

106-2: EE4052

通識課程：

計算機程式設計 之旅

Computer Programming

Unit 14: 資料連結分析

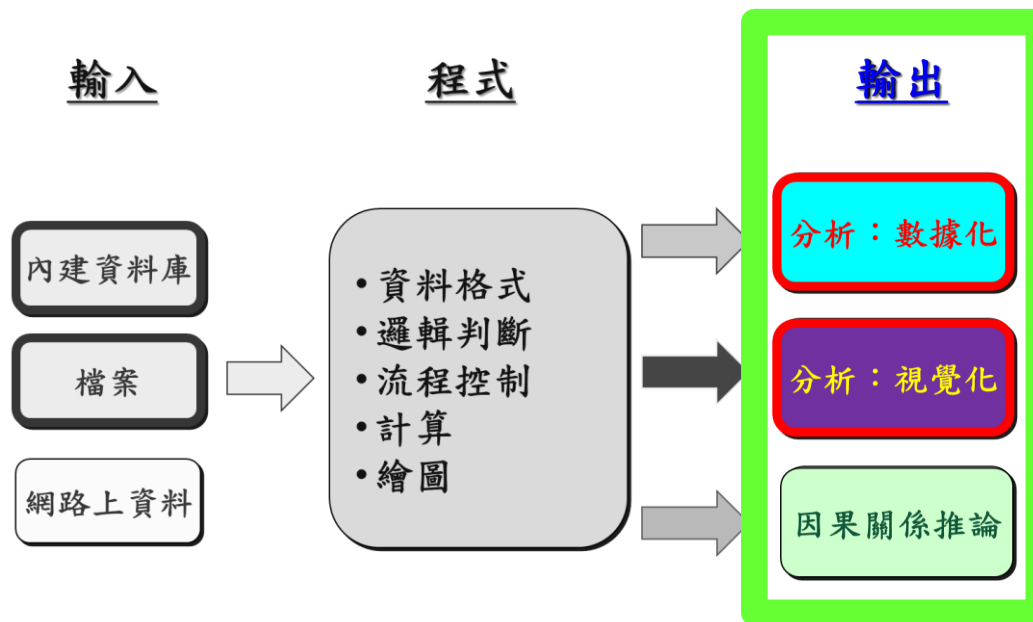
連 豐 力

臺大電機系

Feb 2018 - Jun 2018

課程主題進度

- **U01:** 課程介紹：討論主題，作業，報告，進行方式
- **U02:** 主題，案例，程式，演算法，資源
- **U03:** 設定軟體 **R** 與 **Rstudio**
- **U04:** 數據處理與繪圖指令功能
- **U05:** 資料類別與基本運算
- **U06:** 邏輯判斷與流程控制
- **U07:** 函數：計算與排序
- **U08:** 多維度資料格式
- **U09:** 檔案資料輸入與輸出
- **U10:** 繪圖功能與文字
- **U11:** 多重繪圖與顏色
- **U12:** 資料間的相關性
- **U13:** 探索性資料分析
- **U14:** 資料連結分析
- **U15:** 影像與動畫



啤酒與尿布的關係

- 1990年代，美國零售龍頭業者沃爾瑪 (Wal-Mart) 的資訊工程師，在分析結帳數據時發現，每到**星期五晚上**，**尿布**和**啤酒**的銷售量有正向關聯性。
- 透過調查才知道，原來，美國的**婦女通常在家照顧孩子**，所以她們經常會囑咐**丈夫**在**下班回家**的路上為孩子買尿布，而丈夫在買尿布的同時又會**順手購買**自己愛喝的啤酒。
- 而且，就時間上來看，特別是在週五晚上，父親常常幫家裡到超市買尿布，順便為**週末球賽**會購買啤酒回家。
- 後來沃爾瑪採取合購策略，固定在**每週五**，**啤酒**和**尿布**擺設放在同一區域，意外讓這兩項產品的銷售量**提升30%**。

大綱

- 基本元素：連結規則，支持度，可靠度，提升度
- 軟體套件，核心函數，資料集
- 對產生規則進行強度控制，
 - 透過支持度，可靠度共同控制
 - 主要透過支持度控制
 - 主要透過可靠度控制
 - 主要透過提升度控制
- 改變輸出結果形式
- 連結規則的視覺化

連結規則

基本元素：

支持度

可靠度

提升度

連結規則

- 連結規則 (Association Rule) :
- 一般記為： $X \rightarrow Y$ 的形式，用於表示資料內隱含的**連結性**。
- X ：先決條件， Y ：對應連結結果，
- 例如：連結規則：尿布 \rightarrow 啤酒，
表示：購買了**尿布**的消費者常常也會購買**啤酒**，
即是這兩個購買行為之間具有一定**連結性**。
- 至於連結性的強度，會用：**支持度**，**可靠度**，**提升度**，
等三個核心概念，來控制與評價。
- 以下，以一個數據來說明這三個概念：
10000 個消費者，1000 個購買尿布，2000 個購買啤酒，500 個購買麵包，
800 個同時購買尿布與啤酒，100 個同時購買尿布與麵包。

連結規則 - 支持度

- 支持度 (Support) :
- 指的是： $\{X, Y\}$ 出現的可能性，即同時包含 X 與 Y 的機率：

$$\text{Support}(X \rightarrow Y) = P(X, Y)$$

- 用以衡量所有連結規則在『量』上的多少。
- 利用最小支持度設定值 (minsup, minimum support) 來剔除那些較低出現率的無意義規則，而保留下出現較為頻繁資料所隱含的規則：

$$\text{Support}(\{X, Y\}) \geq \text{minsup}$$

- 例如：minsup = 5%， $\{\text{尿布}, \text{啤酒}\}$ 的支持度 = $800/10000 = 8\%$ ， $\{\text{尿布}, \text{麵包}\}$ 的支持度 = $100/10000 = 1\%$ ，
- 所以， $\{\text{尿布}, \text{啤酒}\}$ 滿足了基本的數量要求，成為頻繁的集合，則： $\text{尿布} \rightarrow \text{啤酒}$ 與 $\text{啤酒} \rightarrow \text{尿布}$ ，兩個規則被保留。

連結規則 - 可靠度

- 可靠度 (Confidence) :
- 表示在連結規則的先決條件 X 發生的條件下，連結結果 Y 發生的機率：

$$\text{Confidence} (X \rightarrow Y) = P(Y | X) = P(X, Y) / P(X)$$

- 用以衡量所有連結規則在『質』上的可用性。
- 利用最小可靠度的設定值 (mincon, minimum confidence) 來實現一些篩選，滿足：

$$\text{Confidence} (X \rightarrow Y) \geq \text{mincon}$$

- 例如：mincon = 70%，
尿布 \rightarrow 啤酒 的可靠度 = $800/1000 = 80\%$ ，
啤酒 \rightarrow 尿布的可靠度 = $800/2000 = 40\%$ ，
- 所以，尿布 \rightarrow 啤酒 滿足要求，被篩選出來的一個強連結規則。

連結規則 - 提升度

■ 提升度 (Lift) :

- 表示在含有 X 的條件下，同時含有 Y 的可能性，與沒有這個條件下，含有 Y 的可能性之比值。

$$\text{Lift}(X \rightarrow Y) = P(Y | X) / P(Y) = \text{Confidence}(X \rightarrow Y) / P(Y)$$

- 用以衡量所有連結規則在『質』上的可用性，與可靠度為互補指標。
- 例如：1000 個消費者，500 人購買茶葉，其中有 450 人同時購買咖啡。另外，50 人沒有，
- 由於，茶葉 \rightarrow 咖啡 的可靠度：450/500 = 90%，相當高，
- 但是：如果沒有購買茶葉的 500 人，其中同樣也有 450 人也同時購買咖啡，其可靠度也是：90%。
- 由此看來：是否購買咖啡，與有沒有購買茶葉並沒有連結，兩者是獨立的，其提升度為：90% / [(450 + 450) / 1000] = 1 (相互獨立)

連結規則

- 選出滿足支持度最小設定值的所有集合，即為：頻繁集合：
 - 一般設定值為：5% ~ 10%。

- 從頻繁集合中找出滿足最小可靠度的所有規則：
 - 通常可靠度的設定值為：70% ~ 90%。

軟體套件 核心函數 資料集

軟體套件 與 核心函數

- 專用於連結分析的軟體套件：arules 與 arulesViz
- Apriori 和 Eclat：兩個快速採擷頻繁集合與連結規則演算法的實現函數。
- `install.packages("arules")` # 安裝 arules 軟體套件
- `library(arules)` # 載入 arules 軟體套件

- `apriori(data, parameter = NULL, appearance = NULL, control = NULL)`
- `eclat(data, parameter = NULL, control = NULL)`

- **parameter:** support = 0.1, confidence = 0.8, maxlen = 10, minlen = 1, target = "rules" / "frequent itemsets"
- **appearance:** X (lhs = "beer"), Y (rhs = "milk")
- **control:** sort = 1 (昇), sort = -1 (降)

- library(arules) # 載入 arules 軟體套件
- data("Groceries") # 取得 Groceries 資料集
- summary(Groceries)

```
> summary( Groceries )

transactions as itemMatrix in sparse format with
9835 rows (elements/itemsets/transactions) and
169 columns (items) and a density of 0.02609146

most frequent items:
      whole milk other vegetables      rolls/buns      soda      yogurt      (Other)
           2513           1903           1809           1715           1372           34055

element (itemset/transaction) length distribution:
sizes
  1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18  19  20
2159 1643 1299 1005  855  645  545  438  350  246  182  117  78  77  55  46  29  14  14  9
 21  22  23  24  26  27  28  29  32
 11  4  6  1  1  1  1  3  1

  Min. 1st Qu.  Median    Mean 3rd Qu.  Max.
 1.000  2.000  3.000  4.409  6.000 32.000

includes extended item information - examples:
      labels level2      level1
1 frankfurter sausage meat and sausage
2   sausage sausage meat and sausage
3  liver loaf sausage meat and sausage
```

- inspect(Groceries[1:10])

```
> inspect( Groceries[ 1:10 ] )
```

```
items
```

```
[1] {citrus fruit,semi-finished bread,margarine,ready soups}  
[2] {tropical fruit,yogurt,coffee}  
[3] {whole milk}  
[4] {pip fruit,yogurt,cream cheese ,meat spreads}  
[5] {other vegetables,whole milk,condensed milk,long life bakery product}  
[6] {whole milk,butter,yogurt,rice,abrasive cleaner}  
[7] {rolls/buns}  
[8] {other vegetables,UHT-milk,rolls/buns,bottled beer,liquor (appetizer)}  
[9] {pot plants}  
[10] {whole milk,cereals} 1
```

- rules0 <- apriori(Groceries, parameter = list(support = 0.001, confidence = 0.5))

Apriori

Parameter specification:

confidence	minval	smax	arem	aval	original	support	maxtime	support	minlen	maxlen	target	ext
0.5	0.1	1	none	FALSE		TRUE	5	0.001	1	10	rules	FALSE

Algorithmic control:

filter	tree	heap	memopt	load	sort	verbose
0.1	TRUE	TRUE	FALSE	TRUE	2	TRUE

Absolute minimum support count: 9

```
set item appearances ...[0 item(s)] done [0.00s].
set transactions ...[169 item(s), 9835 transaction(s)] done [0.00s].
sorting and recoding items ... [157 item(s)] done [0.00s].
creating transaction tree ... done [0.00s].
checking subsets of size 1 2 3 4 5 6 done [0.01s].
writing ... [5668 rule(s)] done [0.00s].
creating S4 object ... done [0.00s].
```

- rules0
- inspect(rules0[1:10])

```
> rules0
```

```
set of 5668 rules
```

```
> inspect( rules0[ 1:10 ] )
```

	lhs	rhs	support	confidence	lift
[1]	{honey}	=> {whole milk}	0.001118454	0.7333333	2.870009
[2]	{tidbits}	=> {rolls/buns}	0.001220132	0.5217391	2.836542
[3]	{cocoa drinks}	=> {whole milk}	0.001321810	0.5909091	2.312611
[4]	{pudding powder}	=> {whole milk}	0.001321810	0.5652174	2.212062
[5]	{cooking chocolate}	=> {whole milk}	0.001321810	0.5200000	2.035097
[6]	{cereals}	=> {whole milk}	0.003660397	0.6428571	2.515917
[7]	{jam}	=> {whole milk}	0.002948653	0.5471698	2.141431
[8]	{specialty cheese}	=> {other vegetables}	0.004270463	0.5000000	2.584078
[9]	{rice}	=> {other vegetables}	0.003965430	0.5200000	2.687441
[10]	{rice}	=> {whole milk}	0.004677173	0.6133333	2.400371

對規則的控制

透過支持度，可靠度共同控制

- **support:** 0.001 -> 0.005
- `rules1 <- apriori(Groceries, parameter = list(support = 0.005, confidence = 0.5))`
- `rules1`
- **confidence:** 0.5 -> 0.6
- `rules2 <- apriori(Groceries, parameter = list(support = 0.005, confidence = 0.6))`
- `rules2`
- **confidence:** 0.6 -> 0.64
- `rules3 <- apriori(Groceries, parameter = list(support = 0.005, confidence = 0.64))`
- `rules3`

```
> rules1  
set of 120 rules
```

```
> rules2  
set of 22 rules
```

```
> rules3  
set of 4 rules
```

透過支持度，可靠度共同控制

- **Support** = 0.005, **confidence** = 0.64
- `rules3 <- apriori(Groceries, parameter = list(support = 0.005, confidence = 0.64))`
- `rules3`
- `inspect(rules3)`

```
> rules3
set of 4 rules
```

```
> inspect( rules3 )
```

lhs	rhs	support	confidence	lift
[1] {butter,whipped/sour cream}	=> {whole milk}	0.006710727	0.6600000	2.583008
[2] {pip fruit,whipped/sour cream}	=> {whole milk}	0.005998983	0.6483516	2.537421
[3] {pip fruit,root vegetables,other vegetables}	=> {whole milk}	0.005490595	0.6750000	2.641713
[4] {tropical fruit,root vegetables,yogurt}	=> {whole milk}	0.005693950	0.7000000	2.739554

主要透過支持度控制

- `rules.sorted_sup <- sort(rules0, by = "support")`
- `inspect(rules.sorted_sup[1:5])`

```
> inspect( rules.sorted_sup[ 1:5 ] )
```

	lhs	rhs	support	confidence	lift
[1]	{other vegetables,yogurt}	=> {whole milk}	0.02226741	0.5128806	2.007235
[2]	{tropical fruit,yogurt}	=> {whole milk}	0.01514997	0.5173611	2.024770
[3]	{other vegetables,whipped/sour cream}	=> {whole milk}	0.01464159	0.5070423	1.984385
[4]	{root vegetables,yogurt}	=> {whole milk}	0.01453991	0.5629921	2.203354
[5]	{pip fruit,other vegetables}	=> {whole milk}	0.01352313	0.5175097	2.025351

主要透過可靠度控制

- `rules.sorted_con <- sort(rules0, by = "confidence")`
- `inspect(rules.sorted_con[1:5])`

```
> inspect( rules.sorted_con[ 1:5 ] )
```

	lhs	rhs	support	confidence	lift
[1]	{rice,sugar}	=> {whole milk}	0.001220132	1	3.913649
[2]	{canned fish,hygiene articles}	=> {whole milk}	0.001118454	1	3.913649
[3]	{root vegetables,butter,rice}	=> {whole milk}	0.001016777	1	3.913649
[4]	{root vegetables,whipped/sour cream,flour}	=> {whole milk}	0.001728521	1	3.913649
[5]	{butter,soft cheese,domestic eggs}	=> {whole milk}	0.001016777	1	3.913649

主要透過提升度控制

- `rules.sorted_lift <- sort(rules0, by = "lift")`
- `inspect(rules.sorted_lift[1:5])`

```
> inspect( rules.sorted_lift[ 1:5 ] )
```

	lhs	rhs	support	confidence	lift
[1]	{Instant food products,soda}	=> {hamburger meat}	0.001220132	0.6315789	18.99565
[2]	{soda,popcorn}	=> {salty snack}	0.001220132	0.6315789	16.69779
[3]	{flour,baking powder}	=> {sugar}	0.001016777	0.5555556	16.40807
[4]	{ham,processed cheese}	=> {white bread}	0.001931876	0.6333333	15.04549
[5]	{whole milk,Instant food products}	=> {hamburger meat}	0.001525165	0.5000000	15.03823

主要透過三度的控制

```
> inspect( rules.sorted_sup[ 1:5 ] )
```

lhs	rhs	support	confidence	lift
[1] {other vegetables,yogurt}	=> {whole milk}	0.02226741	0.5128806	2.007235
[2] {tropical fruit,yogurt}	=> {whole milk}	0.01514997	0.5173611	2.024770
[3] {other vegetables,whipped/sour cream}	=> {whole milk}	0.01464159	0.5070423	1.984385
[4] {root vegetables,yogurt}	=> {whole milk}	0.01453991	0.5629921	2.203354
[5] {pip fruit,other vegetables}	=> {whole milk}	0.01352313	0.5175097	2.025351

```
> inspect( rules.sorted_con[ 1:5 ] )
```

lhs	rhs	support	confidence	lift
[1] {rice,sugar}	=> {whole milk}	0.001220132	1	3.913649
[2] {canned fish,hygiene articles}	=> {whole milk}	0.001118454	1	3.913649
[3] {root vegetables,butter,rice}	=> {whole milk}	0.001016777	1	3.913649
[4] {root vegetables,whipped/sour cream,flour}	=> {whole milk}	0.001728521	1	3.913649
[5] {butter,soft cheese,domestic eggs}	=> {whole milk}	0.001016777	1	3.913649

```
> inspect( rules.sorted_lift[ 1:5 ] )
```

lhs	rhs	support	confidence	lift
[1] {Instant food products,soda}	=> {hamburger meat}	0.001220132	0.6315789	18.99565
[2] {soda,popcorn}	=> {salty snack}	0.001220132	0.6315789	16.69779
[3] {flour,baking powder}	=> {sugar}	0.001016777	0.5555556	16.40807
[4] {ham,processed cheese}	=> {white bread}	0.001931876	0.6333333	15.04549
[5] {whole milk,Instant food products}	=> {hamburger meat}	0.001525165	0.5000000	15.03823

一個例子

- 想要瞭解芥末 (mustard) 的連結規則？
- `rules4 <- apriori(Groceries, parameter = list(maxlen = 2, support = 0.001, confidence = 0.1), appearance = list(rhs = "mustard", default = "lhs"))`
- `rules4`
- `inspect(rules4)`

```
> inspect( rules4 )
```

lhs	rhs	support	confidence	lift
-----	-----	---------	------------	------

[1] {mayonnaise}	=> {mustard}	0.001423488	0.1555556	12.96516
------------------	--------------	-------------	-----------	----------

改變輸出結果形式

改變輸出結果形式

- 想要知道銷售量最高的商品？
- `itemsets_apr <- apriori(Groceries, parameter = list(support = 0.001, target = "frequent itemsets"), control = list(sort = -1))`
- `itemsets_apr`
- `inspect(itemsets_apr[1:5])`

```
> itemsets_apr  
  
set of 13492 itemsets  
  
> inspect( itemsets_apr[ 1:5 ] )  
  
      items                support  
[1] {whole milk}           0.2555160  
[2] {other vegetables}    0.1934926  
[3] {rolls/buns}          0.1839349  
[4] {soda}                 0.1743772  
[5] {yogurt}               0.1395018
```

改變輸出結果形式

- 想要知道**網綁銷售策略**在哪些商品中作用最顯著？
- `itemsets_ecl <- eclat(Groceries, parameter = list(minlen = 1, maxlen = 3, support = 0.001, target = "frequent itemsets"), control = list(sort = -1))`
- `itemsets_ecl`
- `inspect(itemsets_ecl[1:5])`

```
> itemsets_ecl
set of 9969 itemsets

> inspect( itemsets_ecl[ 1:5 ] )

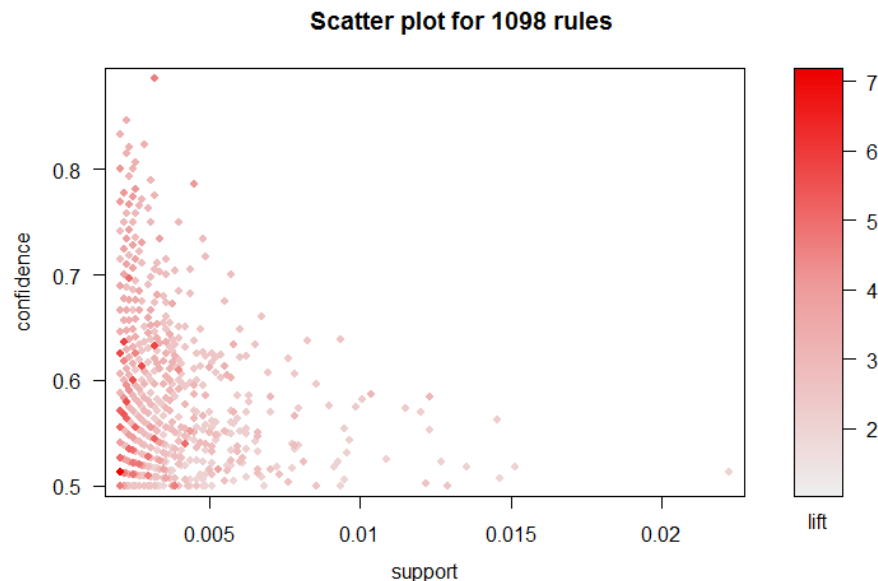
      items                                support
[1] {whole milk,honey}                    0.001118454
[2] {whole milk,cocoa drinks}             0.001321810
[3] {whole milk,pudding powder}           0.001321810
[4] {tidbits,rolls/buns}                  0.001220132
[5] {tidbits,soda}                        0.001016777
```

連結規則的視覺化

連結規則的視覺化

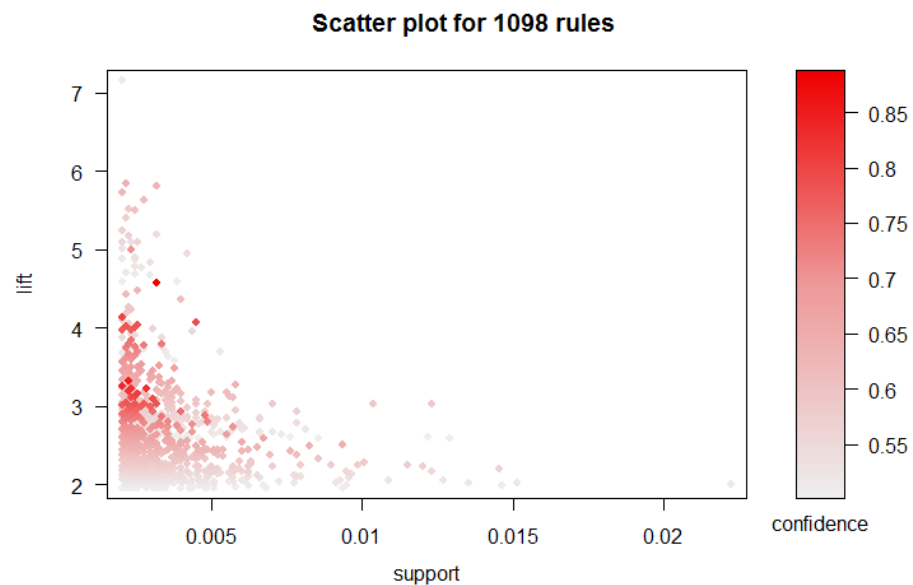
- 視覺化軟體套件：`arulesViz`
- `install.packages("arulesViz")` # 安裝 `arulesViz` 軟體套件
- `library(arulesViz)` # 載入 `arulesViz` 軟體套件

- `rules5 <- apriori(Groceries, parameter = list(support = 0.002, confidence = 0.5))`
- `rules5`
- `plot(rules5)`
- # 散點圖：支持度 vs 可靠度



連結規則的視覺化

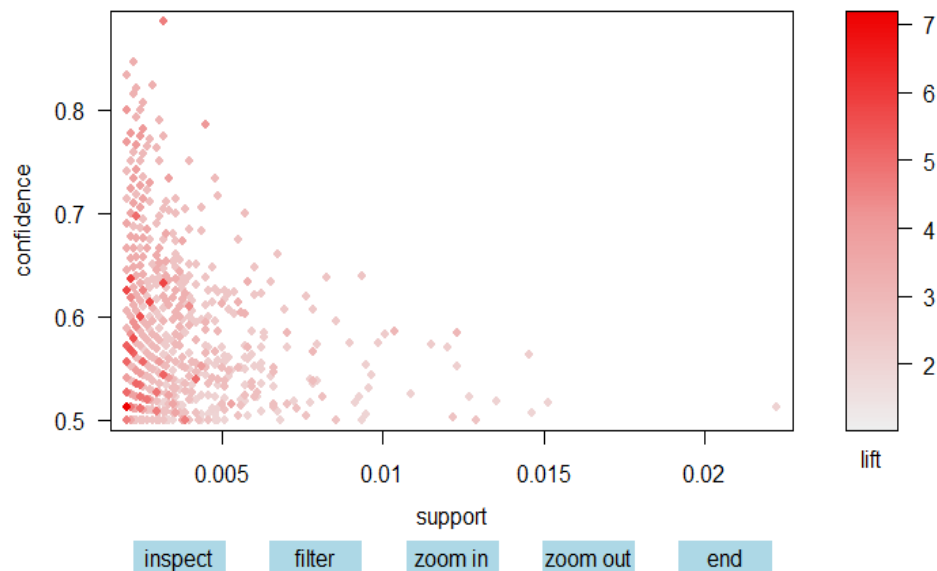
- `plot(rules5, measure = c("support", "lift"), shading = "confidence")`
- # 散點圖：支持度 VS 提升度



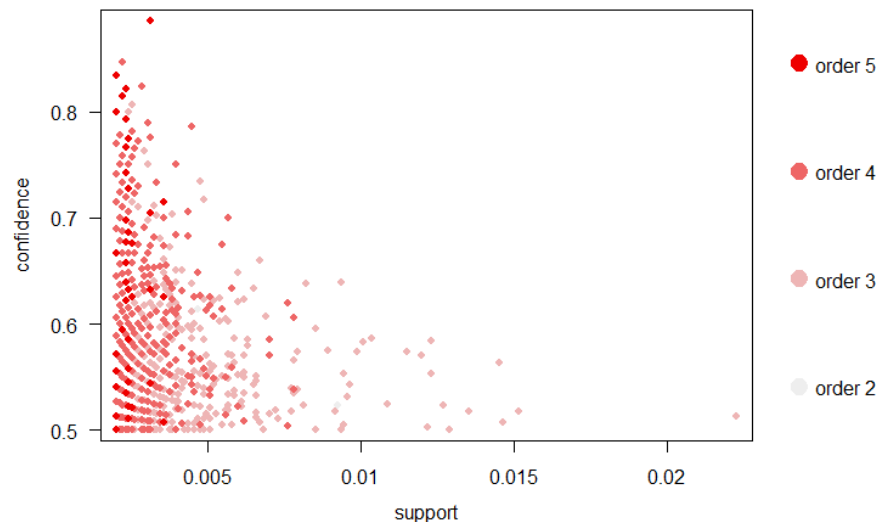
連結規則的視覺化

- `plot(rules5, interactive = T)` # 互動散點圖
- `plot(rules5, control = list(main = "Two key plot"), shading = "order")`
- # Two-key 散點圖 (點的颜色越深, 商品的種類越多)

Scatter plot for 1098 rules

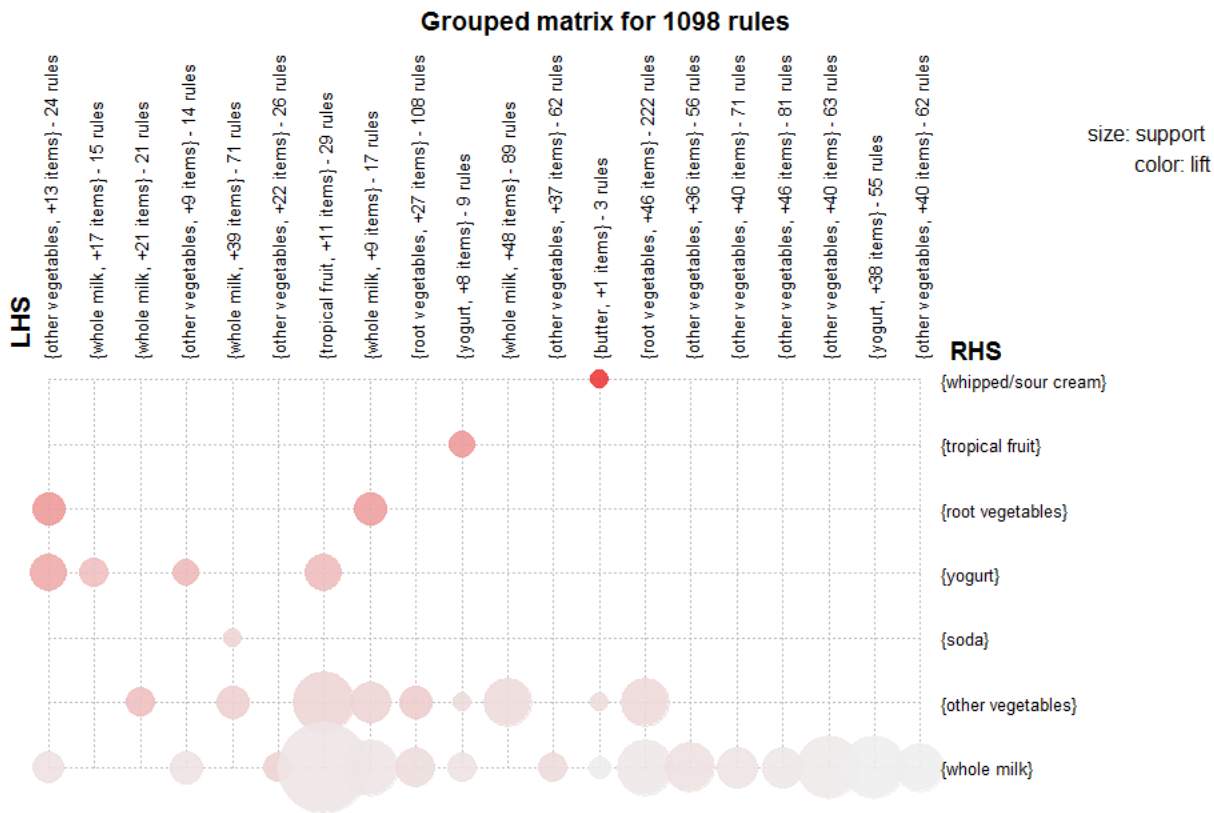


Two key plot



連結規則的視覺化

- `plot(rules5, method = "group")`
- # 群組圖：大小 (size): support, 顏色 (color): lift



連結規則的視覺化

- `plot(rules5[1:50], method = "matrix" , measure = "lift")`
- `plot(rules5[1:50], method = "matrix3D" , measure = "lift")`
- `plot(rules5[1:50], method = "paracoord")`

