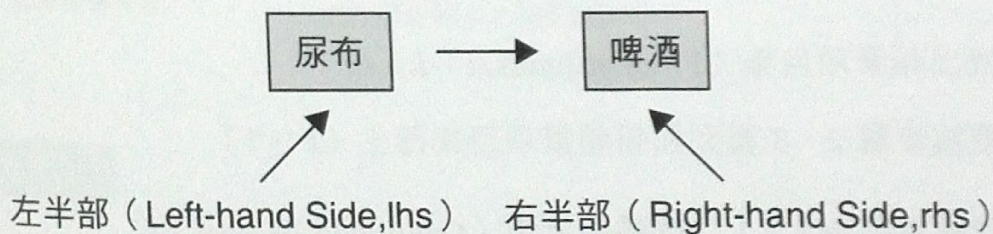


## 關聯性規則

關聯性規則最早是由 Agrawal 等人針對超市購物籃分析 (Market Basket Analysis) 問題提出的，其目的是發現超市交易資料庫中不同商品之間的關聯關係。關聯性規則呈現了顧客購物的行為模式，其結果可以提供做為經營決策、市場預測和制定銷售策略的參考依據。以尿布及啤酒的購物籃為例：



此規則表示尿布及啤酒銷售具有關聯性。

假設總共有  $N$  筆交易，定義  $supp(A)$  為購買項目  $A$  的支持度 (Support)， $conf(A \rightarrow B)$  為購買項目  $A$  也會購買項目  $B$  的信賴度 (Confidence)。支持度可用來判斷規則的有效性；信賴度用來判斷在項目  $A$  的條件下發生項目  $B$  的可能性，其值越高，規則就越具有參考價值。一個強關聯性規則，通常支持度和信賴度值都高；但反過來支持度和信賴度值都高，卻不一定代表此關聯性規則所指的事件彼此間就一定存在著高相關性，同時還需檢查提升度 (Lift) 是否大於 1，提升度  $> 1$ ，表示項目  $A$  與項目  $B$  間有正向關係；提升度值  $= 1$ ，表示項目  $A$  與項目  $B$  間沒有關係；提升度  $< 1$ ，表示項目  $A$  與項目  $B$  間為負向關係。支持度、信賴度及提升度之計算方式如下表示：



$$supp(A) = freq(A)/N$$

$$supp(A \rightarrow B) = freq(A,B)/N$$

$$conf(A \rightarrow B) = freq(A,B)/freq(A)$$

$$lift(A \rightarrow B) = conf(A \rightarrow B)/supp(B)$$

其中

$freq(A)$  : 表示為購買項目 A 的數量

$freq(A,B)$  : 表示為同時購買項目 A 及 B 的數量

1994 年 Agrawal & Srikant 提出 Apriori 演算法，是目前常用之關聯性規則。Apriori 演算法採用一種逐層搜尋的疊代方法 (Level-wise Search) 先找出滿足最小支持度的頻繁項目集 (Frequent Itemsets)，再以最小信賴度為條件，計算頻繁項目集所形成的關聯性規則，當 Apriori 演算法找出滿足使用者訂定的最小支持度 (Minimum Support) 及最小信賴度 (Minimum Confidence) 的關連性規則，這個規則才算成立。Apriori 演算法步驟如下：

Step 1 找出頻繁項目集 (Frequent Itemset)  $L_1$

重複步驟 2、3 直到無新頻繁項目集產生 ( $k \geq 1$ )

Step 2 取得長度為  $k+1$  之候選項目集 (Candidate Itemset)  $C_{k+1}$

組合 (Join) : 將  $L_k$  中的項目集兩兩組合為  $C_{k+1}$

修剪 (Prune) : 修剪子集合不屬於  $L_k$  之候選項目集  $C_{k+1}$  得到長度為  $k+1$  之候選項目集  $C_{k+1}$

Step 3 找出長度為  $k+1$  之頻繁項目集  $L_{k+1}$

計數 (Count) : 計算修剪後候選項目集  $C_{k+1}$  的支持度

刪除 (Delete) : 刪除支持度未達最小支持度的候選項目集  $C_{k+1}$ ，產生長度為  $k+1$  之頻繁項目集  $L_{k+1}$

Step 4 由頻繁項目集產生關聯性規則



2000年 Zaki 提出 Eclat 演算法，它是一種深度優先的演算法，具體做法就是將交易資料庫中的項目 (Item) 作為鍵 (key)，每個項目對應的交易編號 (TID) 作為值 (Value)。Eclat 演算法如下圖表示：

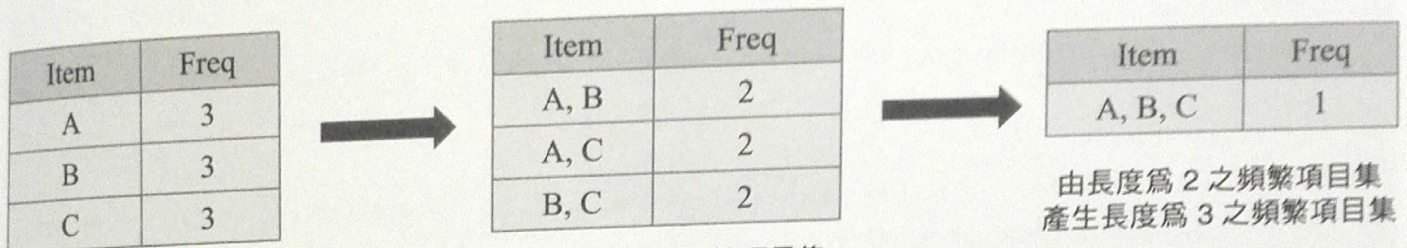
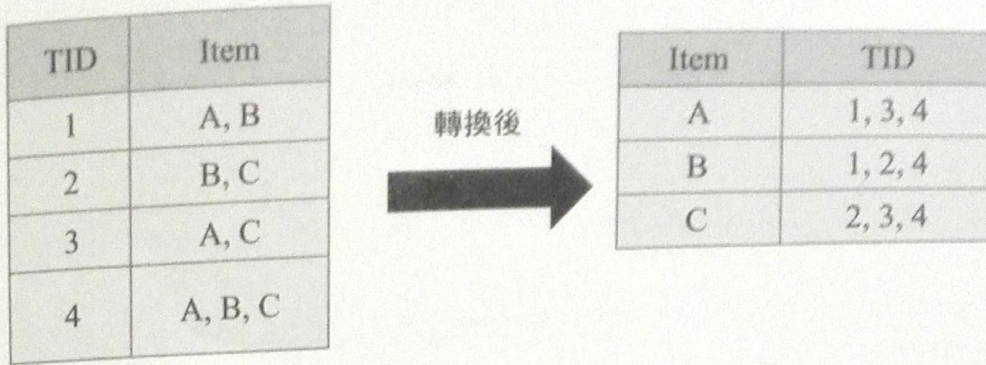


圖 10-1 Eclat 演算法

### 程式範例 10-1

首先使用 arules 套件：

```
> library(arules)
```

使用 Adult 資料集：

```
> data("Adult")
```

設定支持度 = 0.5、信賴度 = 0.9 及不顯示執行過程的相關資訊 (verbose=F)：

```
> rules <- apriori(Adult,parameter = list(supp = 0.5, conf = 0.9),control=list(verbose=F))
```