Association Rule

Hazim Fitri

Contents

ETL

```
library(arules)
```

```
## Loading required package: Matrix
```

```
##
## Attaching package: 'arules'
```

```
## The following objects are masked from 'package:base':
##
##     abbreviate, write
```

```
# see all data in arules package
data(package = 'arules')

# load data
data('Groceries')

# see data class
class(Groceries)
```

```
## [1] "transactions"
## attr(,"package")
## [1] "arules"
```

EDA

```
inspect(head(Groceries))
```

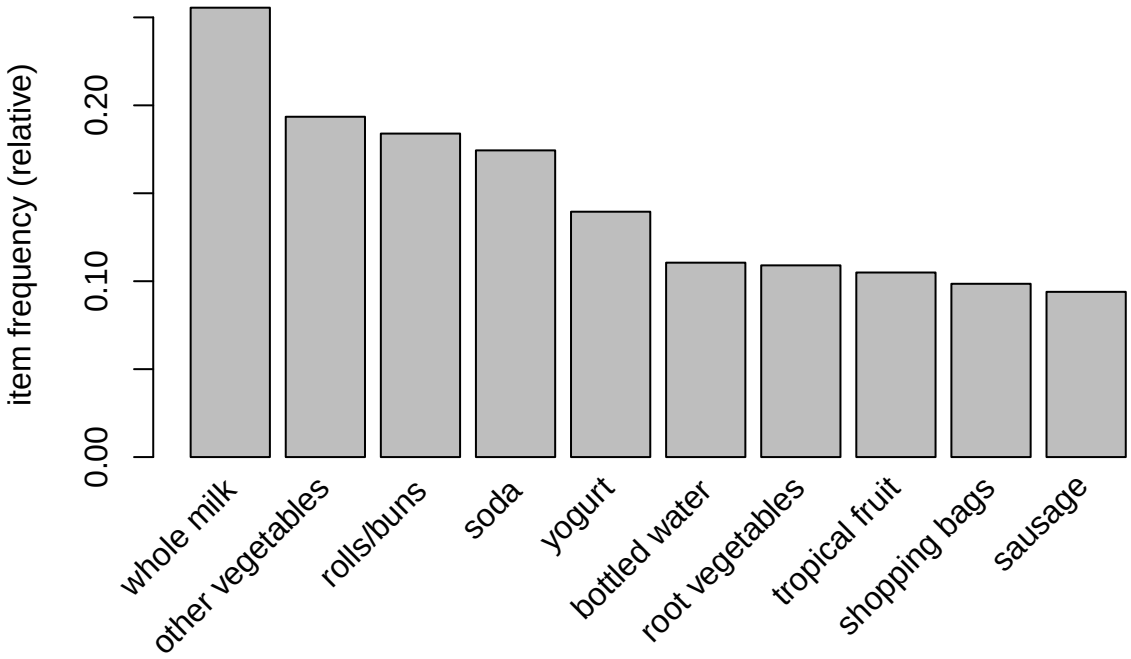
```
##     items
## [1] {citrus fruit,
##      semi-finished bread,
##      margarine,
##      ready soups}
## [2] {tropical fruit,
```

```
##      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}
```

```
data = as(Groceries, 'transactions')
data
```

```
## transactions in sparse format with
##  9835 transactions (rows) and
##  169 items (columns)
```

```
itemFrequencyPlot(Groceries, topN=10)
```



Model Training

```
rules = apriori(data, parameter = list(supp=0.05, conf=0.1))
```

```
## Apriori
```

```
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.1    0.1    1 none FALSE            TRUE       5    0.05      1
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 491
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[169 item(s), 9835 transaction(s)] done [0.00s].
## sorting and recoding items ... [28 item(s)] done [0.00s].
## creating transaction tree ... done [0.00s].
## checking subsets of size 1 2 done [0.00s].
## writing ... [14 rule(s)] done [0.00s].
## creating S4 object  ... done [0.00s].
```

```
head(rules)
```

```
## set of 6 rules
```

```
inspect(head(rules))
```

```
##     lhs    rhs                support   confidence coverage lift count
## [1] {}  => {bottled water}    0.1105236 0.1105236  1        1    1087
## [2] {}  => {tropical fruit}   0.1049314 0.1049314  1        1    1032
## [3] {}  => {root vegetables}  0.1089985 0.1089985  1        1    1072
## [4] {}  => {soda}             0.1743772 0.1743772  1        1    1715
## [5] {}  => {yogurt}           0.1395018 0.1395018  1        1    1372
## [6] {}  => {rolls/buns}       0.1839349 0.1839349  1        1    1809
```

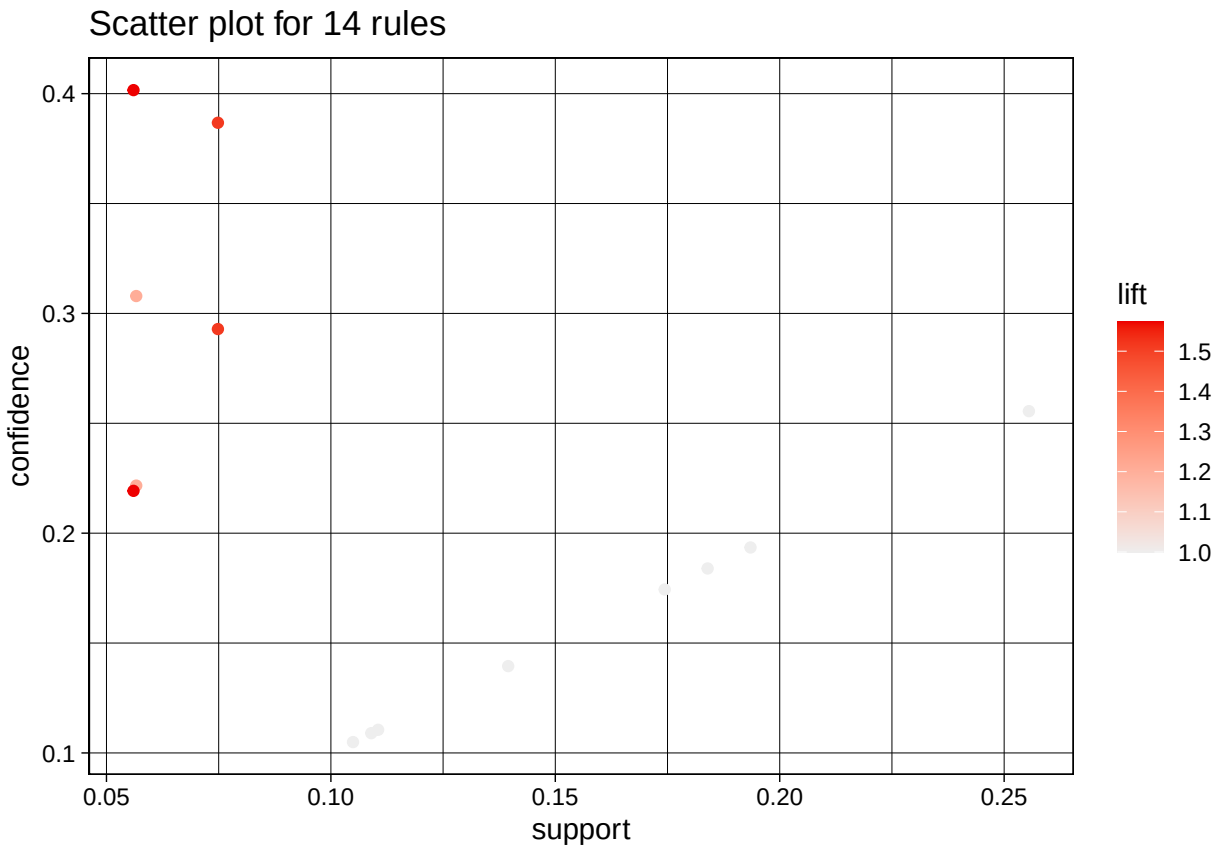
Interpretation

- Rule 1: {Bread} -> {Butter}
- Support : 0.6 (60% of the transactions contain both bread and butter)
- Confidence : 0.8 (80% of the transactions that contain bread also contain butter)
- Lift : 2 (The likelihood of purchasing butter is 2 times higher if bread is bought)

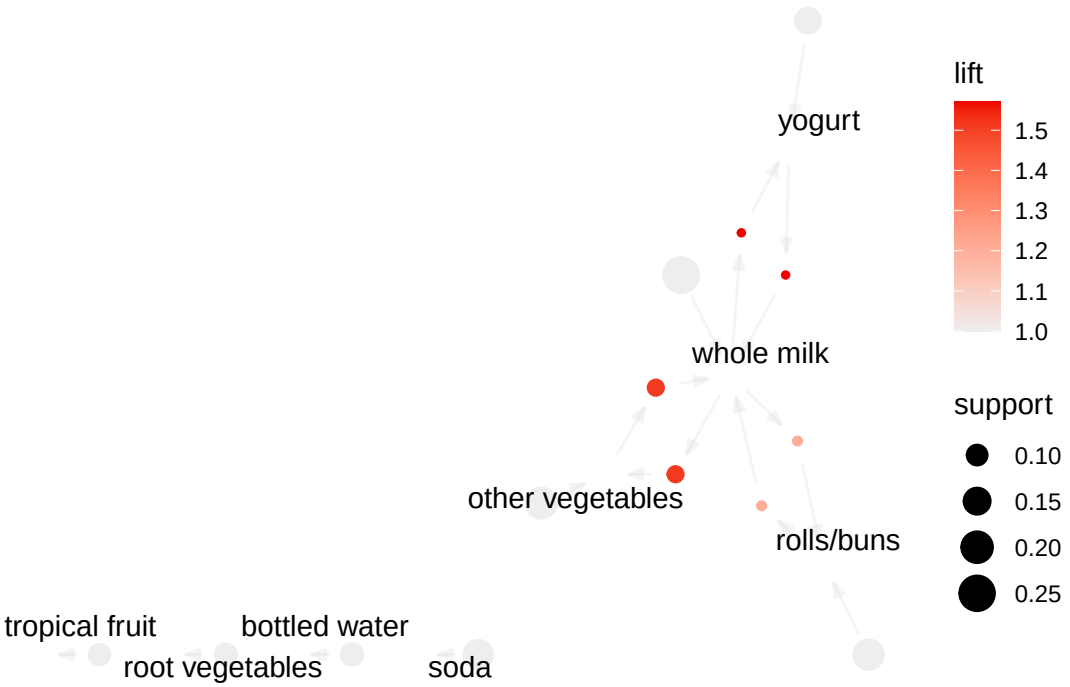
```
sorted_rules = sort(rules, by='lift', decreasing=TRUE)
inspect(sorted_rules[1:5])
```

```
##     lhs                   rhs                support    confidence coverage
## [1] {yogurt}           => {whole milk}       0.05602440 0.4016035  0.1395018
## [2] {whole milk}       => {yogurt}           0.05602440 0.2192598  0.2555160
## [3] {other vegetables} => {whole milk}       0.07483477 0.3867578  0.1934926
## [4] {whole milk}       => {other vegetables} 0.07483477 0.2928770  0.2555160
## [5] {rolls/buns}       => {whole milk}       0.05663447 0.3079049  0.1839349
##     lift     count
## [1] 1.571735 551
## [2] 1.571735 551
## [3] 1.513634 736
## [4] 1.513634 736
## [5] 1.205032 557
```

```
library(arulesViz)
plot(sorted_rules)
```




```
plot(sorted_rules, method = 'graph')
```



```
plot(sorted_rules, method='paracoord')
```