Q8

Raghav Vaidya

2023-08-13

Question

Revisit the notes on association rule mining and the R example on music playlists: playlists.R and playlists.csv. Then use the data on grocery purchases in groceries.txt and find some interesting association rules for these shopping baskets. The data file is a list of shopping baskets: one person's basket for each row, with multiple items per row separated by commas. Pick your own thresholds for lift and confidence; just be clear what these thresholds are and say why you picked them. Do your discovered item sets make sense? Present your discoveries in an interesting and visually appealing way.

Notes:

This is an exercise in visual and numerical story-telling. Do be clear in your description of what you've done, but keep the focus on the data, the figures, and the insights your analysis has drawn from the data, rather than technical details. The data file is a list of baskets: one row per basket, with multiple items per row separated by commas. You'll have to cobble together your own code for processing this into the format expected by the "arules" package. This is not intrinsically all that hard, but it is the kind of data-wrangling wrinkle you'll encounter frequently on real problems, where your software package expects data in one format and the data comes in a different format. Figuring out how to bridge that gap is part of the assignment, and so we won't be giving tips on this front.

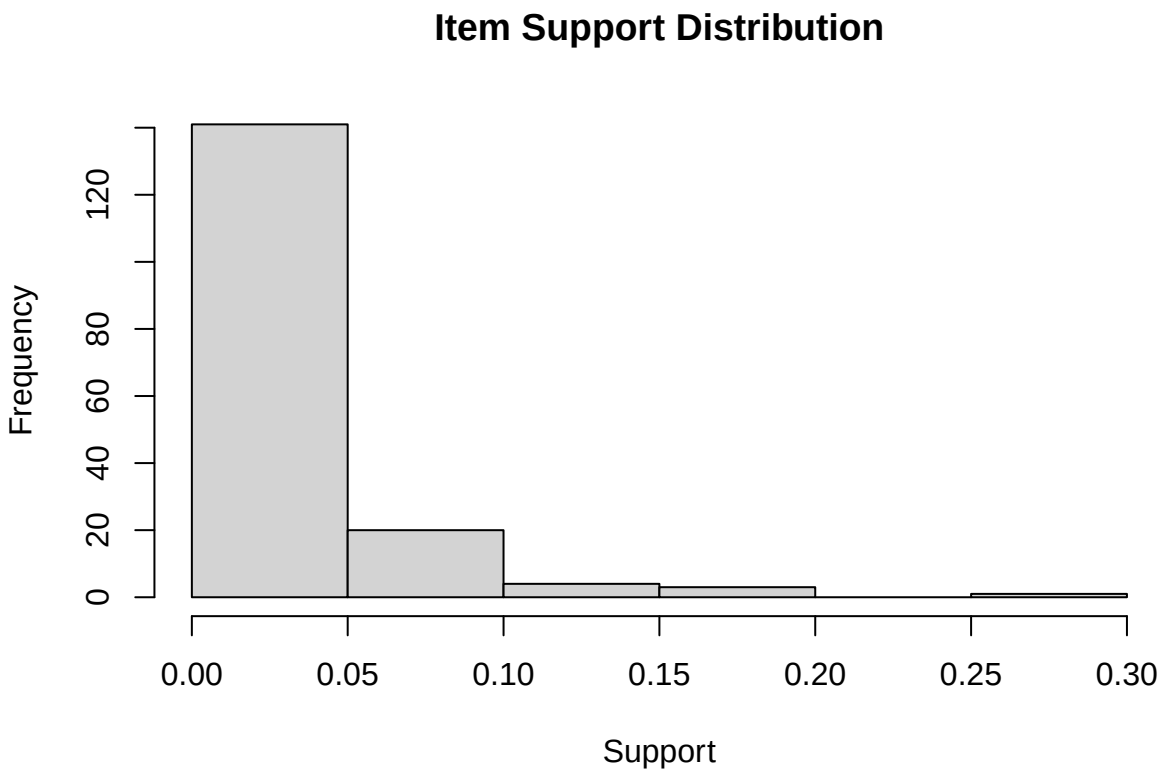
Answer

First we read the text file given and convert it into "transaction" class form.

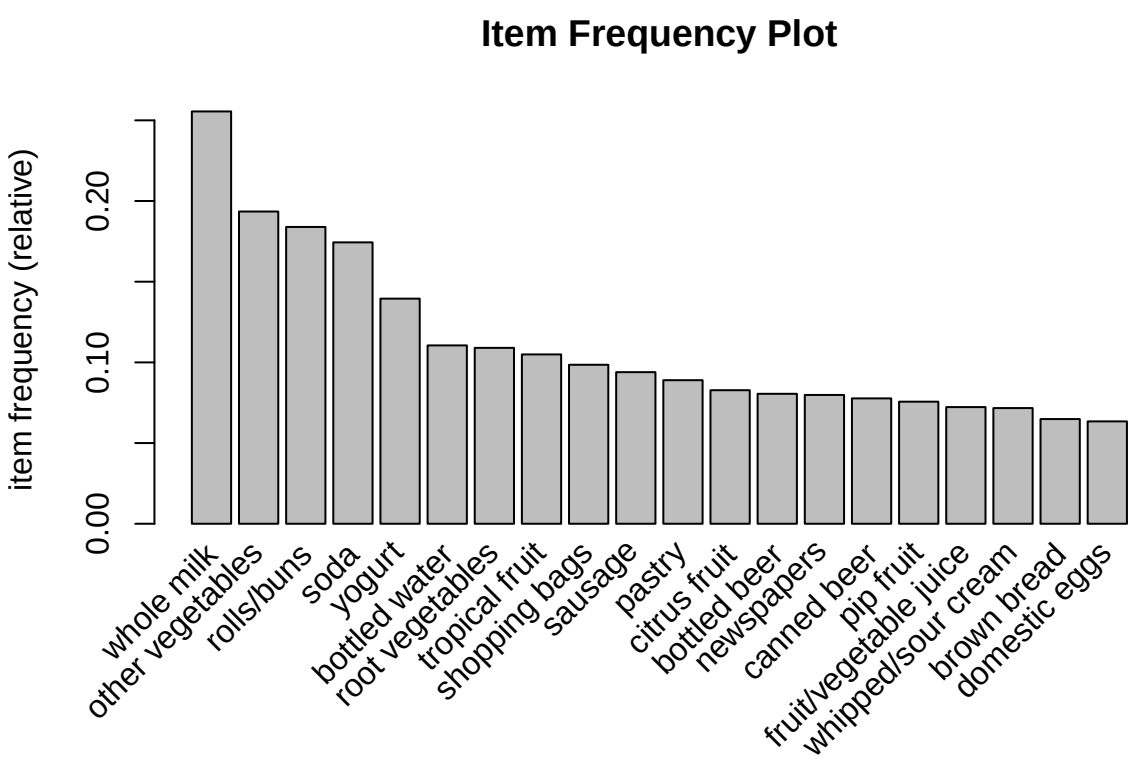
```
#Reading the Groceries text file, splitting and then converting to "transaction class"
transactions <- readLines("groceries.txt")
transactions <- strsplit(transactions, split = ",")
transactions <- as(transactions, "transactions")
```

Now we'll plot Item Support Distribution Histogram to identify the Support frequency of Single Items. We also make a Item Frequency Plot to determine most common items.

```
#Plotting Single item support distribution
item_support <- itemFrequency(transactions)
hist(item_support, main = "Item Support Distribution", xlab = "Support")
```




```r
#Item Frequency Plot
itemFrequencyPlot(transactions, topN = 20, type = "relative", main = "Item Frequency Plot")
```




```
#Getting info on transactions and support values
summary(transactions)
```

```
## 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
##             2513             1903             1809             1715
##           yogurt          (Other)
##             1372            34055
##
## element (itemset/transaction) length distribution:
## sizes
##    1    2    3    4    5    6    7    8    9   10   11   12   13   14   15   16
## 2159 1643 1299 1005  855  645  545  438  350  246  182  117   78   77   55   46
##   17   18   19   20   21   22   23   24   26   27   28   29   32
##   29   14   14    9   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
## 1 abrasive cleaner
```

```
## 2 artif. sweetener
## 3   baby cosmetics
```

```
#Getting info on support value distribution
summary(item_support)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## 0.0001017 0.0038637 0.0104728 0.0260915 0.0310117 0.2555160
```

As we can see, Whole Milk is the most common item occuring in more than 25% of the transactions.
Now to start with rule mining, let's take the support as 0.025 which is near the mean of item support distribution. Now we'll look for association rules by setting initial parameters as 2 for min-len and 0.3 for confidence. We set 2 because upon seeing the length distribution of the items we see that majority of grocery lists have a length of at least 2 (i.e 2 items in the bag). 1 might be too low and 3 might be too high for an initial parameter. Also let's start confidence from 0.1 and slowly make our way up to 0.8 which is considered good for most cases.

```
#Plotting for Top 10
rules <- apriori(transactions, parameter = list(supp = 0.025, conf = 0.1, target = "rules", minlen = 2))
```

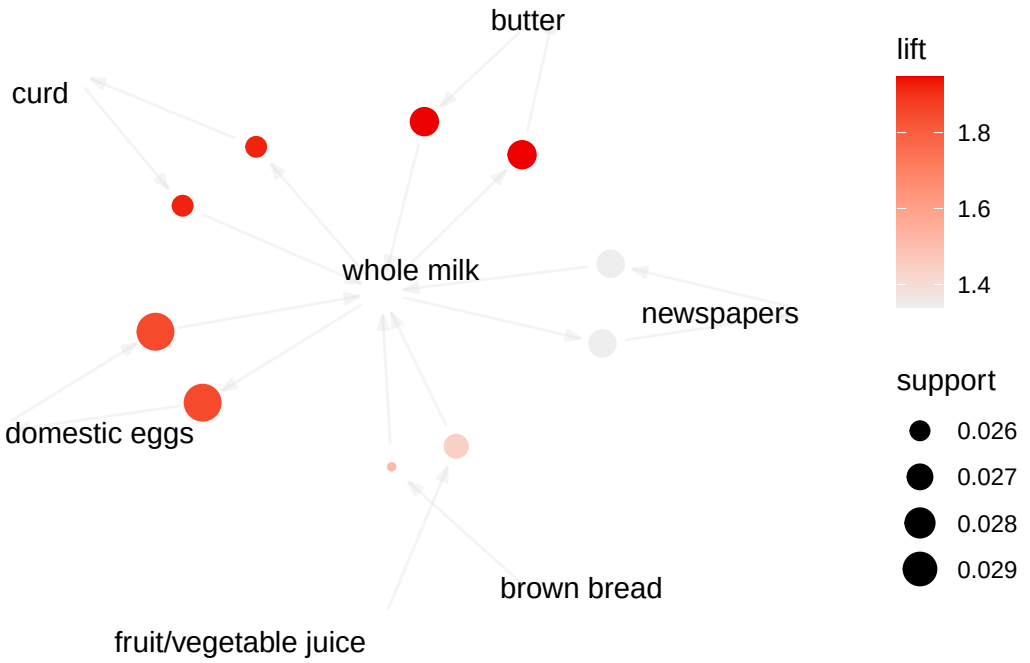
```
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.1    0.1    1 none FALSE            TRUE       5   0.025      2
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 245
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[169 item(s), 9835 transaction(s)] done [0.00s].
## sorting and recoding items ... [54 item(s)] done [0.00s].
## creating transaction tree ... done [0.00s].
## checking subsets of size 1 2 3 done [0.00s].
## writing ... [67 rule(s)] done [0.00s].
## creating S4 object  ... done [0.00s].
```

```
plot(rules[1:10], method = "graph", control = list(type = "items"))
```

```
## Warning: Unknown control parameters: type
```

```
## Available control parameters (with default values):
## layout    =  stress
## circular  =  FALSE
## ggraphdots    =  NULL
```

```
## edges     =  <environment>
## nodes     =  <environment>
## nodetext  =  <environment>
## colors    =  c("#EE0000FF", "#EEEEEEFF")
## engine    =  ggplot2
## max   =  100
## verbose   =  FALSE
```



We got 67 rules. Most of the nodes are going in and out of Whole Milk so this graph is not much useful. Now let's try to reduce the Support to 1st Qt which is around 0.015 and increase confidence to 0.2.

```
#Plotting for Top 10
rules <- apriori(transactions, parameter = list(supp = 0.015, conf = 0.2, target = "rules", minlen = 2))
```

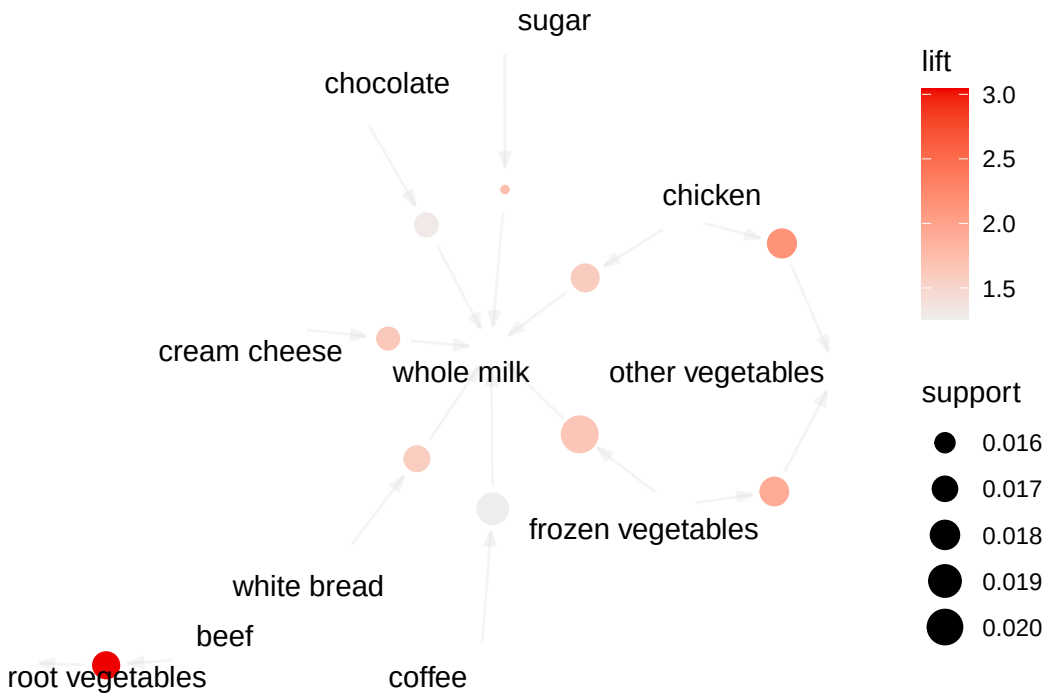
```
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.2    0.1    1 none FALSE            TRUE       5   0.015      2
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
```

```
## Absolute minimum support count: 147
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[169 item(s), 9835 transaction(s)] done [0.00s].
## sorting and recoding items ... [73 item(s)] done [0.00s].
## creating transaction tree ... done [0.00s].
## checking subsets of size 1 2 3 done [0.00s].
## writing ... [115 rule(s)] done [0.00s].
## creating S4 object  ... done [0.00s].
```

```
plot(rules[1:10], method = "graph", control = list(type = "items"))
```

```
## Warning: Unknown control parameters: type

## Available control parameters (with default values):
## layout    =  stress
## circular  =  FALSE
## ggraphdots    =  NULL
## edges     =  <environment>
## nodes     =  <environment>
## nodetext  =  <environment>
## colors    =  c("#EE0000FF", "#EEEEEEFF")
## engine    =  ggplot2
## max   =  100
## verbose   =  FALSE
```



We get 115 rules with way items. We see that Whole Milk is still a common occurrence. Let's see how much further we can improve. We'll try decreasing the support ten fold and increasing confidence to 0.8.

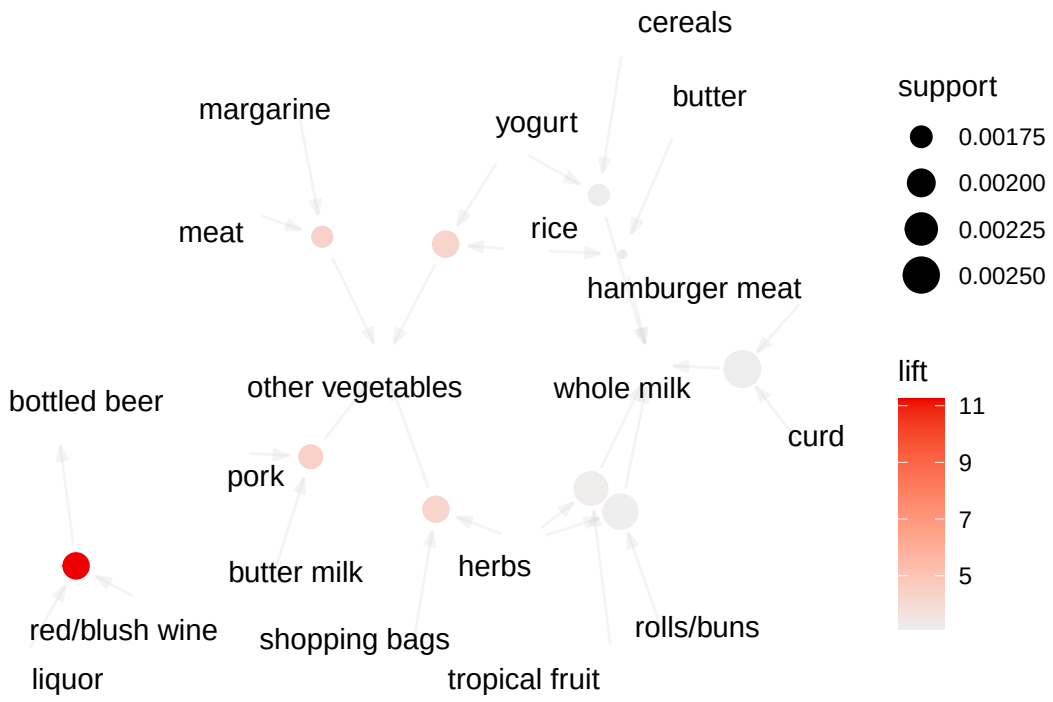
```
#Plotting for Top 10
rules <- apriori(transactions, parameter = list(supp = 0.0015, conf = 0.8, target = "rules", minlen = 2)
```

```
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.8    0.1    1 none FALSE            TRUE       5  0.0015      2
##  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: 14
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[169 item(s), 9835 transaction(s)] done [0.00s].
## sorting and recoding items ... [153 item(s)] done [0.00s].
## creating transaction tree ... done [0.00s].
## checking subsets of size 1 2 3 4 5 6 done [0.00s].
## writing ... [60 rule(s)] done [0.00s].
## creating S4 object  ... done [0.00s].
```

```
plot(rules[1:10], method = "graph", control = list(type = "items"))
```

```
## Warning: Unknown control parameters: type
```

```
## Available control parameters (with default values):
## layout    =  stress
## circular  =  FALSE
## ggraphdots    =  NULL
## edges     =  <environment>
## nodes     =  <environment>
## nodetext  =  <environment>
## colors    =  c("#EE0000FF", "#EEEEEEFF")
## engine    =  ggplot2
## max   =  100
## verbose   =  FALSE
```



We got 60 rules.
We can conclude a few things from the graphs:
1. People are likely to purchase root vegetables if they purchase beef.
2. If people purchase chicken or frozen vegetables, they will likely also buy other vegetables.
3. If Red/blush wine or liquor is bought, bottled beer is extremely likely to be purchased.

The above observations do make sense to a certain extent.
1. Root vegetables are an important ingredient for beef stew which is a popular dish, hence they are bought together with beef.
2. For the second observation- we can not reason certainly as we don't know what other vegetables consist of but vegetables- frozen or unfrozen are important ingredients for many chicken dishes.
3. Finally for the third observation, it is evident that people who drink alcohol like most types of alcohols. Hence people who indulge in drinking tend to buy such items together. One other explanation is that people tend to buy a variety of alcohols for social events such as parties, get-togethers etc. Hence people buy a lot of these items together, usually in bulk, for such events.