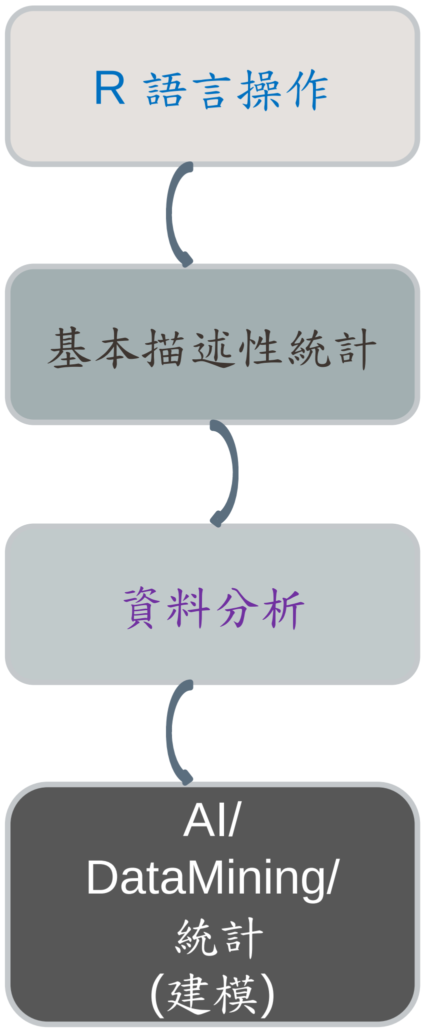
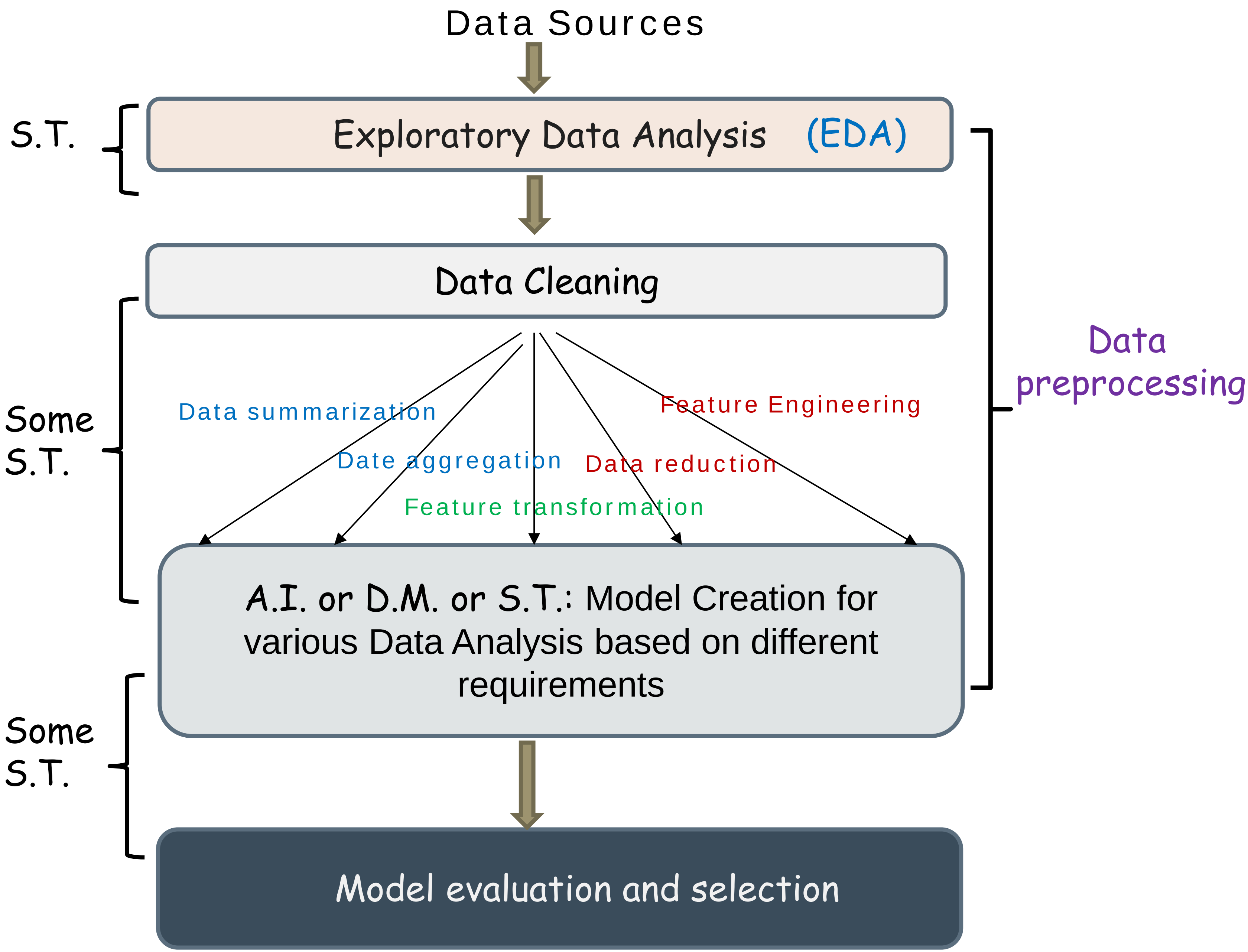
ASSOCIATION RULES

Contents

- Basic Description
- Association rule
- More Examples
- Implementation in R
- Case study
- Other Things





Data Sources
S.T.
Exploratory Data Analysis (EDA)
Data Cleaning
Some S.T.
Data summarization
Date aggregation
Feature transformation
Data reduction
Feature Engineering
Data preprocessing
A.I. or D.M. or S.T.: Model Creation for various Data Analysis based on different requirements
Some S.T.
Model evaluation and selection

Basic Description

Apriori演算法

- Apriori 演算法是一種最有影響力的探勘關聯規則的頻繁項集演算法，使用一種稱作逐層搜尋的方法，k- 項集用於探索（k+1）- 項集。
  - 首先，找出頻繁 1- 項集的集合。該集合記作L1。
  - L1 用於找頻繁2- 項集的集合 L2
  - 而L2 用於找L3，如此下去，直到不能找到 Lk- 項集。

每找一個 Lk 需要一次資料庫掃描。

關聯分析

- 關聯分析是一種在大規模資料集中尋找相互關係的任務。這些關係可以有兩種形式:
  - 頻繁項集（frequent item sets）: 經常出現在一起的物品的集合。
  - 關聯規則（associational rules）: 暗示兩種物品之間可能存在很強的關係或關聯性

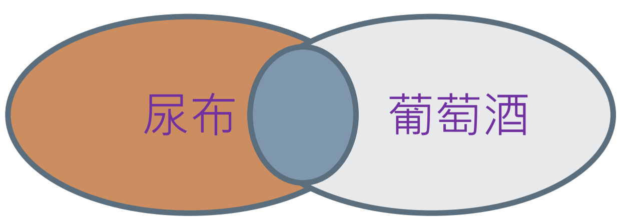
關聯分析例子

- 關聯分析 (關聯規則學習): 下面是用一個 雜貨店簡單交易清單的例子來說明這兩個概念，如下表所示:

交易號碼	交易商品
0	豆奶，萵苣
1	萵苣，尿布，葡萄酒，甜菜
2	豆奶，尿布，葡萄酒，橙汁
3	萵苣，豆奶，尿布，葡萄酒
4	萵苣，豆奶，尿布，橙汁

交易號碼	交易商品
0	豆奶，萵苣
1	萵苣，尿布，葡萄酒，甜菜
2	豆奶，尿布，葡萄酒，橙汁
3	萵苣，豆奶，尿布，葡萄酒
4	萵苣，豆奶，尿布，橙汁

- 頻繁項集: {葡萄酒, 尿布, 豆奶} 就是一個L_3頻繁項集的例子。
- 關聯規則: 尿布 -> 葡萄酒 就是一個關聯規則。這意味著如果顧客買了尿布，那麼他很可能會買葡萄酒。
- 支援度(support): 資料集中包含該項集(同時出現)的記錄所佔的比例。例如上圖中，{豆奶} 的支援度為 4/5。{豆奶, 尿布} 的支援度為 3/5。
- 可信度(confidence): 針對一條諸如 {尿布} -> {葡萄酒} 這樣具體的關聯規則來定義的。這條規則的 可信度 被定義為 support({尿布, 葡萄酒})/support({尿布})，support({尿布, 葡萄酒}) = 3/5，support({尿布}) = 4/5，所以 {尿布} -> {葡萄酒} 的confidence = 3/5 / 4/5 = 3/4 = 0.75。



交易號碼	交易商品
0	豆奶，萵苣
1	萵苣，尿布，葡萄酒，甜菜
2	豆奶，尿布，葡萄酒，橙汁
3	萵苣，豆奶，尿布，葡萄酒
4	萵苣，豆奶，尿布，橙汁

- Support 和 confidence 是用來量化關聯分析 是否成功的一個方法。
  假設想找到support 大於 0.8 的所有項集應該如何去做呢？
  一個辦法是生成一個物品所有可能組合的清單，然後對每一種組合統計它出現的頻繁程度，但是當物品成千上萬時，上述做法就非常非常慢了。
  我們需要詳細分析下這種情況並討論下 Apriori 原理，該原理會減少關聯規則學習時所需的計算量。

一般定義

- k項集
  如果事件A中包含k個元素，那麼稱這個事件A為k項集，並且事件A滿足最小 support度門檻值的事件稱為頻繁k項集。
- 由頻繁項集產生強關聯規則
  - K維資料項集LK是頻繁項集的必要條件,是它所有K-1維子項集也為頻繁項集，記為LK-1
  - 如果K維資料項集LK的任意一個K-1維子集Lk-1，不是頻繁項集，則K維資料項集LK本身也不是最大資料項集。
  - Lk是K維頻繁項集，如果所有K-1維頻繁項集合Lk-1中包含LK的K-1維子項集的個數小於K，則Lk不可能是K維最大頻繁資料項集。
  - 同時滿足最小support門檻值和最小 confidence門檻值的規則稱為**強規則**。

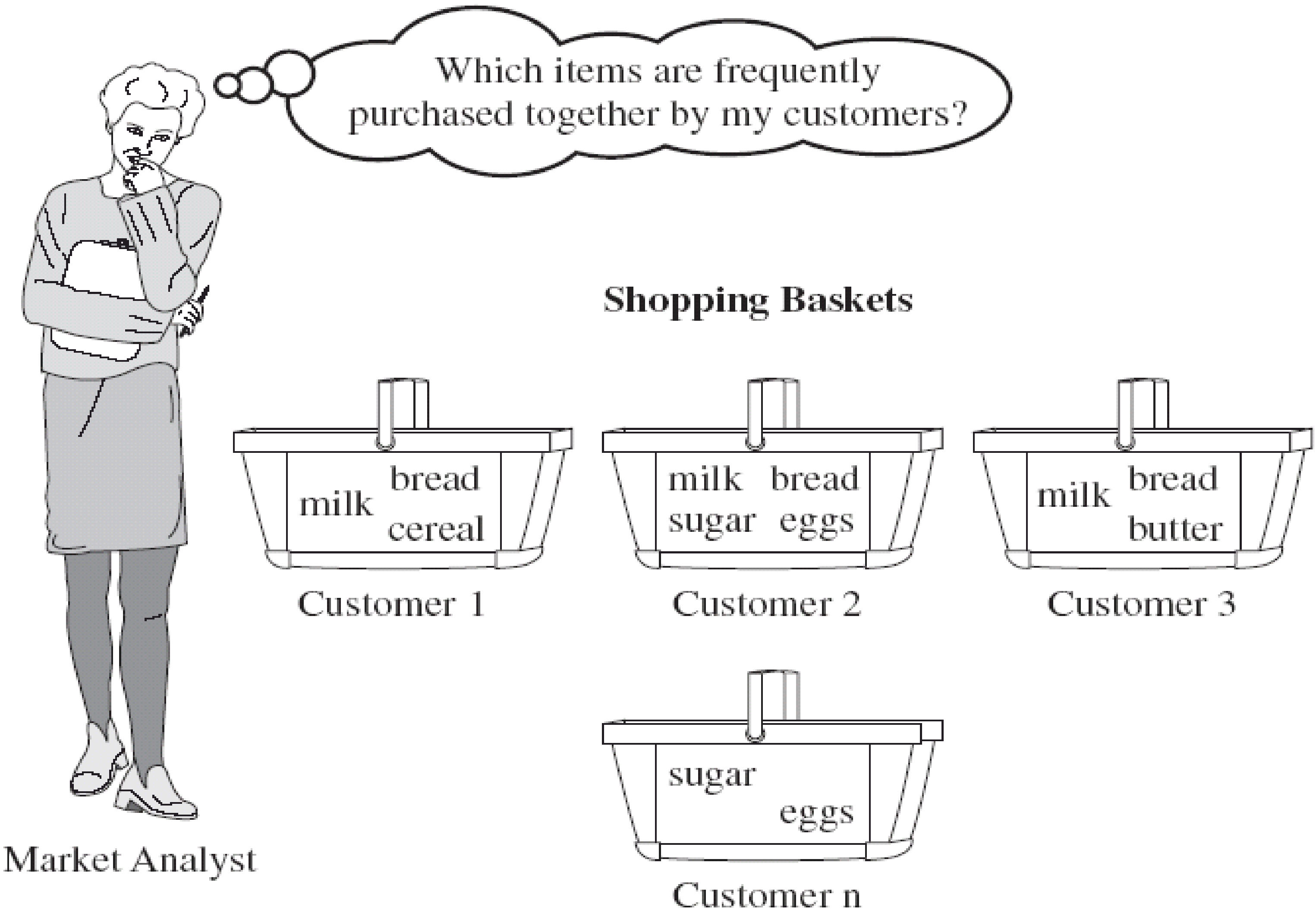
關聯法則 (Association rule)

from 1. 陳士杰 老師 2. wiki

關聯法則

- 是一種在大型資料庫中發現變數之間的有趣性關係的方法。它的目的是利用一些有趣性的量度來辨識資料庫中發現的強規則。
- 基於強規則的概念，Rakesh Agrawal等人引入了關聯規則, 以發現由超市的POS系統記錄, 大批交易資料中產品之間的規律性。例如，從銷售資料中發現的規則 {洋蔥, 土豆}→{漢堡} 會表明如果顧客一起買洋蔥和土豆，他們也有可能買漢堡的肉。此類資訊可以作為做出促銷定價或產品置入等行銷活動決定的根據。
- 除了上面的購物籃分析中的例子以外， 關聯規則如今還被用在許多應用領域中，包括網路用法挖掘、入侵檢測、連續生產及生物資訊學中。與序列挖掘相比，關聯規則通常不考慮在事務中、或事務間的專案的順序(註: 不一定)。

關聯法則



Which items are frequently purchased together by my customers?
Shopping Baskets
milk bread cereal
milk bread sugar eggs
milk bread butter
sugar eggs
Customer 1
Customer 2
Customer 3
Customer n
Market Analyst

關聯法則

- 如果問題的全域是商店中所有被購買商品的集合，則對每種商品都可以用一個「布林量」來表示該商品是否被顧客購買，則每個購物籃都可以用一個布林向量表示；而透過分析布林向量則可以得到商品被頻繁關聯或被同時購買的模式，這些模式就可以用關聯規則表示。

（0001001100，但這種方法失去了什麼訊息？）

關聯法則探勘?

- Association rule mining:
  - Find frequent patterns, associations, correlations, or causal structures (因果結構)among sets of items or objects in datasets.
- Examples.
  - Rule form: "Body → Head [support, confidence]".
    - buys(x, "diapers") → buys(x, "beers") [0.5%, 60%]
    - virus(x, "A") ^ browser(x, "IE") → computer_crash(x, "level A") [1%, 75%]

Items

- Itemset: 所有的項目集合 I={A,B,C,D,E,F}
- 每個交易T由交易識別符號TID標識，它的購買項目集合 (purched itemset)
  - ex：TID(2000)={A,B,C}
- Assume D是 dataset (ie. 資料集)

D

TID	購買的item
2000	A,B,C
1000	A,C
4000	A,D
5000	B,E,F

Used terms

- Item
  - I1, I2, I3, …
  - A, B, C, …
- Itemset
  - {I1}, {I1, I7}, {I2, I3, I5}, …
  - {A}, {A, G}, {B, C, E}, …
- 1-Itemset
  - {I1}, {I2}, {A}, …
- 2-Itemset
  - {I1, I7}, {I3, I5}, {A, G}, …
- K-itemset: k's items in a set

關聯法則表示法

- 假設$I = \{I_1, I_2, \ldots, I_m\}$是項的集合。給定一個交易資料庫$D = \{t_1, t_2, \ldots, t_n\}$，其中每個事務（Transaction）t是I的非空子集，即$t \subseteq I$，每一個交易都與一個唯一的識別元TID（Transaction ID）對應。

- TID（Transaction ID）對應。**關聯規則**是形如$X \Rightarrow Y$的蘊涵式，其中$X, Y \subseteq I$且$X \cap Y = \emptyset$，X和Y分別稱為關聯規則的**先導**(antecedent或left-hand-side, LHS)和**後繼**(consequent或right-hand-side, RHS) 。關聯規則$X \Rightarrow Y$在D中的**支援度**（support）是D中事務包含$X \cup Y$的百分比，即機率$P(X \cup Y)$；**置信度**（confidence）是包含X的事務中同時包含Y的百分比，即條件機率$P(Y|X)$。

- 如果同時滿足**最小 support 值**和**最小 confidence值**，則認為關聯規則是有趣的。這些值由分析師,或專家設定。

More Examples

評量關聯性的法則應用-Example

- Support: 支持度
- Confidence: 可靠度/ 信心度
- Lift: 提升度
- Questions:
  - #transactions: 10,000 筆
  - #purchase Diaper: 1,000 筆
  - #purchase Beer: 2,000 筆
  - #purchase Bread: 500 筆
  - #purchase Diaper and Beer: 800 筆
  - #purchase Diaper and Bread: 100 筆

- Questions:
  - #transactions: 10,000 筆
  - #purchase Diaper: 1,000 筆
  - #purchase Beer: 2,000 筆
  - #purchase Bread: 500 筆
  - #purchase Diaper and Beer: 800 筆
  - #purchase Diaper and Bread: 100 筆

Support: 支持度

- Support (X → Y) = P(X, Y)
- 指在所有 itemset {X, Y} 中, **同時**含有X 和 Y的機率
- 篩選標準 Support(X → Y) ≧ minsup
- 滿足篩選標準的 itemsets, 稱為 Frequent itemsets.
- Frequent itemsets篩選僅與 support有關
- Calculation:
  - Assume: minsup = 5%
  - Support(Diaper → Beer) = 8% ✓ p.s. 800/10,000
  - Support(Diaper → Bread) = 1% × p.s. 100/10,000
  - So, Support(Diaper → Bread) (註: {Diaper → Bread}) 相關 itemsets會被剔除. 因為低於 5%

Confidence: 可靠度

- Questions:
  - #transactions: 10,000 筆
  - #purchase Diaper: 1,000 筆
  - #purchase Beer: 2,000 筆
  - #purchase Bread: 500 筆
  - #purchase Diaper and Beer: 800 筆
  - #purchase Diaper and Bread: 100 筆

- Confidence $(X \rightarrow Y) = P(Y|X) = P(X, Y)/P(X)$
  $= P(X \wedge Y)/P(X)$
- 指在所有 itemset {X, Y} 中, 發生X下, 也關聯發生 Y的機率
- 篩選標準 Confidence$(X \rightarrow Y) \geqq$ minconf
- Calculation:
  - Assume: minconf = 70%
  - Confidence (Diaper → Beer) = P(Diaper, Beer)/ P(Diaper)
    = 800/ 1000= 80% ✓
  - Confidence (Beer → Diaper) = P(Beer, Diaper)/ P(Beer)
    = 800/ 2,000= 40% ×

Lift: 提升度

- Questions:
  - #transactions: 10,000 筆
  - #purchase Diaper: 1,000 筆
  - #purchase Beer: 2,000 筆
  - #purchase Bread: 500 筆
  - #purchase Diaper and Beer: 800 筆
  - #purchase Diaper and Bread: 100 筆

- Lift (X → Y) = P(Y| X) / P(Y)= Confidence(X→Y)/ P(Y)
  = P(X ∧ Y)/[P(X) * P(Y)]
- 指在所有 itemset {X, Y} 中, 發生X下, 也關聯發生 Y的機率. 相較於Y單獨出現的機率. 提升度可以輔助可靠度.
- 提升度通過衡量使用規則後, 的提升效果來判斷規則是否可用。簡單來說就是使用**關聯規則後**的商品, 在購物車中出現的次數是否高於商品單獨出現在購物車中的頻率。
- Calculation:
  - #customers: 1,000/ #purchase tea: 500, 其中#purchase coffee: 450/ 500
  - Confidence (tea → coffee)=450/ 500=90%. Say: like tea, then like coffee. **動機**: However, not sure for those people don't like tea, then like coffee? Seems: no links (p.s. independent) between tea and coffee.
  - Lift(tea → coffee)= (450/500) / (450/1000)= 2 > 1 ← tea對coffee關聯性高
  - 當 Lift值為1, 表示 X, Y相互獨立, 無提升作用, 當值大於 1, 則表示X對Y的提升度越高, 即表明連結性越強.

演算法基本想法

- 選出滿足**支援度最小設訂值**的所有itemset, 即頻繁itemsets.
- 從頻繁 itemsets 中找出滿足最小可靠度的所有規則

Implementation in R

Package: arules & arulesViz

- arules: Provides the infrastructure for representing, manipulating and analyzing transaction data and patterns (frequent itemsets and association rules). 用於關連法則 dataset 的產生
- arulesViz: Extends package 'arules' with various visualization techniques for association rules and itemsets. The package also includes several interactive visualizations for rule exploration.

Packages for the algorithms

- Apriori: apriori() 經典的關連法則演算法. 效能低
- Eclat: eclat() 效能較 Apriori好.
- FP-Growth:效能較Apriori 與 Eclat好

The parameters in the algorithms

- data: 資料集
- parameter: 設定 support、confidence, … p.s. use support=0.1, confidence=0.8 as default values
- appearnance: 設定 lhs, rhs, …
- control: 設定 ascending, descending, …

Usage

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

Data set

```
install.packages("arules")
library ( arules )

data("Groceries")
summary(Groceries)
inspect(Groceries[1:10])
```

```
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
            2513             1903             1809
            soda           yogurt          (Other)
            1715             1372            34055

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

Data set

- Check the first 10 transactions

```
> inspect(Groceries[1:10], linebreak=FALSE)
     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}
```

Case study

rule0 setting

慢慢篩選 itemsets

設定門檻+記錄演算法執行細節

```
> #p22
> rules0=apriori(Groceries,parameter=list(support=0.001,confidence=0.5))
Apriori

Parameter specification:
 confidence minval smax arem  aval originalSupport 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 [0.02s].
writing ... [5668 rule(s)] done [0.00s].
creating S4 object  ... done [0.00s].
```

rule0 inspection

```
> rules0
set of 5668 rules
> inspect(rules0[1:10], linebreak=FALSE)
     lhs                     rhs                  support     confidence lift     count
[1]  {honey}              => {whole milk}         0.001118454 0.7333333  2.870009 11
[2]  {tidbits}            => {rolls/buns}         0.001220132 0.5217391  2.836542 12
[3]  {cocoa drinks}       => {whole milk}         0.001321810 0.5909091  2.312611 13
[4]  {pudding powder}     => {whole milk}         0.001321810 0.5652174  2.212062 13
[5]  {cooking chocolate}  => {whole milk}         0.001321810 0.5200000  2.035097 13
[6]  {cereals}            => {whole milk}         0.003660397 0.6428571  2.515917 36
[7]  {jam}                => {whole milk}         0.002948653 0.5471698  2.141431 29
[8]  {specialty cheese}   => {other vegetables}   0.004270463 0.5000000  2.584078 42
[9]  {rice}               => {other vegetables}   0.003965430 0.5200000  2.687441 39
[10] {rice}               => {whole milk}         0.004677173 0.6133333  2.400371 46
```

不斷的調整門檻

- 不斷的調整門檻,篩選 itemsets
- 可先考慮 support 與 confidence, 來篩選 itemsets
- rules0 is set up for the 1st selection

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

```
> rules0
set of 5668 rules
```

Other rules settings-way 1

- 透過support與confidence共同控制, 多次篩選

```
rules1=apriori(Groceries,parameter=list(support=0.005,confidence=0.5))
rules1
rules2=apriori(Groceries,parameter=list(support=0.005,confidence=0.6))
rules2
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
```

Other rules settings

```
> inspect(rules3, linebreak=FALSE)
    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
    count
[1] 66
[2] 59
[3] 54
[4] 56
```

Other rules settings-way 2

- 對support給予固定值(min support),來篩選
- 如下, 若依照support來選擇(in descending)

```
> rules.sorted_sup = sort ( rules0, by="support" )
> inspect ( rules.sorted_sup [1:10], linebreak=FALSE )
     lhs                                     rhs                support    confidence lift     count
[1]  {other vegetables,yogurt}            => {whole milk}       0.02226741 0.5128806  2.007235 219
[2]  {tropical fruit,yogurt}              => {whole milk}       0.01514997 0.5173611  2.024770 149
[3]  {other vegetables,whipped/sour cream} => {whole milk}      0.01464159 0.5070423  1.984385 144
[4]  {root vegetables,yogurt}             => {whole milk}       0.01453991 0.5629921  2.203354 143
[5]  {pip fruit,other vegetables}         => {whole milk}       0.01352313 0.5175097  2.025351 133
[6]  {root vegetables,yogurt}             => {other vegetables} 0.01291307 0.5000000  2.584078 127
[7]  {root vegetables,rolls/buns}         => {whole milk}       0.01270971 0.5230126  2.046888 125
[8]  {other vegetables,domestic eggs}     => {whole milk}       0.01230300 0.5525114  2.162336 121
[9]  {tropical fruit,root vegetables}     => {other vegetables} 0.01230300 0.5845411  3.020999 121
[10] {root vegetables,rolls/buns}         => {other vegetables} 0.01220132 0.5020921  2.594890 120
```

Other rules settings-way 3

- 對confidence給予固定值(min cinfidence),來篩選
- 如下, 若依照confidence來選擇(in descending)

```
> rules.sorted_con = sort ( rules0, by="confidence" )
> inspect ( rules.sorted_con [1:10], linebreak=FALSE )
     lhs                                                  rhs                support     confidence lift     count
[1]  {rice,sugar}                                      => {whole milk}       0.001220132 1          3.913649 12
[2]  {canned fish,hygiene articles}                    => {whole milk}       0.001118454 1          3.913649 11
[3]  {root vegetables,butter,rice}                     => {whole milk}       0.001016777 1          3.913649 10
[4]  {root vegetables,whipped/sour cream,flour}        => {whole milk}       0.001728521 1          3.913649 17
[5]  {butter,soft cheese,domestic eggs}                => {whole milk}       0.001016777 1          3.913649 10
[6]  {citrus fruit,root vegetables,soft cheese}        => {other vegetables} 0.001016777 1          5.168156 10
[7]  {pip fruit,butter,hygiene articles}               => {whole milk}       0.001016777 1          3.913649 10
[8]  {root vegetables,whipped/sour cream,hygiene articles} => {whole milk}   0.001016777 1          3.913649 10
[9]  {pip fruit,root vegetables,hygiene articles}      => {whole milk}       0.001016777 1          3.913649 10
[10] {cream cheese ,domestic eggs,sugar}               => {whole milk}       0.001118454 1          3.913649 11
```

Other rules settings-way 4

- 對lift控制(p.s. setup support=0.001, confidence=0.5),來篩選

```
> rules.sorted_lift = sort ( rules0, by="lift" )
> inspect ( rules.sorted_lift [1:8] )
    lhs                                                   rhs              support     confidence lift     count
[1] {Instant food products,soda}                       => {hamburger meat} 0.001220132 0.6315789  18.99565 12
[2] {soda,popcorn}                                     => {salty snack}    0.001220132 0.6315789  16.69779 12
[3] {flour,baking powder}                              => {sugar}          0.001016777 0.5555556  16.40807 10
[4] {ham,processed cheese}                             => {white bread}    0.001931876 0.6333333  15.04549 19
[5] {whole milk,Instant food products}                 => {hamburger meat} 0.001525165 0.5000000  15.03823 15
[6] {other vegetables,curd,yogurt,whipped/sour cream}  => {cream cheese }  0.001016777 0.5882353  14.83409 10
[7] {processed cheese,domestic eggs}                   => {white bread}    0.001118454 0.5238095  12.44364 11
[8] {tropical fruit,other vegetables,yogurt,white bread} => {butter}       0.001016777 0.6666667  12.03058 10
```

Other rules settings

- Conclusion: lift 是 association rule中較可靠的指標, 如上所獲得的結論.

Another thinking - promotion

- 如何**促銷**冷門的商品? 如, mustard (芥末)
- 想法:可以透過 association rule中的 **rhs="mustard"**設定, 來搜尋rhs中, 僅僅包含mustard的連結規則, 進而找到mustard的強連結商品,再將mustard與找到的商品, 作為綑綁

```
rules4=apriori(Groceries,parameter=list(maxlen=2,supp=0.001,conf=0.1),appearance=list(rhs="mustard",default="lhs"))
inspect (rules4)
```

```
    lhs              rhs       support     confidence lift     count
[1] {mayonnaise} => {mustard} 0.001423488 0.1555556  12.96516 14
```

- Conclusion: mayonnaise與 mustard是強連結商品. 可考慮一起放至於同一架上, 或是加入同一款促銷的活動 (p.s. where #maxlen = #items of lhs + #items of rhs)

Another thinking - frequent itemsets

- 如何輸出月銷量最高的商品? 或是, 被綑綁的商品中, 那些促銷後較顯著? 將target設為 "frequent itemsets" 可產生類似的輸出.
- sort = -1 表 descending

```
itemsets_apr = apriori (Groceries,
                parameter = list (supp=0.001,target = "frequent itemsets"),
                control=list(sort=-1))
```

```
> itemsets_apr
set of 13492 itemsets
> inspect(itemsets_apr[1:5])
    items              support   count
[1] {whole milk}       0.2555160 2513
[2] {other vegetables} 0.1934926 1903
[3] {rolls/buns}       0.1839349 1809
[4] {soda}             0.1743772 1715
[5] {yogurt}           0.1395018 1372
```

Another thinking - revisit frequent itemsets

- 以 eclat() 演算法, 來解決上一頁的問題.
- Frequent itemsets 篩選僅與 support有關
- 下面例子亦可於Frequent itemsets中加入 confidence設定, 但輸出結果相同

```
itemsets_ecl = eclat(Groceries,
                     parameter = list( minlen=1, maxlen=3,supp=0.001,
                     target = "frequent itemsets"),control=list(sort=-1))
```

```
> itemsets_ecl
set of 9969 itemsets
> inspect(itemsets_ecl[1:5])
    items                         support     count
[1] {whole milk,honey}            0.001118454 11
[2] {whole milk,cocoa drinks}     0.001321810 13
[3] {whole milk,pudding powder}   0.001321810 13
[4] {tidbits,rolls/buns}          0.001220132 12
[5] {tidbits,soda}                0.001016777 10
```

Visualization

- 安裝/ 引用相關套件

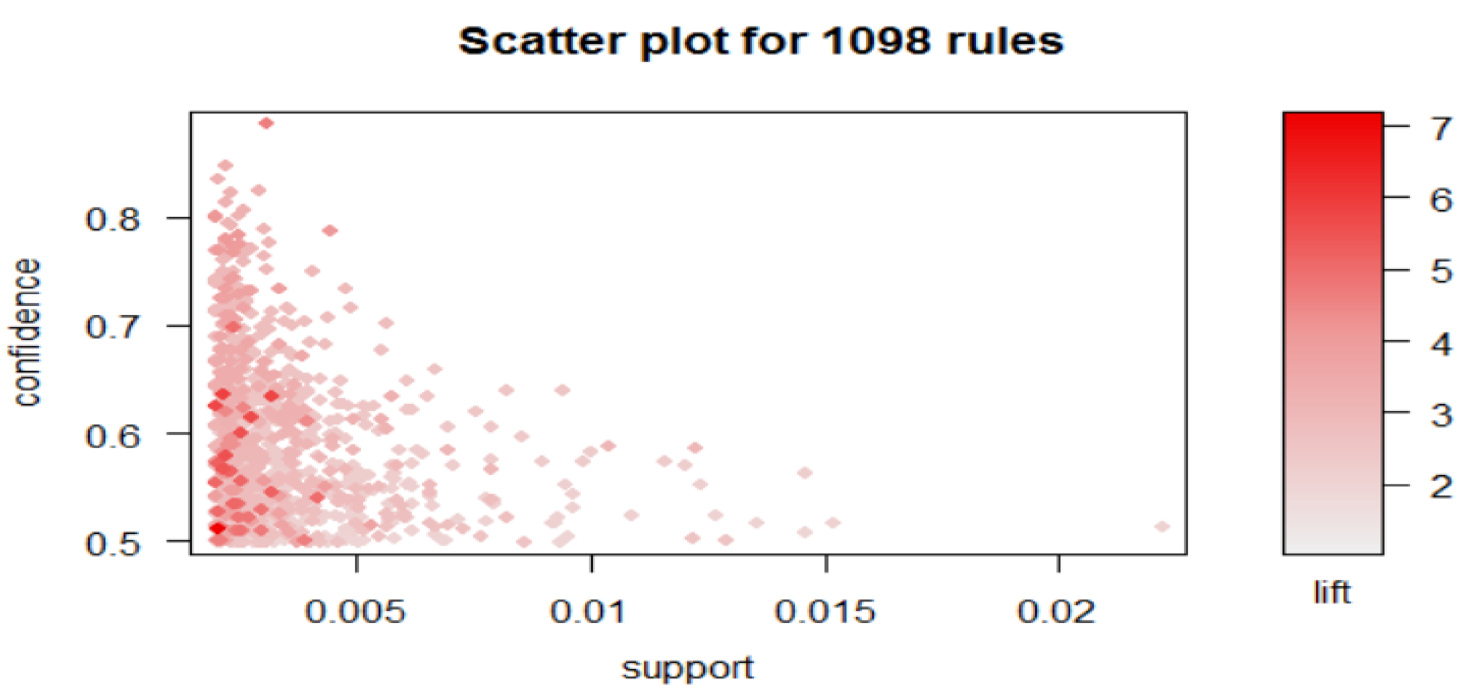
```
install.packages("digest")
library(digest)
install.packages("arulesViz")
library(arulesViz)

library ( MASS );library ( scatterplot3d );
library ( vcd );library ( grid )
library ( colorspace );library ( seriation );
library ( cluster );library ( TSP );library ( gclus )
```

Visualization

- 以下為1098條rules的 support與confidence的散佈圖
- 下圖中 lift提高, support的數值會變小

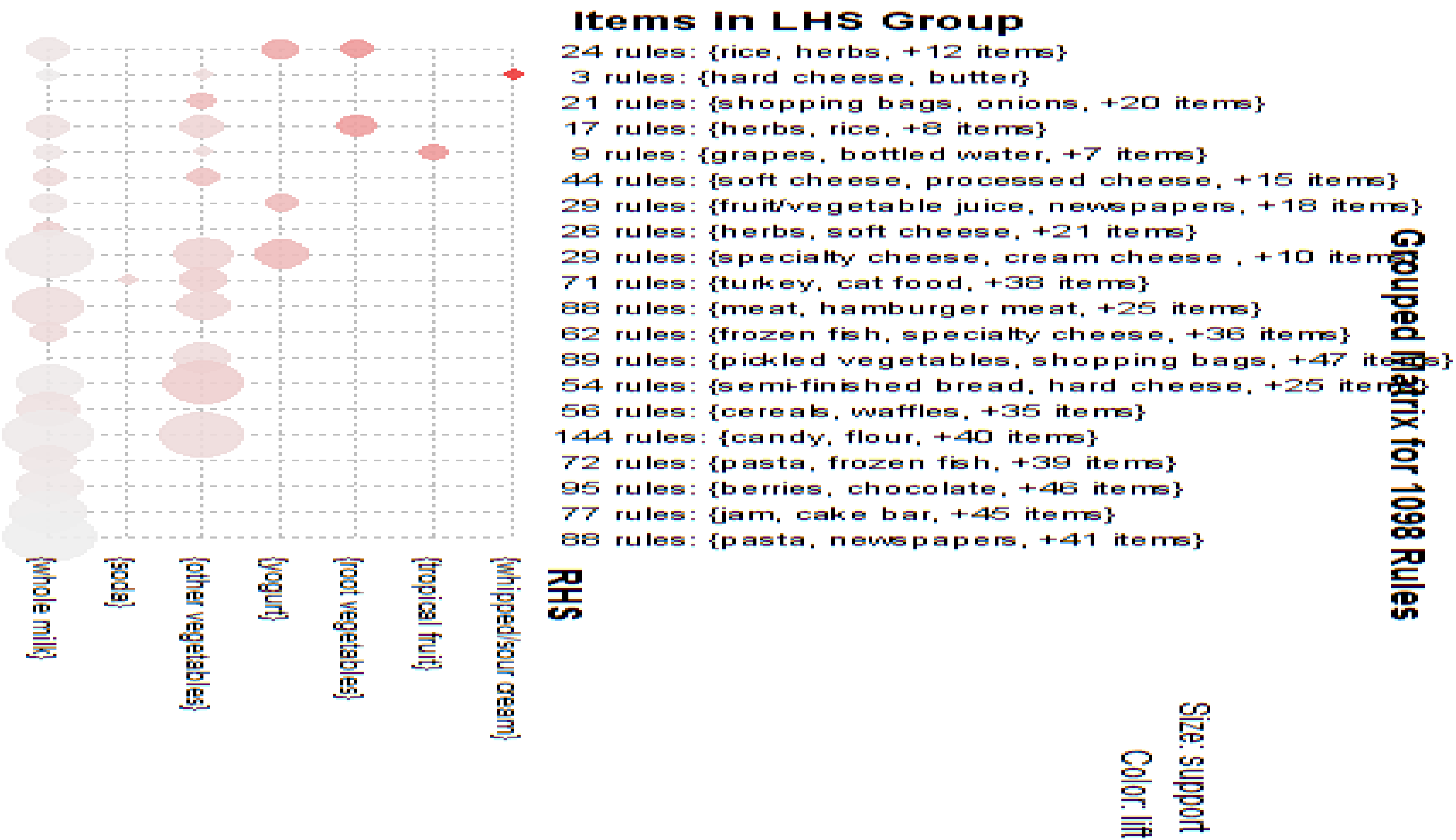
```
rules5 = apriori(Groceries,parameter=list(support=0.002,confidence=0.5))
```

```
> rules5
set of 1098 rules
> plot(rules5)
```



Visualization

```
png(file="grouped.png")
plot(rules5,method="grouped")
dev.off()
```

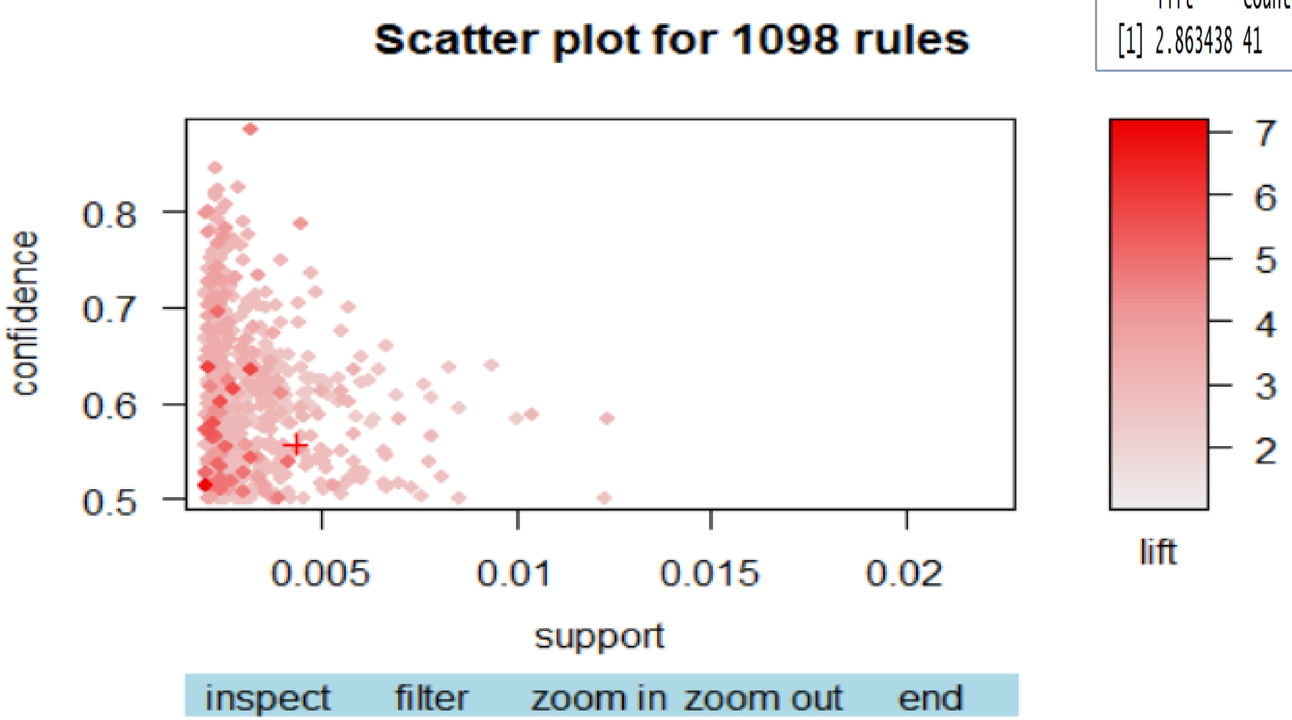


Visualization - Interactive

- 如何於散狀圖中找出規則的對應商品
- Double click 點選圖形, 或是點選兩十字中的陰影, 再點選 inspect 即可看到對應的規則

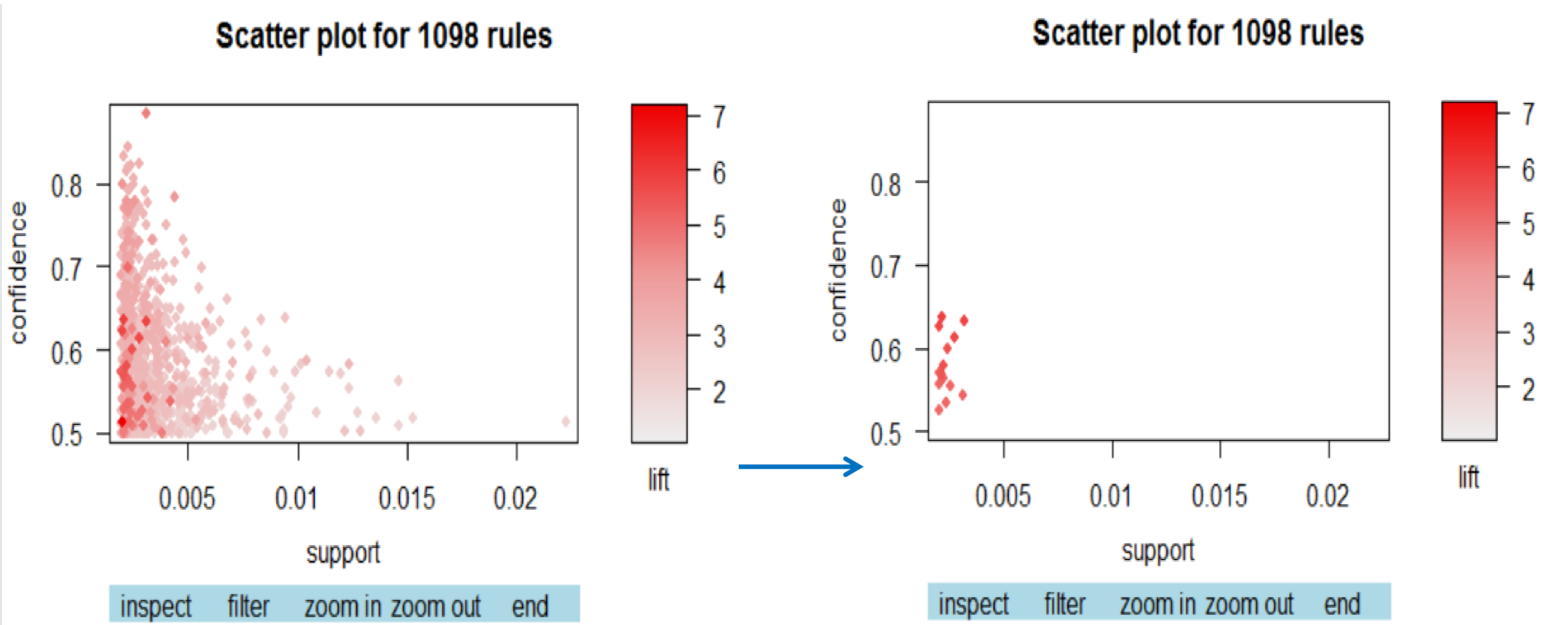
```
plot(rules5,interactive=TRUE)
```

```
Number of rules selected: 1
    lhs                            rhs                support     confidence
[1] {whipped/sour cream,pastry} => {other vegetables} 0.004168785 0.5540541
    lift     count order
[1] 2.863438 41    3
```



Visualization - Interactive

- 先點選 [filter], 再點選 lift, 篩選所要的規則



Summary

- Support (A ⇒B) = P(A ∩B)：A與B共同出現的機率，數值越大越好
  Confidence (A ⇒B) = P(B | A)：在A出現的前提下，出現B的的機率，數值越大越好
  Lift (A ⇒B) = P(B | A) / P(B)：B單獨出現比率與前項Confidence (A ⇒B)的比較，當數值大於1表示規則有效，數值越大效果越好

Other Tings

Other things you need to care

- How about training data?
- How to find the goal property for Corelation?
- How to find the goal property for Covariance?

The End