在分群的依據中。Chiu et al. [15] 依據兩叢集間類別屬性值的一致性決定兩叢集的類別屬性距離，其亦解決 k-prototypes 只有兩種類別值距離的缺點。此外其先利用 BIRCH [5] 建 CF-tree，而 CF-tree 的子葉部份即為第一階段的叢集結果，接著將各叢集視為單一資料，再利用階層式叢集法作運算。Li et al. [16] 提出給予出現次數少的類別值較大的權值的概念，當兩筆資料的距離會比那些類別值出現次數多資料的還要更接近。He et al. [17] 提出以數值屬性值與叢集中各物件的數值平均差計算物件與叢集的數值屬性距離；而物件之類別值於叢集中出現的比例來代表其與叢集間的類別屬性距離。

但是以上演算法大都將各屬性視為獨立個體，忽略彼此之間的相關性。例如購買豪宅的人，通常會有高收入，開豪華轎車，帶高級手表，和買一些名牌商品。因此本文提出一個對混合屬性資料叢集的新方法-TMCM (a Two-step Method for Clustering Mixed numeric and categorical data)。此方法先根據 co-occurrence 的理論，找出屬性間相互的關聯性，再依據此關聯性將所有類別形屬性轉換為數值型屬性。如此一來就可將資料帶入一般叢集演算法。但一般叢集演算法都有其先天上的限制或缺點，因此在此研究將採用混合型的叢集演算法，來改善此一缺點。此研究所提出方法不僅保留了屬性值之間的關聯性，而且改善單一階段叢集演算法的缺陷，因此在對混合數值和類別型資料叢集運算時能得到良好品質。

II. TMCM 演算法

通常作為叢集演算法輸入的屬性是選取資料集中具代表性的特徵。大部份傳統叢集演算法假設屬性與屬性是彼此獨立、沒有關係的，且在計算物件間相似度時也會將各屬性看成獨立的個體，但實際上屬性與屬性之間是可能存在關聯的。因此在此研究中提出的 co-occurrence 概念就是去找出各屬性的相互關係，並將這關係加入叢集演算法的分群依據中。所謂 co-occurrence 就是若兩個項目經常同時出現在同一物件，則代表此兩項目具有相當程度的關聯性。若兩類別型屬性中的項目有相當高的關聯性，則在轉換成數值資料過程中，他們應該擁有相近的值。如在表 1 的例子中當溫度為 hot 時溼度是 high，而溫度為 cool 時溼度可能是 normal 或 low。因此就溫度與溼度之間的關係而言，hot 與 high 之間的距離會比 cool 與 high 之間的距離來得接近。此時若要分配數值給類別形屬性值時，hot 和 high 應該得到比 cool 和 high 更相近的數值，因為此 hot 和 high 二項目有很高的關聯性。

表 1. co-occurrence 例子

溫度	溼度	風
hot	high	false
hot	high	false
cool	low	true
cool	normal	true

本文所提出的 TMCM 就是基於此理論來將類別型屬性轉換為數值型屬性。完整演算法如下：

TMCM Algorithm:

- ```

{
 //資料前處理 phase
 1. 讀取輸入的資料，並將數值型屬性常態化。
 2. 找出類別值最多的類別屬性作 base attribute。
 3. 計算各類別值與 base items 伴隨出現的次數，以建矩陣 M。
 4. 根據矩陣 M 計算各類別值與 base items 的相似度，紀錄於矩陣 D。

 //類別型屬性數值化 phase
 5. 找出與 base attribute 組內變異數最小的數值屬性，求出各 base item 在此數值型屬性的平均值，即可對 base items 數值化。
 6. 將數值化後的 base items 以矩陣 D 中與其它類別屬性值的相似度作為權值求出其餘的類別屬性值的平均值，即可對其它類別屬性值數值化。

 //以混合型叢集演算法計算資料，評估分群品質 phase
 7. 先以 HAC 階層式叢集演算法將資料分為若干子群。(在本實驗發現分為原資料量的 1/3 子群，是一適當的設定。)
 8. 計算 step 7 叢集中心點，再加入每個類別型屬性的個數當新屬性。
 9. 將 step 8 的資料叢集再以 K-means 將資料分成想要的 k 群。
 10. 計算 entropy 評估分群結果。
}

```

所提出的方法第一個步驟為資料前處理。因為所收集的資料來源是多方面，在此步驟先將資料做處理，將沒有關係的項目刪除，將格式轉換為特定模式，並將數值型屬性值常態化，使其介於 0~1 之間。常態化的目的為避免某些數值範圍較大的屬性主宰分群結果。

此方法會選取擁有最多類別值的類別屬性 A 作為 base attribute。此策略是讓其他非 bases item 可對映到多個 base items，而不會產生多個非 base items 對映到一個 base item

的現象。接著分別計算所有類別值各自出現的次數，還有與 base items 同時出現的次數。並以一個  $n \times m$  的矩陣  $M$  來儲存這些資訊。

$$M = \begin{pmatrix} m_{11} & m_{12} & \dots & m_{1m} \\ m_{21} & m_{22} & \dots & m_{2m} \\ \dots & \dots & \dots & \dots \\ m_{n1} & m_{n2} & \dots & m_{nm} \end{pmatrix}$$

其中  $n$  為 base item 的個數；  
 $m$  為 base attribute 以外其餘類別屬性值的總數；  
 $m_{ij}$  代表  $M$  中 base items  $i$  與類別屬性值  $j$  同時出現在同一物件的次數。

建完矩陣  $M$  後，即可計算 base items 和其它類別屬性值的相似度。並將這些相似度值儲存在與  $M$  同樣大小的矩陣  $D$ 。計算公式如下：

$$D_{xy} = \frac{|m(X,Y)|}{|m(X)| + |m(Y)| - |m(X,Y)|} \quad (1)$$

其中  $x$  代表類別值  $x$ ；  
 $y$  代表類別值  $y$ ；  
 $X$  代表在資料集中類別值為  $x$  的條件；  
 $Y$  代表在資料集中類別值為  $y$  的條件；  
 $m(X)$  指類別值為  $x$  出現的資料集合；  
 $m(X, Y)$  指類別值  $x$  和  $y$  一起出現的資料集合。

在此公式中，若某兩屬性值都會一起出現，則其相似度為 1；但若都沒一起出現過則相似度為 0，因此  $D_{xy}$  值越大代表屬性值  $x$  和  $y$  的相似度越高，且其值在 0-1 之間。因為矩陣  $D$  中記錄了類別值之間的相似度，因此可依此相似度定義類別值之間的關係，在此研究並且定義相似度門檻值，只有當類別值之間的相似度大於門檻值時才會儲存該相似度，否則以零表示。

接下來將對類別屬性的各類別值定量。先利用 co-occurrence 的概念以數值型屬性對 base items 數值化。此數值型屬性的選擇依據為計算各數值型屬性與 base attribute 的單因子變異數之組內變異，並選取使組內變異數最小的數值型屬性。組內變異數計算公式如下：

$$SS_w = \sum_j \sum_i (X_{ij} - \bar{X}_j)^2 \quad (2)$$

其中  $\bar{X}_j$  代表第  $j$  個 base item 對應數值屬性  $w$  的平均數；

$X_{ij}$  代表第  $j$  個 base item 對應數值屬性  $w$  內第  $i$  筆數值。

將 base items 對應選出的數值型屬性求出各 base item 平均值，即可對 base items 數值化。接下來利用下列公式以數值化後的 base items 再對其餘的類別型屬性值  $x$  數值化。

$$F(x) = \sum_{i=1}^d a_i * v_i \quad (3)$$

其中  $d$  為 base item 數；  
 $a_i$  為類別屬性值  $x$  和第  $i$  個 base item 的相似度；  
 $v_i$  是第  $i$  個 base item 數值化後的值。

至此，所有的類別型屬性都被轉換為數值型屬性，因此資料中全部都是數值型屬性，此時可將資料輸入距離型叢集演算法了。在此研究中，為改善單一叢集演算法的缺點，提出一種二次叢集演算法。HAC (Hierarchical Agglomerative Clustering)[2] 演算法是一種階層式的叢集方法，它先將每一資料點視為一群，在依據相似性將最相近的群合併，直到所有群都已合併或想要的群數目達到。其優點是不必輸入參數，但其執行速度太慢是其缺點。因此在此計劃中，一開始將所有資料利用 HAC 演算法，將全部資料合併成 1/3 數目的子群，再將這些生成的群集計算中心點，並加入以類別型屬性值，成為新的物件帶入下一階段 k-means[2] 的計算。

表 2 舉例如何新增屬性。假設在第一階段生成的叢集 1 中包含四筆資料，在下一層分群中將以叢集中心來代表這四筆資料。並將叢集中心由原本的三維屬性增加為 10 維。新增的七維屬性分別代表各類別值(hot、cool、mild、high、normal、FALSE、TRUE)在此叢集中的出現次數，如表 3 所示。

表 2. 叢集 1 包含四筆具有三維屬性的資料

| temperature | humidity | windy |
|-------------|----------|-------|
| hot         | high     | FALSE |
| hot         | high     | FALSE |
| cool        | normal   | TRUE  |
| mild        | normal   | TRUE  |

表 3. 以叢集內各類別值出現次數當新增屬性範例

| hot | cool | mild | high | normal | FALSE | TRUE |
|-----|------|------|------|--------|-------|------|
| 2   | 1    | 1    | 2    | 2      | 2     | 2    |

因為此階段所形成的子群不再是單一物件，而是可能多個相似物件聚集成一族群，此時再把這些子群當成新物件做另一次叢集計算。因為 k-means 是分割式叢集法中最為人熟知、發展最久的一種方法，且由於其叢集概念與實際上均相當簡單，且在處理上所需的時間與空間成本都相當低，因此至今仍最廣為採用，因此在第二階段採用此方法。K-means 在第二階段的輸入是前階段所形成的子群，不是單一物件，因此可降低 noise 及 outliers 對 k-means 的影響，而提高分群的品質。此外利用此方法也可降低選取初始點對結果影響的問題。當叢集完成後，其結果用 Entropy 公式來評估叢集品質，並和其他方法做比較。

Entropy 可用來判斷資料中各類性質分佈的均衡程度，也就是判斷在同一叢集中，組成元素分佈的情形，通常在組成元素均勻分佈且出現機率相等時，entropy 值最高，相反的，若是組成元素中以某一類元素出現機率較高，則所得的 entropy 較低。也就是說，在某一叢集中，所包含某一類元素機率較高，且特別集中時，則其 entropy 會降低。因此在實驗中 entropy 值越小代表分群所得的叢集越能代表資料中的某個類別，因此被認定叢集品質越佳。

Entropy 公式如下：

$$E = - \sum_{j=1}^m ((n_j / n) * \sum_{i=1}^l P_{ij} * \log(P_{ij})) \quad (4)$$

其中  $m$  為叢集數；  
 $l$  為資料集的分類總數；  
 $n_j$  為叢集  $j$  的資料數；  
 $n$  為總資料數；  
 $P_{ij}$  表資料位於叢集  $j$  且屬於類別  $i$  的機率。

### III. 實驗結果

本文所提出的方法應用在 UCI 資料庫 (<http://archive.ics.uci.edu/ml/datasets.html>) 所取得的 3 組資料集合，來評估此方法的表現。這些資料集合都已經分類好，因此只需將叢集結果與資料所屬類別相比較，即可依 entropy 值來評估叢集品質。因為 k-prototype 是目前相當有名的混合型屬性叢集演算法，也廣泛被引用，因此本實驗將與其叢集結果來比較。此外 SPSS Clementine 是目前相當受歡迎的一個資料探勘商業軟體，其採用[15]的演算法，來對混合型屬性資料做叢集運算，因此該軟體也被列入比較的對象。

1) Contraceptive method choice data set: 本資料集合描述 1987 年印尼已婚婦女所採用的避孕方法。分為不使

用、長期法與短期法三種。共有 1473 筆資料、9 個屬性，其中 2 個是數值型、7 個是類別型。

表 4 為將三種演算法對此資料做叢集計算所得到的結果。第二行顯示資料被分為跟原資料相同類別的數目後得到的 entropy 的值，第三行為利用本文所提出的 TMCM 方法跟 k-prototype 和 SPSS Clementine 二種方法 entropy 值改善比率；表 5 和表 6 也是相同的表示方式。

表 4. 各演算法將 Contraceptive method choice data set 叢集後之 entropy

| 演算法             | entropy      | TMCM 改善比例      |
|-----------------|--------------|----------------|
| k-prototype     | 0.955436 (A) | 37.15% (A-C)/C |
| SPSS Clementine | 0.852992(B)  | 22.45% (B-C)/C |
| TMCM            | 0.696616(C)  |                |

2) Heart disease data set: 這個資料集合共有 270 筆資料、13 個屬性，其中 5 個是數值型和 8 個是類別型。這 13 個屬性用來判斷各資料所描述的病患其血管阻塞程度是否超過 50%，而原資料亦視病患血管阻塞是否超過 50% 分為 2 類。

表 5 為將三種演算法對此資料做叢集計算所得到的結果。

表 5. 各演算法將 Heart disease data set 叢集後之 entropy

| 演算法             | Entropy      | TMCM 改善比例      |
|-----------------|--------------|----------------|
| k-prototype     | 0.442941 (A) | 4.91% (A-C)/C  |
| SPSS Clementine | 0.466809(B)  | 10.57% (B-C)/C |
| TMCM            | 0.422194(C)  |                |

3) Credit approval data set: 這個資料集合描述各屬性對信用卡認證的影響。為了保護資料的機密，各屬性名稱與類別屬性值都被轉換成無意義的符號，但這並不影響實驗。資料集合共有 653 筆資料、15 個屬性。其中有 6 個是數值型屬性；9 個是類別型屬性，而資料分為兩類。

表 6 為將三種演算法對此資料做叢集計算所得到的結果。

表 6. 各演算法將 Heart disease data set 叢集後之 entropy

| 演算法             | entropy     | TMCM 改善比例      |
|-----------------|-------------|----------------|
| k-prototype     | 0.957794(A) | 47.55% (A-C)/C |
| SPSS Clementine | 0.764163(B) | 17.72% (B-C)/C |
| TMCM            | 0.649113(C) |                |

由表 4-表 6 可得知，上述三個資料集合，採用本文所提出的 TMCM 二次叢集法後，entropy 值與 k-prototype 相比平均可改善 29.87%，與 SPSS Clementine 相比平均可改善 16.91%。

