# Market Basket Analysis - II

# Contents

# Targeting Items

Association rules should be used to explain decision making and further utilised to form effective strategies.

Continuing with the example of consumer's buying preferences, the following two questions can be of interest. Reference item is Whole Milk.

What are customers likely to buy if they purchase whole milk?

# Targeting Items

```
library(arules)
data("Groceries")

rules<-apriori(Groceries,parameter=list(supp=0.001,conf=
0.15,minlen=2),appearance=list(default="rhs",lhs="whole
milk"),control=list(verbose=FALSE))

rules<-sort(rules,decreasing=TRUE,by="confidence")

inspect(rules[1:5])
```

- We have already seen the basic arguments used in **apriori().**
- **minlen=** in **parameter=** is used to specify how many items to be considered

  Default is **minlen=1** which means that rules with only one item will be created.
- **appearance=** is used to restrict item appearance.
- **control=** controls the algorithmic performance of mining algorithm.
- **verbose=FALSE** ensures R does not show progress report.

* Here we continue with previous ppt & use the same inbuilt dataset “Groceries”

# Targeting Items

```
# Output
```

| | lhs | | rhs | support | confidence | lift | count |
|---|---|---|---|---|---|---|---|
| [1] | {whole milk} | => | {other vegetables} | 0.075 | 0.29 | 1.5 | 736 |
| [2] | {whole milk} | => | {rolls/buns} | 0.057 | 0.22 | 1.2 | 557 |
| [3] | {whole milk} | => | {yogurt} | 0.056 | 0.22 | 1.6 | 551 |
| [4] | {whole milk} | => | {root vegetables} | 0.049 | 0.19 | 1.8 | 481 |
| [5] | {whole milk} | => | {tropical fruit} | 0.042 | 0.17 | 1.6 | 416 |

**Interpretation:**

- Based on confidence, customers are most likely to move to other vegetables immediately after buying whole milk.

# Visualise Rules

```
#Creating Interactive Graph

library(arulesViz)
rules<-apriori(Groceries,parameter=list(supp=0.001,conf=0.15,minlen=2)
,
appearance=list(default="rhs",lhs="whole
milk"),control=list(verbose=FALSE))

plot(rules,method="graph",interactive=TRUE,shading=NA)
```

# Interactive Graphs

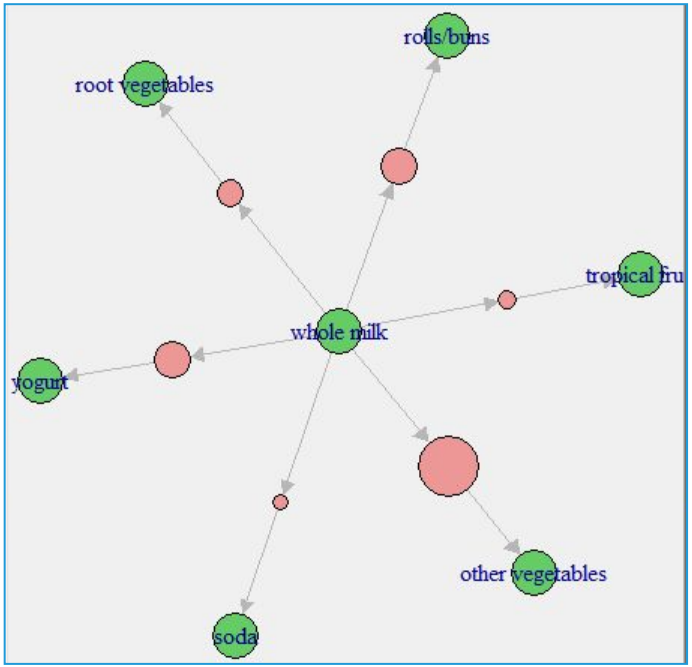
Upon running the command for interactive graph in **`arulesViz`**, a new window opens

Graph uses vertices and edges to visualise rules

- Vertices are itemsets or items
- Edges indicate relationship in rules
- Labels on the edges or colour or width of the arrows displaying edges represent interest measures

Note that graph-based visualisations are viable only for a small set of rules, as more number of rules would make the graph cluttered and difficult to interpret

# Interpreting the Interactive Graph




Out target item is whole milk, which is at the centre of the graph

- The coloured vertex represent confidence. As seen before, confidence for {whole milk, other vegetables} is the highest, followed by yogurt and rolls/buns.
- Lowest confidence in the top five is for {whole milk, tropical fruit}

# Case Study

## Background

- Transactions data collected from point of sales is generally in long format. Arules package requires the data to be in wide format.

## Objective

- To convert available data to a format suitable for association analysis and conduct analysis via arules package in R

## Available Information

- Each transaction is given a unique ID
- Items basket contains five items, items purchased during each transaction are recorded

# Data Snapshot

Transactions Data for MBA

| id | item |
|---|---|
| 1 | B |
| 1 | C |
| 1 | D |
| 1 | E |
| 2 | A |
| 2 | B |
| 2 | C |
| 2 | D |
| 2 | E |
| 3 | A |
| 3 | B |

| Columns | Description | Measurement | Possible values |
|---|---|---|---|
| id | Transaction Id | - | Positive Integers |
| item | Items purchased | A,B,C,D,E | 5 |

# Data Conversion

```
#Convert the Data
trans<-read.transactions("Transactions Data for MBA.csv",format="single",sep=",",cols=c("id","item"),header=TRUE)
```

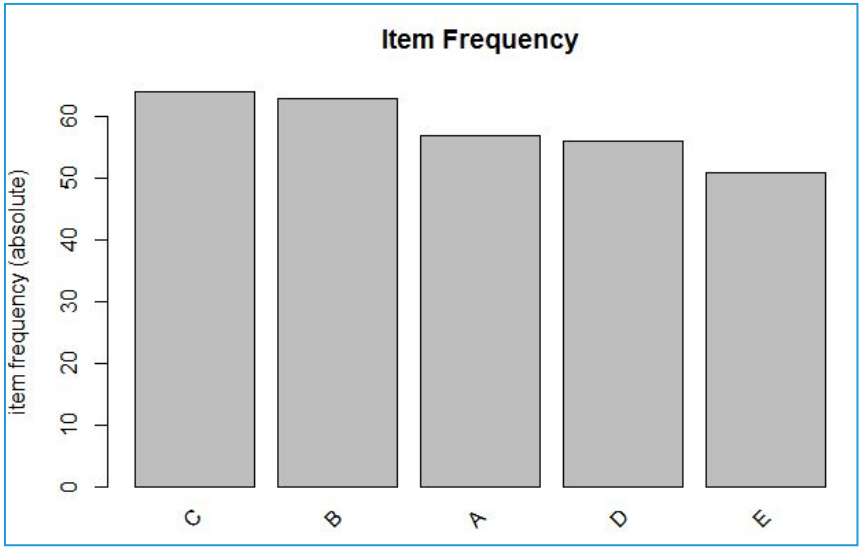
**read.transactions()** in package arules reads a transactions data file and creates a transaction object.
**format=** indicates the format of the dataset. **"single"** implies each line corresponds to a single item, containing at least ids for the transaction and the item. **"basket"** implies each line in the transaction data file represents a transaction where the items (item labels) are separated by the characters specified by **sep**.
For the **"single"** format, **cols=** is a numeric or character vector of length two giving the numbers or names of the columns (fields) with the transaction and item ids, respectively.
**sep=** is a character string specifying how fields are separated in the data file. The default ("") splits at whitespaces.

# Association Analysis

```
#Visualise Frequency
itemFrequencyPlot(trans,topN=5,type="absolute")
```

**itemFrequencyPlot()** calculates item frequency and returns a barplot.
**topN=** instructs R to plot only top N highest item frequency or lift
**type=** is a character string indicating whether item frequencies should be displayed relative of absolute.

```
# Output
```




**Interpretation:**

- The plot shows items by frequency in a descending order.

# Association Analysis

```
#Get the Rules

rules<-apriori(trans,parameter=list(supp=0.001,conf=0.8))

inspect(rules[1:5])
```

**apriori()** is used to mine frequent itemsets, association rules or association hyperedges using this algorithm with specified support and confidence
**inspect()** in package **arules** displays association and additional information formatted for online inspection

# Association Analysis

```
# Output
```

```
Parameter specification:
 confidence minval smax arem  aval originalSupport maxtime support
        0.8    0.1    1 none FALSE            TRUE       5   0.001
 maxlen target   ext
     10  rules FALSE

Algorithmic control:
 filter tree heap memopt load sort verbose
    0.1 TRUE TRUE  FALSE TRUE    2    TRUE

Absolute minimum support count: 0

set item appearances ...[0 item(s)] done [0.00s].
set transactions ...[5 item(s), 100 transaction(s)] done [0.00s].
sorting and recoding items ... [5 item(s)] done [0.00s].
creating transaction tree ... done [0.00s].
checking subsets of size 1 2 3 4 5 done [0.00s].
writing ... [5 rule(s)] done [0.00s].
creating S4 object  ... done [0.00s].
```

**Interpretation:**

The output displays parameter specification, algorithmic control and absolute minimum support count.

It also lists down tasks performed and time taken to complete them.

We are interested in knowing how many rules are created; Here 5 rules are created.

# Association Analysis

```
# Output
```

| | lhs | | rhs | support | confidence | lift | count |
|---|---|---|---|---|---|---|---|
| [1] | {A,D} | => | {C} | 0.25 | 0.81 | 1.3 | 25 |
| [2] | {A,D,E} | => | {C} | 0.11 | 0.85 | 1.3 | 11 |
| [3] | {A,B,E} | => | {C} | 0.10 | 0.83 | 1.3 | 10 |
| [4] | {A,B,D} | => | {C} | 0.16 | 0.84 | 1.3 | 16 |
| [5] | {A,B,D,E} | => | {C} | 0.06 | 1.00 | 1.6 | 6 |

**Interpretation:**

- **inspect()** returns list of lhs and rhs items, their support, confidence and lift values

# Quick Recap

In this session, we learnt **Market Basket Analysis:**

**Market Basket Analysis**

- Mathematical modeling technique based upon the theory that if you buy a certain group of items, you are likely to buy another group of items
- Transactions, Support, Confidence and Lift are the key concepts used in this analysis
- The analysis is performed by creating and studying rules based on different itemsets

**Market Basket Analysis in R**

- Package **`arules`** and **`arulesViz`** are used for undertaking MBA
- **`itemFrequencyPlot()`** plots frequency
- **`apriori()`** function creates rules. **`inspect()`** displays association and additional information
- **`read.transactions()`** converts raw point of sales transactions data to a wide-format transactions object
- **`plot()`** in **`arulesViz`** can create static or interactive plots