Business Data Analytics. Practice Session
Market Basket Analysis

Source: See [1]

#Loading the packages
library(arules)
library(arulesViz)# For visualization
## There are issues in simply installing arulesViz
# First try to install viridisLite and then arulesViz and load in the same order. If that does not
# work, follow: following list of packages
# dendextend, Scatterplot3d , viznetwork, vcd, viridisLite.
# See this: https://stackoverflow.com/questions/48962946/error-package-or-namespace-load-
failed-for-arulesviz-object-cividis-is-not?rq=1

##Reading the data file
mydata< - read.csv("C:\\Users\\brai\\Desktop\\Cosmetics.csv",header=T)
summary(mydata)
names(mydata)
head(mydata)
Scatterplot3d , viznetwork, vcd, viridisLite.

##Finding association rules
rules <- apriori(mydata)
# Around 68000 rules are generated with default values. Confidence = 0.8 , Support = 0.1

# Let us see summary of the rules
summary(rules)

##Rules with specified parameter valus
rules <- apriori(mydata,parameter = list(minlen=2, maxlen=10,supp=.7, conf=.8))
inspect(rules)# List down all the rules
# If you observe the output, they are NO rules. That is what they are NOT purchasing.

## Finding interesting rules

```
rules <- apriori(mydata, parameter = list(minlen=2, maxlen=3, supp=.01, conf=.7), appearance=list(rhs=c("Foundation=Yes"), lhs=c("Bag=Yes", "Blush=Yes"), default="lhs"))  # Try summary(mydata) to see "Foundation" is the number one item.
inspect(rules)
```

## Visualization

### Graphs and Charts

```
plot(rules)
plot(rules, method="grouped")
plot(rules, method="graph", control=list(type="items"))
```

## Finding interesting rules-2

## Only for items which were really pucahes. (Check rule 11)

```
quality(rules) <- round(quality(rules), digits=3)
inspect(rules.sorted)
```

# Sort them by lift

```
rules.sorted <- sort(rules, by="lift")
inspect(rules.sorted)
```

## Finding redundancy

```
subset.matrix <- issubset(rules.sorted, rules.sorted)
subset.matrix[lower.tri(subset.matrix, diag=T)] <- NA
redundant <- colSums(subset.matrix, na.rm=T) >= 1
which(redundant)
```

## Removing redundant rules

```
rules.pruned <- rules.sorted[!redundant]
inspect(rules.pruned)
```

References:

[1] https://www.youtube.com/watch?v=91CmrpD-4Fw&t=13s